Università di Genova

DIPARTIMENTO DI INFORMATICA, BIOINGEGNERIA, ROBOTICA E INGEGNERIA DEI SISTEMI

# Markerless Human Motion Analysis

Matteo Moro

Theses Series DIBRIS-TH-2022-42

Università di **Genova**

Dipartimento di Informatica, Bioingegneria,
Robotica ed Ingegneria dei Sistemi

Ph.D. Thesis in
Computer Science and Systems Engineering
Computer Science Curriculum

# Markerless Human Motion Analysis

by

Matteo Moro

April, 2022

Ph.D. Thesis in Computer Science and Systems Engineering (S.S.D. INF/01)
Computer Science Curriculum
Dipartimento di Informatica, Bioingegneria,
Robotica ed Ingegneria dei Sistemi (DIBRIS)
Università di Genova

***Candidate***
Matteo Moro
`matteo.moro@edu.unige.it`

***Title***
Markerless Human Motion Analysis

***Advisors***
Francesca Odone
DIBRIS, Università di Genova
`francesca.odone@unige.it`

Maura Casadio
DIBRIS, Università di Genova
`maura.casadio@unige.it`

***External Reviewers***
Henry Medeiros
Agricultural and Biological Engineering
University of Florida
`hmedeiros@ufl.edu`

Alessandro Bevilacqua
Dipartimento di Informatica - Scienza e Ingegneria
Università di Bologna
`alessandro.bevilacqua@unibo.it`

***Location***
DIBRIS, Università di Genova
Via Dodecaneso, 35
I-16145 Genova, Italy

***Submitted On***
April 2022

# Abstract

Measuring and understanding human motion is crucial in several domains, ranging from neuroscience, to rehabilitation and sports biomechanics. Quantitative information about human motion is fundamental to study how our Central Nervous System controls and organizes movements to functionally evaluate motor performance and deficits. In the last decades, the research in this field has made considerable progress. State-of-the-art technologies that provide useful and accurate quantitative measures rely on marker-based systems. Unfortunately, markers are intrusive and their number and location must be determined a priori. Also, marker-based systems require expensive laboratory settings with several infrared cameras. This could modify the naturalness of a subject's movements and induce discomfort. Last, but not less important, they are computationally expensive in time and space. Recent advances on markerless pose estimation based on computer vision and deep neural networks are opening the possibility of adopting efficient video-based methods for extracting movement information from RGB video data. In this contest, this thesis presents original contributions to the following objectives: (i) the implementation of a video-based markerless pipeline to quantitatively characterize human motion; (ii) the assessment of its accuracy if compared with a gold standard marker-based system; (iii) the application of the pipeline to different domains in order to verify its versatility, with a special focus on the characterization of the motion of preterm infants and on gait analysis. With the proposed approach we highlight that, starting only from RGB videos and leveraging computer vision and machine learning techniques, it is possible to extract reliable information characterizing human motion comparable to that obtained with gold standard marker-based systems.

# Contents

# 1

# Introduction

## 1.1 Topic overview

Human motion understanding is a relevant task in many fields of science and medicine. Quantitative and qualitative motion analysis, *e.g.*, predicting and describing human behavior while performing different actions, is essential in neuroscience to understand the brain behaviour in both physiological and pathological conditions [Bateson and Martin, 2021; Chambers et al., 2020; Moro et al., 2020; Reich et al., 2021]. Moreover, it is helpful for human-computer interaction applications, where a computer can be controlled with dedicated gestures [Betke, Gips, and Fleming, 2002; Fu and Huang, 2007; Moro et al., 2021b], for human-robot interaction, where a robot can detect changes in human keypoints to provide dedicated assistance [Droeschel and Behnke, 2011; Narayanan et al., 2020] and for augmented reality applications for gaming and rehabilitation [Kang et al., 2020; Song, Demirdjian, and Davis, 2012]. Lastly, human motion understanding is largely adopted in proxemic recognition in order to study and predict how people interact [Kim et al., 2021; Wang et al., 2020].

## 1.2 Motivations

Nowadays, the gold standard techniques commonly adopted to accurately characterize and study human motion rely on wearable sensors, motion capture systems and physical markers placed on the body skin [Lopez-Nava and Muñoz-Meléndez, 2016] (see Figure 1.1 for an example of a standard setup). However, markers are intrusive, they may limit natural movements, and their location must be assigned *a priori* by expert operators, making the study of human motion biased [Carse et al., 2013]. Furthermore, they are cumbersome, making the analysis of motion patterns challenging in some application fields such as infants motion analysis [Garello et al., 2021; Meinecke et al., 2006] (see Figure 1.2 for two examples).

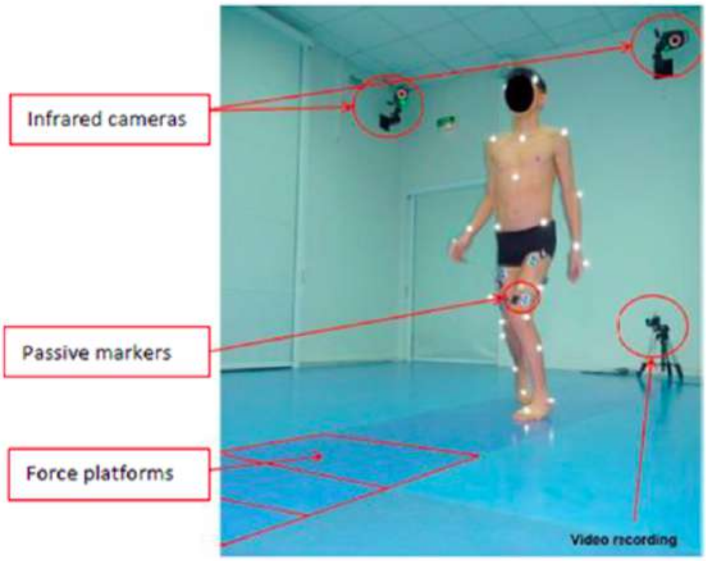


Figure 1.1: Example of marker-based setup for gait analysis. Source: [Khouri and Desailly, 2017].

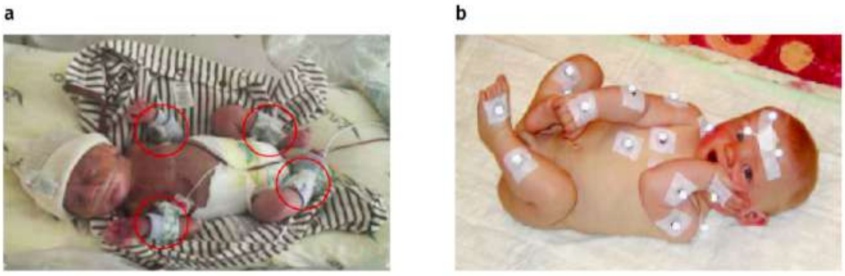


Figure 1.2: Example of application field where markers make the analysis of motion difficult. (a) source: [Fan et al., 2012]; (b) source: [Meinecke et al., 2006].

For these reasons, recently, RGB video analysis has become a possible alternative to marker-based systems to perform human motion analysis [Colyer et al., 2018; Needham et al., 2021]. This is due to the increasing progress – in terms of accuracy and computational resources needed – of deep learning algorithms in solving computer vision problems [Voulodimos et al., 2018]. In particular, recent advances on pose estimation algorithms based on deep neural networks are opening the possibility of adopting efficient methods for tracking human pose and extracting motion information starting from common RGB video data [Zheng et al., 2020]. Pose estimation consists in identifying position and orientation of the subject body in images or image sequences, and it involves body landmark points detection and skeleton estimation. The latter may be carried out by exploiting spatial [Cao et al., 2017; Insafutdinov et al., 2016] or spatio-temporal relationships [Liu et al., 2017]. In this context, it is necessary to study and implement a markerless system able to extract quantitative information related to human motion and to compare the measures obtainable with this system to the ones commonly computed using gold standard marker-based systems.

## 1.3 Objectives

The general aim of this work is to propose and test a new approach for the analysis of human motion. The system overcomes the limitations of current gold standard marker-based systems by leveraging computer vision and deep learning techniques, while maintaining a similar accuracy level. In particular, the work carried out during this Ph.D. had the following main objectives.

1. *Design and implementation of a video-based markerless system to quantitatively characterize human motion*. Indeed, we need to comply with the following requirements: (i) accuracy and precision, (ii) versatility and (iii) interpretability. Firstly, it is necessary to have a system as **precise** and **accurate** as possible in order to detect and highlight all the possible motion patterns (for example, also the fine pathological movements). Then, we need a **versatile** system because our goal is to use the implemented pipeline to study human motion in different situations and with different tasks. Furthermore, we want it to be usable even in cases where marker-based techniques limit the possibility of analysis due to their cumbersome nature. Finally, we need **interpretability**. In fact, it is necessary to fully understand and control all the steps of the analysis.

2. *Evaluation of the accuracy of the implemented pipeline with respect to gold standard marker-based system*. This test is necessary in order to evaluate and measure the sources of systematic error impacting our markerless pipeline. To accomplish that it is necessary to have multimodal datasets that include both marker and video acquisitions simultaneously.

3. *Application of the implemented pipeline*. We select applications from different fields. At the beginning, we focus on two main applications related to the rehabilitation domain: gait analysis and the study of preterm infants' motion patterns. We select these two applications because they present different challenges. The first one is important because it is a well known and standard procedure commonly used in the rehabilitation field to highlight and monitor motion patterns in people with neurological diseases, *e.g.*, stroke, multiple sclerosis or Parkinson [Biase et al., 2020]. Furthermore, gait analysis results can be used to tailor appropriate and specific rehabilitation treatments. For these reasons, gait analysis would benefit from a more accessible system based on RGB cameras. The main challenges of this application are: (i) the fact that, since there are largely used and well defined protocols, it is necessary to produce results as accurate as possible; (ii) the need for a 3D analysis, which requires a multi-view camera system. The analysis of preterm infants motion is essential to detect and highlight the presence of abnormal motion patterns due to lesions involving areas of the brain [Prechtl, 1990] intended for

the control of movement and posture. An early diagnosis of pathological cases would allow the start of early rehabilitation treatment that could significantly increase the chances of recovery. For these reasons, a less obtrusive technique with respect to marker-based systems is necessary to increase the accessibility of this procedure. In this case, the main challenges of this application are: (i) the low number of related works and (ii) the need to create an interpretable pipeline without well established guidelines.

## 1.4 Publications

The work carried out during this Ph.D. led to the following publications.

- Moro M., Marchesi G., Odone F. and Casadio M., (2020, March). Markerless gait analysis in stroke survivors based on computer vision and deep learning: a pilot study. In Proceedings of the 35th Annual ACM Symposium on Applied Computing (pp. 2097-2104) [Moro et al., 2020] (Chapter 6)

- Moro M., Casadio M., Mrotek L. A., Ranganathan R., Scheidt R., and Odone F. (2021, September). On The Precision Of Markerless 3d Semantic Features: An Experimental Study On Violin Playing. In 2021 IEEE International Conference on Image Processing (ICIP) (pp. 2733-2737). IEEE. [Moro et al., 2021a] (Chapter 7)

- Moro M., Rizzoglio F., Odone F., and Casadio M. (2021, January). A Video-Based MarkerLess Body Machine Interface: A Pilot Study. In International Conference on Pattern Recognition (pp. 233-240). Springer, Cham. [Moro et al., 2021b] (Chapter 7)

- Garello L., Moro M., Tacchino C., Campone F., Durand P., Blanchi I., Moretti P., Casadio M. and Odone F. (2021, august). A Study of At-term and Preterm Infants' Motion Based on Markerless Video Analysis. In Proceedings of 29th European Signal Processing Conference, EUSIPCO [Garello et al., 2021] (Chapter 5)

- Garbarino D., Moro M., Tacchino C., Moretti P., Casadio M., Odone F. and Barla A. (2021, November). Attributed Graphettes-based Preterm Infants Motion Analysis. In 10th International Conference on Complex Networks and their Applications. [Garbarino et al., 2021] (Chapter 5).

- Moro, M., Marchesi, G., Hesse, F., Odone, F., and Casadio, M. (2022). Markerless vs. Marker-Based Gait Analysis: A Proof of Concept Study. Sensors [Moro et al., 2022] (Chapter 6)

And to the following works under review:

- Pastore V. P., Moro M., Odone F. VisionTool: a semi-automatic tool for effective semantic feature extraction. *Scientific Reports* (Chapter 8)

- Moro M., Pastore V. P., Tacchino C., Durand P., Blanchi I., Moretti P., Odone F. and Casadio M. A Markerless Pipeline to Analyze Spontaneous Movements of Preterm Infants. *Computer Methods and Programs in Biomedicine* (Chapter 5)

## 1.5 Thesis overview

This Thesis is structured in two main parts.

1. In Part I (*Background and Proposed Approach*), we focus on the background behind human motion analysis and on the proposed markerless approach. In particular, in Chapter 2 we present state-of-the-art methods for marker-based analysis (*e.g.*, motion capture systems and wearable sensors). Then, in Chapter 3 we present an overview of the current methods for markerless human motion analysis. Lastly, in Chapter 4 we report in details the method developed and tested during this project.

2. In Part II (*Applications*), we present all the applications and the tests to assess the reliability of the proposed markerless pipeline. Firstly, we highlight the results obtained by applying the implemented pipeline to the study and the characterization of preterm infants' spontaneous motion (Chapter 5) and to gait analysis (Chapter 6). Then, in Chapter 7 we show the results for other application examples in order to highlight the versatility of the proposed approach. Lastly, we describe the tool we developed to have full control of semantic features detection (Chapter 8).

PART I

# Background and Proposed Approach

This part of the document covers a detailed presentation of state-of-the-art methods for human motion analysis. Firstly, we focus on gold standard techniques based on wearable sensors and markers. Then, we present markerless systems and their relative advantages and disadvantages with respect to the gold standard techniques. Lastly, we describe our proposed markerless method to perform human motion analysis.

2

# Standard Approaches for Motion Analysis

In this chapter we briefly review state-of-the-art methods for human motion analysis. Markers and motion capture systems are considered the gold standards.

## 2.1 Human motion analysis

Human motion analysis is defined as the quantitative and/or qualitative description of motor patterns. Among the most common applications of human motion analysis we can mention medical evaluation, monitoring people, and action classification and recognition. In medicine and rehabilitation, particularly interesting is the quantitative evaluation, since it allows to compute biomechanical variables, such as joint angles and spatio-temporal gait parameters. Moreover, the quantitative characterization of human motion helps expert physicians to monitor motion patterns after orthopedic injuries and in people with neurological diseases, *e.g.*, stroke, spinal cord injury, multiple sclerosis or Parkinson [Biase et al., 2020]. Furthermore, it can be used to tailor appropriate and specific rehabilitation treatments. In this context, quantitative assessments ensure repeatability and objectivity of the analysis, compared to visual observations only [Wren et al., 2020].

The accurate characterization of human movements is commonly performed with motion capture systems and markers. These systems are considered as gold standard. Among the possible alternatives, we can mention wearable sensors (such as Inertial Measurements Units (IMU)), depth cameras and, recently, also RGB cameras.

## 2.2 Markers and motion capture systems

To extract quantitative information about human motion, a gold standard has been established. A gold standard is a method which is commonly accepted as the best one to achieve a specific goal. For human motion analysis the gold standard is stereophotogrammetry, consisting in a 3D motion capture system based on infrared markers and infrared cameras. In detail, the infrared markers are attached to the body skin according to a strict protocol. Multiple protocols have been defined depending on the task under study [Ferrari, Marin-Jimenez, and Zisserman, 2008]. Infrared markers can be (i) active or (ii) passive. In the former, they generate beams of infrared light. In the latter, they reflect the infrared light generated by a different source. A set of infrared cameras (usually at least 8), distributed across the room, capture the infrared signals coming from the markers (see Figure 1.1 for an example of marker-based setup). The cameras need to be calibrated in order to be able to retrieve the position of the markers in the 3D space. The position of each infrared marker is first detected in each camera's image planes (2D). The different viewpoints are then combined to retrieve the position of the markers in the 3D space. For this reason, these systems require each marker to be visible by at least two or three cameras in order to be clearly detected in 3D space. This becomes a problem if the markers are occluded by body parts or walking aids or if the subject leaves the field of view of specific cameras. It is important to note that the cameras can not differentiate which markers they see. Thus, manual work is generally necessary to specify the names of the detected 3D markers, which may lead to significant effort for post-processing the recordings.

The most commonly used commercial motion capture systems are: (i) the Vicon (Vicon Motion Systems Ltd., Oxford, UK) [*Vicon*]; (ii) the Optitrack (NaturalPoint Inc, OR, USA) [*Optitrack*]; (iii) the Optotrack (Northern Digital Inc, Ontario, Canada) [*Optotrack*]. These systems are generally referred to as the gold standards due to their high precision and built-in advanced software.

As mentioned in the introduction, these systems have multiple drawbacks which are frequently mentioned in the literature [Carse et al., 2013; Colyer et al., 2018]:

- they are expensive and, thus, they can not be afforded by many clinicians and researchers;
- they are cumbersome and they can cause discomfort and a lack of naturalness during motion;
- they require trained personnel especially for markers positioning. In fact, markers need to be positioned carefully in specific anatomic points, resulting in a time consuming procedure.

For these reasons, researchers have been looking for markerless alternatives, described in Section 2.3. For further details on stereophotogrametry see [Cappozzo et al., 2005; Chiari et al., 2005a; Della Croce et al., 2005; Leardini et al., 2005; Van Hamersveld et al., 2019; Wade et al., 2022], a series of papers that describe the theoretical background as well as practical details and sources of errors for these systems.

## 2.3 Markerless approaches

Nowadays a big effort is being made to reduce the cost and the invasiveness of the systems for the quantitative characterization of human motion [Colyer et al., 2018]. Markerless technologies have the potential to solve or reduce some of the issues of marker-based approaches, as they are non invasive and they can be built with inexpensive and commonly used hardware. In this section, we report alternatives to marker-based systems to analyze human motion.

wearable sensors. Wearable sensors commonly used in state-of-the-art works in clinical applications are: (i) accelerometers and gyroscopes (*i.e.*, inertial sensors); (ii) magnetometer (*i.e.*, magnetic sensors); (iii) the combination of (i) and (ii) (*e.g.*, Inertial Measurement Unit (IMU)) [Ahmad et al., 2013]. Promising application fields where wearable sensors have been considered as an alternative to marker-based techniques in the study of human motion are: balance [Chiari et al., 2005b; Hasegawa et al., 2021], force localization [Acer and Yıldız, 2018] and, in general, for sensorimotor feedback [Inertial, 2018]. Furthermore, they are largely adopted for human-robot interaction studies [Islam, Xu, and Bai, 2018]. Systems for human motion analysis based on wearable sensors are less expensive and portable, but, unfortunately, they suffer from the same other issues as marker-based approaches. In particular, similar to markers, they are obtrusive and they can affect the naturalness of the motion. For further details on wearable sensors for human motion analysis see [Lopez-Nava and Muñoz-Meléndez, 2016].

video cameras. Thanks to recent improvements in hardware and software technologies in computer vision, the use of video cameras has started to be considered as a possible alternative for human motion analysis. In particular, RGB and depth (RGB-D) cameras have been adopted in many studies [Castelli et al., 2015; Clark et al., 2013; Kidziński et al., 2020; Kwolek et al., 2019; Tsuji et al., 2020; Varol et al., 2021] involving the qualitative and quantitative characterization of human motion. In general, video systems have many advantages with respect to marker-based techniques thanks to their low-cost, portability and minimal invasiveness.

RGB-D cameras (*e.g.*, the Kinect) are sensors that are able to retrieve sparse depth information during acquisition. Kinect (Microsoft, NM, USA) is able to track 3D body keypoints thanks to depth data obtained relying on infrared light, without using obtrusive markers. Nonetheless, these systems may be less accurate with respect to the gold standard. Thus, it is important to evaluate their reliability and their precision level depending on the specific application field. Kinect is the most studied RGB-D sensor and it has already been validated in different contexts [Behrens et al., 2016; Clark et al., 2013; Xu and McGorry, 2015] and its use is not recommended (especially in the medical domain) for accurate quantitative analysis due to its precision and accuracy level [Carmo Vilas-Boas et al., 2019; Mentiplay et al., 2015].

RGB cameras have the same advantages of RGB-D cameras (low cost, portable and not obtrusive) and, in addition, they have the potential to reach higher spatial and temporal resolutions [Pueo, 2016]. Moreover, RGB cameras are common sensors largely adopted also in our everyday life (*e.g.*, in all the modern smartphones and laptops) and there are few studies that quantitatively compare the information extracted with RGB video-based markerless techniques with those retrieved with gold standard marker-based systems [Needham et al., 2021]. Furthermore, in the computer vision literature, we find new methods for automatic analysis of RGB video data that allow to obtain estimates of features (body keypoints) and structures (human skeletons) that can be then adopted to characterize the motion. The background of these system is presented in Chapter 3.

# 3

# Video-based Human Pose Estimation

In this chapter we present the main concepts behind state-of-the-art algorithms for video-based human motion analysis and a detailed description of the actual approach that has been developed and employed.

## 3.1 Problem definition

The first aim of this thesis project is the implementation of a markerless pipeline to characterize human motion. This pipeline should be low cost, easy to use and portable. Moreover, it should not affect in any way the naturalness of the motion. Following these considerations, we adopt common RGB cameras for data acquisition. Starting from RGB videos/images, the first step is the detection of meaningful semantic features (also called landmark points or keypoints) on the human body. These keypoints are then linked in order to build the human skeleton (see Figure 3.1 for an example). The detection of the $(x, y)$ positions in the image plane of semantic body keypoints is the analogous of what is done with marker-based motion capture systems, where a set of infrared cameras is adopted to acquire the $(X, Y, Z)$ positions in the 3D world of physical markers placed on the body skin.

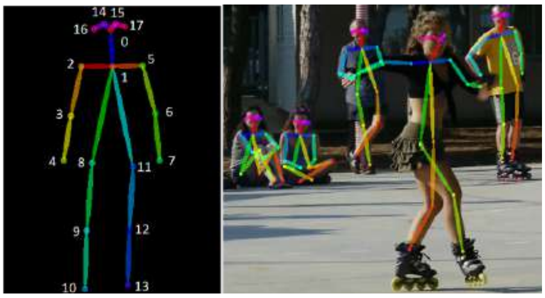
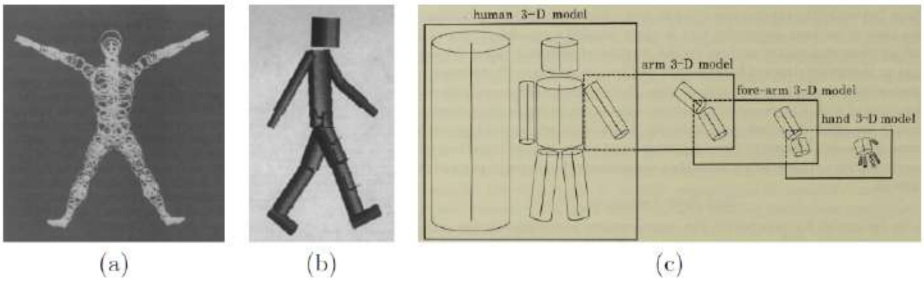


Figure 3.1: Examples of human skeleton with the meaningful body keypoints (left panel) and pose estimation from an RGB image (right panel). Source: [Cao et al., 2017].

## 3.2 Early approaches

Early works about RGB video-based pose estimation rely on model-based approaches and attempt to retrieve directly the 3D model of the human body starting from a single image. One of the first work in this direction is presented in [O'rourke and Badler, 1980]. The authors rely on the human body model definition presented by [Badler and O'Rourke, 1977] and shown in Figure 3.2 (a). The model is built with about 600 sphere primitives and it is composed by 25 articulated joints. The model incorporates angle limits and collision detection. Similarly, in the works done by [Hogg, 1983; Marr and Nishihara, 1978], the authors, instead of sphere primitives, use a collection of hierarchical 3D cylinders to model human body (we show them in Figure 3.2 (b-c)).



(a) (b) (c)

Figure 3.2: Examples of body models. (a) source: [O'rourke and Badler, 1980]; (b) source: [Hogg, 1983]; (c) source: [Marr and Nishihara, 1978].

After these first attempts to extract 3D models from single images, researchers realized the complexity of the task and they started focusing on 2D models. In [Lee and Chen, 1985], the authors recover the stick figure of the body pose starting from known 2D landmark points locations (see Figure 3.3 (a)). Other tools to extract human pose are: (i) pictorial structures, described by [Fischler and Elschlager, 1973] and more recently adopted by [Andriluka, Roth, and Schiele, 2012; Eichner et al., 2012; Ferrari, Marin-Jimenez, and Zisserman, 2008; Ramanan, 2006; Ramanan, Forsyth, and Zisserman, 2005]; (ii) puppet-like representations, described by [Hinton, 1976] and shown in Figure 3.3 (b); (iii) human silhouettes (see for example [Felzenszwalb and Huttenlocher, 2005]); (iv) articulated model formed by a variable number of rectangles (an example is presented in [Ronfard, Schmid, and Triggs, 2002] and shown in Figure 3.3 (c)); (v) coarse part-based models (see for instance [Ioffe and Forsyth, 2001]).

Prior to the rise of deep learning and, specifically, to the rise of convolutional neural networks (CNNs), the most used state-of-the-art methods to estimate 2D human poses in RGB images were **deformable part-models** [Felzenszwalb et al., 2009; Felzenszwalb, McAllester, and Ramanan, 2008]. Deformable part-models can be considered as an extension of pictorial structures. The basic idea is to represent the human pose by a collection of rigid *parts*, arranged in a deformable configuration.

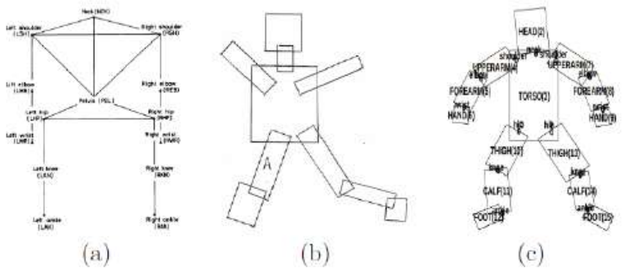


Figure 3.3: Examples of body models representing stick figures (a) and pictorial structures (b-c). (a) source [Lee and Chen, 1985]; (b) source [Hinton, 1976]; (c) source [Ronfard, Schmid, and Triggs, 2002].

## 3.3 2D human pose estimation algorithms

While earlier work focuses mainly on graphical models [Andriluka, Roth, and Schiele, 2009], recent methods on human pose estimation are based on deep learning (*i.e.*, [Newell, Yang, and Deng, 2016; Pishchulin et al., 2016; Toshev and Szegedy, 2014; Wei et al., 2016]). The focus on deep learning and CNNs is also related to the availability of large-scale datasets with 2D pose annotations (*e.g.*, MPII (Max Planck Institut Informatik) [Andriluka et al., 2014] and MS-COCO (Microsoft Common Objects in Context) [Lin et al., 2014b]). Thanks to these labeled datasets, it is possible to implement methods and to reach impressive results on the accuracy of 2D pose estimation.

These algorithms are implemented and structured in order to solve two different problems:

- the detection in the image plane of meaningful body keypoints (for example those shown in the left panel of Figure 3.1), also referred as *semantic features detection* step;
- the construction of the body skeleton by connecting the appropriate pairs of keypoints.

Depending on the logic behind the algorithm for semantic features detection and on the number of people detected in the image plane, there are two main ways to classify CNN-based pose estimation algorithms.

1. ***Top-down vs bottom-up***. Top-down approaches first detect bounding boxes of single human beings in the scene and then detect the desired keypoints inside the image snippets. Generally, for bounding box detection and keypoint detection two different neural networks are adopted. On the other hand, bottom-up approaches proceed the other way round: first they detect all keypoints in the entire image; then they associate all detected body parts with the corresponding persons and they build the skeleton.

2. ***Multi-person vs single-person***. This distinction identifies if an algorithm has been implemented in order to detect and build the skeleton of one or

more people in the image plane. If an image with more than one person is given as input to a single-person algorithm, it will output the skeleton of only one of them depending on the one detected first. As a general statement, multi-person algorithms are more difficult to be implemented and more challenging because they need also to understand the number of people in the image and how to assign the keypoints to the different people.

### 3.3.1 *Examples of 2D pose estimators*

We now report meaningful CNN-based algorithms for human pose estimation starting from RGB images.

DEEPPOSE. DeepPose [Toshev and Szegedy, 2014] is the first CNN-based algorithm for human pose estimation that could reach the performances of previous model-based approaches. DeepPose is a single-person top-down algorithm and it is structured as a regression problem for body keypoints. It starts from a 7 layers AlexNet [Krizhevsky, Sutskever, and Hinton, 2012] and it adds an extra final layer that outputs the keypoints coordinates in the image plane (see left panel of Figure 3.4). DeepPose introduces two main novelties: (i) it is able to detect body landmarks even in presence of occlusions (*i.e.*, in a holistic fashion) and (ii) it refines the keypoints estimates with a cascade of regressors (see right panel of Figure 3.4).

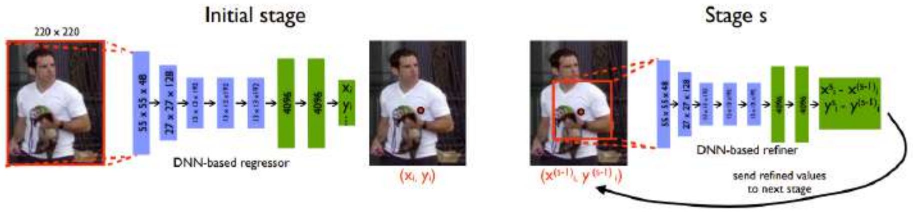


Figure 3.4: DeepPose architecture (source: [Toshev and Szegedy, 2014]). Convolutional layers in blue and fully connected ones in green.

DEEPCUT AND DEEPERCUT. DeepCut [Pishchulin et al., 2016] is the first bottom-up multi-person algorithm for human pose estimation. The implemented architecture jointly solves the two tasks of keypoints detection and keypoints association. It is able to infer the number of people in the image, to identify body landmarks and to disambiguate keypoints between people in close proximity of each other (see Figure 3.5). The keypoints detector is built with a combination of Fast Region-based Convolutional Network (Fast R-CNN) [Girshick, 2015] and VGG [Simonyan and Zisserman, 2014].

DeeperCut [Insafutdinov et al., 2016] is the updated version of DeepCut. Specifically, the authors improve the keypoints detector by replacing the architecture composed by the Fast R-CNN and the VGG with a Residual Network (ResNet-152) [He et al., 2016]. Moreover, they propose novel imageconditioned pairwise terms that allow to increase the accuracy of the connections between pairs of keypoints.

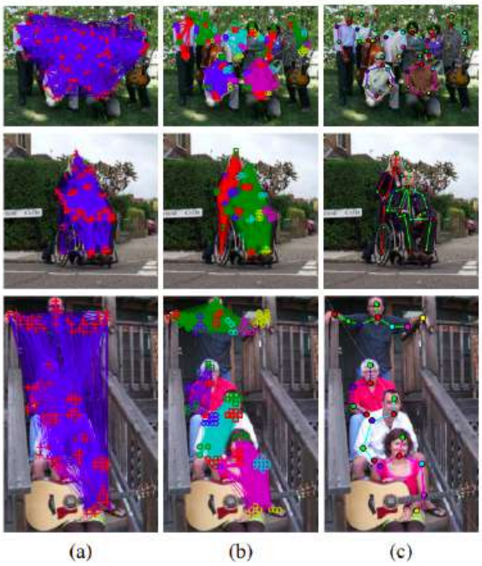


Figure 3.5: (a) Initial keypoints estimates and all the possible connections among them; (b) the connections are jointly clustered in order to assign them to a person (one colored subgraph = one person) and each keypoint is labeled (different colors and symbols correspond to different body keypoints); (c) predicted poses (source: [Pishchulin et al., 2016]).

OPENPOSE. OpenPose [Cao et al., 2017] is one of the first open source systems for multi-person 2D pose estimation that works in real-time. It is a bottom-up architecture that introduces the use of non-parametric representations known as Part Affinity Fields (PAFs) in order to solve the task of keypoints association to build the human skeleton. As shown in Figure 3.6, Openpose takes an image as input and predicts confidence maps for detecting body landmarks and PAFs for landmarks associations. The network architecture (reported in Figure 3.7) iteratively predicts PAFs and confidence maps and then combines them in order to build the final skeleton(s). The authors provide the code and the weights of the architecture trained on different big datasets: MPII human pose [Andriluka et al., 2014] and MS-COCO [Lin et al., 2014b].

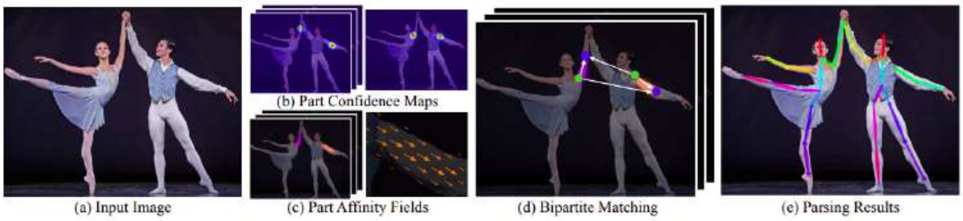


Figure 3.6: Example of Part Affinity Fields and confidence maps extracted with Openpose (source: [Cao et al., 2017]).

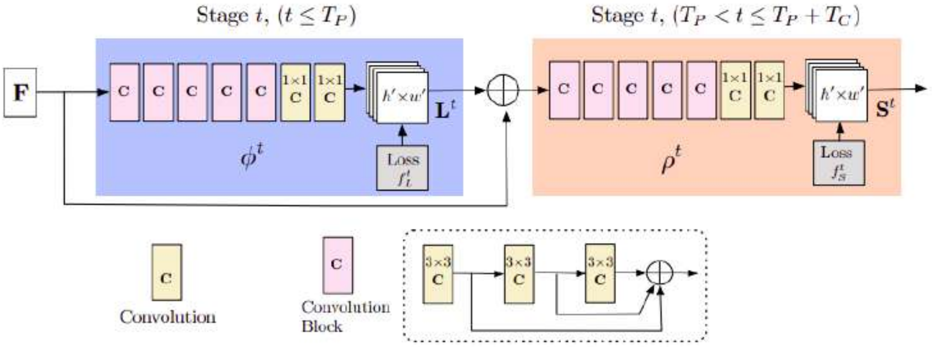


Figure 3.7: Openpose architecture (source: [Cao et al., 2019]). The first part (blue rectangle, $\Phi^t$) predicts Part Affinity Fields (PAFs) $L^t$ and the second one (orange rectangle, $p^t$) predicts confidence maps $S^t$ for each keypoint. The input $F$ represents a set of feature maps obtained by analyzing the original image with a CNN initialized by the first 10 layers of VGG-19 [Simonyan and Zisserman, 2014].

HIGH-RESOLUTION NETWORK (HRNET). High-Resolution Network (HRNet) is proposed by [Sun et al., 2019] and it is a top-down multi-person pose estimation approach that outperforms all the previous algorithms. This architecture differs from the other approaches because it starts from a high-resolution subnetwork, and it gradually adds lower resolution subnetworks to form more stages and connect the different subnetworks in parallel (the architecture is summarized in Figure 3.8).

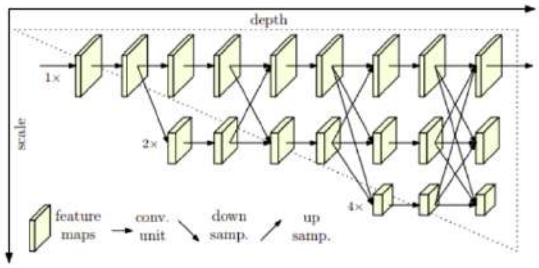


Figure 3.8: HRNet architecture (source: [Sun et al., 2019]).

MEDIAPIPE AND MOVENET. Mediapipe [Bazarevsky et al., 2020; *MediaPipe*] and MoveNet [*Pose Detection with MoveNet and TensorFlow js*] are two of the latest pose estimation architectures released. They are bottom-up single-person architecture implemented by Google's researchers. With respect to previous works, these two algorithms have a different objective: they do not intend to improve keypoints detection or pairwise connections accuracy, but on the speed of the estimations. In fact, while previous algorithm required a large amount of computational resources, these systems can run real-time also in common laptops without a Graphic Processing Unit (GPU).

For a more detailed review of 2D human pose estimation algorithms, the reader is referred to the survey [Zheng et al., 2020] that is continuously updated at the website [*Deep Learning-Based Human Pose Estimation: A Survey*].

## 3.4 2D semantic features detectors

In some applications, it may be appropriate to detect semantic features only. The main advantage of semantic features detectors with respect to pose estimators is that they require a lower number of labeled examples to self-define new points in the image which should be tracked. Moreover, these points are not limited to keypoints on the human body. This is particularly interesting for the applications presented in this thesis, since we are interested in the detection of landmark points that are not always those detected in pose estimators.

Semantic features detection is usually structured as an image segmentation task, in the form of a multi-class classification problem. More formally, we represent a dataset as a set of N images $I = \{I_0, I_1, .., I_N\}$ with pixels $\boldsymbol{x}(x_1, x_2)$ on a discrete grid $m \times n$ with intensities $\boldsymbol{I}_i(\boldsymbol{x}) \in \mathrm{J} \subset R$. The dataset $I$ is usually split into three separated subsets: $I_{TRAIN}$ for training, $I_{VAL}$ for validation and $I_{TEST}$ for testing. For each training (and validation) image $I_i$, we assume a ground truth is available as a set of binary segmentation masks $M_{Il}$ with pixels intensities $\in$ [0,1]; $l \in \{0, 1, ..., L\}$ represents the semantic label, and $L$ is the number of keypoints to detect. Let $M'_I$ be the cumulative ground truth (GT) matrix, with pixel intensities $\in [0, L]$. A multi-class neural network is trained to learn a function $F : I \rightarrow M'$ that maps each pixel $\boldsymbol{x} \in I$ to its semantic label $l$ with some probability. To maximize such probability, a loss function is defined to estimate the deviation of the network prediction from GT, at each training step (*i.e.*, the training error). To minimize the prediction error, the loss function is decreased iteratively during training, until a defined set of stopping criteria is met.

Usually, in addition to a set of defined keypoints, a background class is added to the set of semantic labels. Thus, each pixel of an image can be assigned either to one of the keypoints classes or to the background. Considering that the number of pixels belonging to the keypoints area are generally significantly less than the ones belonging to the background (*i.e.*, everything in the image which is not a keypoint to detect), the problem becomes an imbalanced multi-class classification problem, and imbalance between classes is handled by using a set of weights for each class, with an inverse proportion with respect to the number of pixels belonging to the specific feature class. The results of this process are maps (one for each class, *i.e.*, one for each keypoint) similar to the GT ones $M'_I$ of the same size of the input image that are called confidence maps (or probability maps). In these maps each pixel intensity corresponds to the confidence of that pixel belonging to the correspondent keypoint.

### 3.4.1 *Examples of 2D semantic features detectors*

We now report meaningful algorithms for semantic features detection. It is worth mentioning that in the literature there are few examples of semantic features detectors.

DEEPLABCUT (DLC). One of the most important features detector is DeepLabCut (DLC) [Mathis et al., 2018]. This detector, in its first implementation, starts from part of the architecture presented in DeeperCut [Insafutdinov et al., 2016], *i.e.*, a ResNet He et al., 2016 pretrained on Imagenet [Deng et al., 2009], and adds a deconvolutional layer at the end in order to retrieve probability maps (see Figure 3.9). In this context, it is possible to fine tune the architecture with few annotated frames and it can be adjusted to track the desired points of interest (transfer learning). To accomplish this fine tuning with ease, DLC provides routines and a Graphical User Interface (GUI) for extracting frames from videos and generate training data by manual labeling.

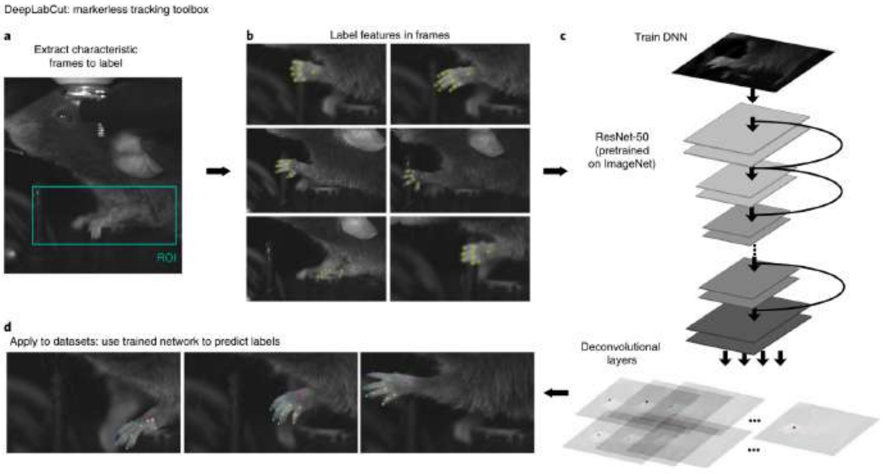

DeepLabCut: markerless tracking toolbox
a
Extract characteristic frames to label
ROI
b
Label features in frames
c
Train DNN
ResNet-50 (pretrained on ImageNet)
Deconvolutional layers
d
Apply to datasets: use trained network to predict labels

Figure 3.9: Example of DeepLabCut workflow (source: [Mathis et al., 2018]).

VISIONTOOL. To overcome the problem of the low number of tools that guide the users in the process of training and testing a semantic features detector and to have more control on the detection of semantic features in our application tasks, we implement VisionTool, an open-source python toolbox capable of providing accurate features detectors for different applications, including motion analysis. VisionTool leverages transfer-learning with a large variety of deep neural networks allowing the users to select the one that fit the particular problem under investigation. The toolbox offers a friendly Graphical User Interface (GUI), efficiently guiding the user through the entire process of features detection. VisionTool is presented in Chapter 8.

## 3.5 3D human pose estimation algorithms

The final aim of pose estimation from RGB videos is the study and the characterization of human motion/actions. Since humans interact in the 3D world, a possible approach to retrieve 3D human pose is to combine pose detections from different view points of the same person/people and geometrically reconstruct the 3D information leveraging stereo vision techniques [Hartley and Zisserman, 2004].

GEOMETRIC 3D RECONSTRUCTION. If we have multiple calibrated cameras viewing the same scene (*i.e.*, we know the intrinsic and extrinsic camera parameters that can be combined to form the cameras matrices), we can perform 2D pose estimation on the images from each view and, then, we can use the final estimated semantic 2D points in each image plane and compute the corresponding 3D points by triangulation [Hartley and Zisserman, 2004], see Figure 3.10.

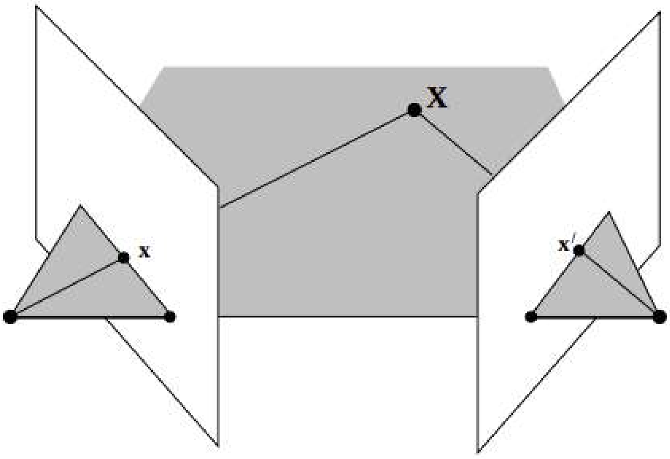


Figure 3.10: The points $x$ and $x'$ in the two image planes are projected in the 3D space ($X$) knowing the camera parameters (source: [Hartley and Zisserman, 2004]).

In this context, there are recent works that leverage triangulation increasing the accuracy of the 3D reconstruction by refining the detection on the image planes of the different viewpoints. For example, in the work by [Zhang et al., 2021], the authors propose Adafuse, a deep learning-based methods to refine the detection of the 2D estimates of the body keypoints retrieved with a pose estimation algorithm and they prove that with this refinement also the error after the 3D reconstruction is reduced. Similar works were presented by [Cheng et al., 2019; Pavllo et al., 2019; Qiu et al., 2019].

MONOCULAR 3D RECONSTRUCTION. Unfortunately, it is not always possible to have multiple view points of the same action. For this reason, recently, a lot of effort has been devoted to retrieve the 3D human pose starting from a single image [Pavlakos et al., 2017; Zhou et al., 2017, 2016] or from consecutive video frames [Tekin et al., 2016]. These methods are now largely adopted in the computer vision community, but, unfortunately, they do not fit the requirements of the application fields treated in this thesis (medical and rehabilitation). In fact, we need systems for the 3D reconstruction fully interpretable and as accurate as possible.

One of the earliest methods adopting an end-to-end CNN-based approach to retrieve the 3D pose directly from a single image is the work of [Li and Chan, 2014]. They present a two-level architecture that simultaneously detects 2D body landmarks and regresses 3D coordinates. In most cases, input to these architectures are RGB images and the GT used during training are the $(X, Y, Z)$ coordinates of each keypoint in camera coordinates or in normalized coordinates (thanks to large public available datasets such as HumanEva [Sigal, Balan, and Black, 2010] and Human3.6M [Ionescu et al., 2013]). Another possibility explored by [Martinez et al., 2017; Moreno-Noguer, 2017] is to input to the model the 2D keypoints locations in the image plane retrieved with 2D pose estimators. Unfortunately, 2D pose information alone is ambiguous. Thus, different approaches use a combination of RGB image and 2D pose as inputs [Mehta et al., 2017; Popa, Zanfir, and Sminchisescu, 2017; Rogez and Schmid, 2016; Rogez, Weinzaepfel, and Schmid, 2017; Tome, Russell, and Agapito, 2017]. One big challenge in 3D pose estimation is generalization capability.

For a more detailed review of 3D human pose estimation algorithms, the reader is referred to the following surveys [Bartol et al., 2020; Zheng et al., 2020].

## 3.6 Evaluation metrics

To properly evaluate the accuracy and the reliability of the detection provided by the deep learning-based pose estimators, numerous metrics have been introduced to compare the estimates $\tilde{x}_i$ (that we will exchangeably call estimate, keypoint or landmark point) with the GT positions $x_i$, with $i = 1, ..., N$ and $N$ the number of keypoints detected.

AVERAGE PRECISION (AP) AND AVERAGE RECALL (AR). The most common metrics to assess the performance of a semantic keypoint detection algorithm are Average Precision (AP) and Average Recall (AR). The explanations of these two metrics follow the ones provided in [*COCO 2020 Keypoint Detection Task*]. As a reminder, we quickly summarize the basics behind the terms precision and recall: they are statistical variables, which can be computed based on the entries of binary confusion matrices, see Table 3.1.

The formulas to compute precision and recall are [Davis and Goadrich, 2006]:

$$Precision = \frac{TP}{TP + FP} \tag{3.1}$$

| | | **Ground truth** | |
|---|---|---|---|
| | | *Positive* | *Negative* |
| **Prediction** | *Positive* | True Positive (TP) | False Positive (FP) |
| | *Negative* | False Negative (FN) | True Negative (TN) |

Table 3.1: General binary confusion matrix.

$$Recall = \frac{TP}{TP + FN} \tag{3.2}$$

In other words, precision is the fraction of true positives over the number of predicted positives, whereas recall is the fraction of true positives over the number of actual positives. In order to compute precision and recall for a keypoints detection problem, it is necessary to decide if a keypoint is correctly or wrongly detected. To do that, the Object Keypoint Similarity (OKS) between the estimate ($\tilde{x}_i$) and the GT ($x_i$) needs to be computed. If the OKS is greater than a specific threshold $\tau$ ($OKS > \tau$), then the keypoint estimation is considered correctly detected (TP), otherwise it is considered a wrong detection (FP).

$$OKS = \frac{\sum_i e^{\frac{-d_i^2}{2s^2k_i^2}} \delta(v_i > 0)}{\sum_i \delta(v_i > 0)} \tag{3.3}$$

with $i$ representing the index for the set of keypoints; $d_i = |\tilde{x}_i - x_i|$ is the Euclidean distance between a detected landmark point and the corresponding GT; $s$ is the object scale (*e.g.*, square root of object bounding box area in pixels); $k_i$ is a per-keypoint constant that controls falloff; $v_i$ is the visibility flag of GT ($v_i = 0$ means keypoint not labeled, $v_i = 1$ labeled but not visible, $v_i = 2$ labeled and visible). $v_i > 0$ in the equation means that the similarities are averaged only over labeled keypoints and the ones not labeled do not affect the OKS. The term $e^{\frac{-d_i^2}{2s^2k_i^2}}$ is identified as Keypoint Similarity (KS) and it is a Gaussian centered in the GT position $x_i$ and with standard deviation $sk_i$. OKS is a number in the interval $[0, 1]$: perfect predictions have $OKS = 1$ and predictions for which all the estimates of a certain keypoint are off by more than a few standard deviations $sk_i$ have $OKS \approx 0$. The OKS is the analogous of the Intersection over Union (IoU) [Rezatofighi et al., 2019] for object detection.

As mentioned, the variable $k_i$ aims to control the falloff of the Gaussian for each keypoint and it is often selected taking into account the variability of human performance annotating the same keypoint. More than one annotator is usually involved and the standard deviation of their annotations is computed ($\sigma_{i,human}$). Many studies consider $k_i = 2\sigma_{i,human}$.

Finally, a keypoint is considered as correctly detected if $OKS > \tau$. Common choices for $\tau$ are $\{0.50, 0.55, 0.60, ..., 0.95\}$. $OKS > 0.75$ is considered as a strict metric. Once we know if an estimate is correct or not, we can compute AP and AR. Usually, AP and AR are averaged over multiple OKS values.

Interestingly, as it is possible to see from Figure 3.11 (source: [Ruggero Ronchi and Perona, 2017]), the KS depends on the dimension of the keypoint detected (thanks to the role of the variable $s$): if we fix the distance $d_i$ a larger object is related with higher similarity.

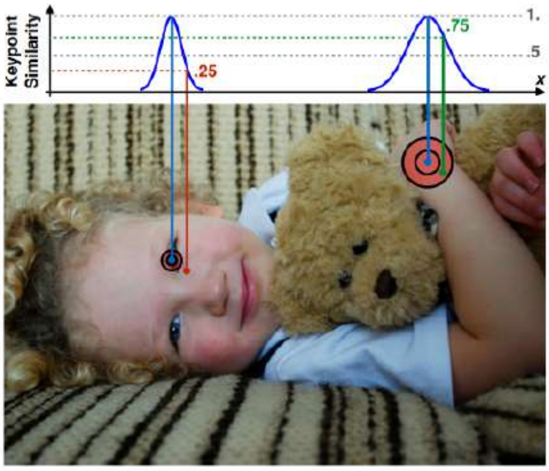


Figure 3.11: Keypoint Similarity for two different keypoints (eye and wrist). The blue points define the GT positions, the red (eye) and green (wrist) points define the estimates. The 2D orange Gaussians around the GT points have different standard deviations $sk_i$, consequently, two estimates that have the same distance from their GT can have different KS values (source: [Ruggero Ronchi and Perona, 2017]).

**percentage of correct parts (pcp).** The PCP [Eichner et al., 2012] is not related directly with one estimate's position but with the distance between two of them ($\tilde{x}_n$ and $\tilde{x}_m$, with $n$ and $m \in \{1, ..., N\}, n \neq m$). For this reason, PCP is interesting to evaluate pose estimation algorithms (in particular to evaluate the accuracy of each limb composing the skeleton) and not semantic features detectors. Specifically, a limb is considered correctly detected if the distance between the difference of the two estimates forming the limb ($\tilde{x}_n$ and $\tilde{x}_m$) and the difference between the true limb joints ($x_n$ and $x_m$) is less than a threshold, usually identified as a percentage $\alpha$ of the GT limb length (commonly 50%, meaning $\alpha = 50$, and consequently denoted as $PCP@0.5$):

$$||(\tilde{x}_n - \tilde{x}_m) - (x_n - x_m)||^2 < \frac{\alpha * ||x_n - x_m||^2}{100} \tag{3.4}$$

In the following we describe more straightforward metrics to understand. All of them have in common the logic behind the decision to consider an estimate $\tilde{x}_i$ correctly or wrongly detected: the definition of a circle of radius $r_{thr}$ around the GT keypoint $x_i$ (considered as the center). If the estimate falls inside the circle, it is considered as correctly detected:

$$|\tilde{x}_i - x_i| < r_{thr} \tag{3.5}$$

They only difference regards the choice of the threshold radius $r_{thr}$ of the circle around the GT.

**percentage of correct keypoints (pck).** The PCK [Yang and Ramanan, 2012] is computed considering $r_{thr}$ as a percentage of: (i) the torso diameter (usually the 20%, $PCK@0.2$); (ii) the head bone link (usually the 50%, $PCKh@0.5$ with $h$ indicating that we are referring to the head bone link).

**percentage of detected joints (pdj).** The PDJ [Toshev and Szegedy, 2014] is computed considering $r_{thr}$ a fraction of the torso diameter. For instance, $PDJ@0.2$ equal to the distance between predicted and true joint $< 0.2$ * torso diameter.

**mean per joint position error (mpjpe).** Until now we have reported metrics for the evaluation deep learning algorithms for 2D pose estimation, the ones we are more interested in. The metrics for evaluating 3D pose estimators are fewer and much simpler. This is due to the fact that reliable deep learning algorithms for 3D pose estimation have been proposed and studied only in the last few years. Hence, not as many metrics have been established over time. The most common one is the MPJPE, defined for each keypoint as the mean Euclidean distance in the 3D space (in mm) between the estimated keypoint ($\tilde{X}_i$) and the correspondent GT ($X_i$).

# 4

# Proposed Approach to Markerless Human Motion Analysis

In this chapter, we report all the steps of the pipeline we propose to perform markerless human motion analysis starting from RGB video acquisitions.

## 4.1 Introduction

Our aim is to create a markerless system able to describe in a quantitative way human motion. We test our pipeline also in the medical and the rehabilitation domains to support physicians in their medical evaluations and to allow them to timely plan appropriate rehabilitation treatments. In this context, an important requirement to take into account is the need for interpretable methods: for this reason, we can not rely on end-to-end architectures and we need to build a multi-level system to better understand and control the results of each step. In this way it is also possible to use this system in different application scenarios with just small changes. Specifically, we divide the pipeline into the following steps.

1. *Landmark points detection* (Section 4.2): we leverage CNN-based approaches and we detect the positions of meaningful landmark points in the image plane. The keypoints we detect are selected according to the problem we address.

2. *Landmark points' trajectories filtering* (Section 4.3): since we are interested in the characterization of human actions, we analyze videos (and not just single images). For this reason, at the end of step 1, we have the keypoints' detections for each frame composing the video. Thus, we filter the trajectories in order to improve the spatio-temporal consistency.

3. *3D reconstruction* (Section 4.4): if the acquisition setup includes multiple viewpoints, the information from the different views is combined in order to reconstruct the 3D positions $(X, Y, Z)$ of each landmark point. This is particularly useful because the standard marker-based approaches to study human motion provide 3D information.

4. *Motion characterization* (Section 4.5): depending on the problem we address and on which aspect of the motion we characterize, the processing step can include (i) the extraction of quantitative motion parameters, (ii) the classification / characterization of motion patterns.

## 4.2 Landmark points detection

We start from the same logic behind gold standard marker-based systems and we detect the position of meaningful landmark points in the videos of humans. To do that, it is first necessary to locate them in 2D image planes. Depending on the application, we start by relying on: (i) pose estimation algorithms [Bazarevsky et al., 2020; Cao et al., 2017]; (ii) pre-trained semantic features detectors [Mathis et al., 2018] fine tuned on specific keypoints depending on the motion patterns to characterize. Since we need full control on the overall procedure, within this project, we also implement our semantic feature detector: VisionTool (see Chapter 8 for a complete overview).

Inputs to our procedure are frames extracted from RGB videos. Independently from the features detector adopted, the outputs are semantic probability maps related to each keypoint and the positions of the keypoints themselves in the image plane (the location of the pixels with the highest value in the probability maps – see Figure 4.1 for examples). Specifically, for each video we have $\{(x, y, c)_l^t\}_{t=0}^T$, where $l$ is the index for the different keypoints (that depend on the application scenario) and $t$ is the index indicating the frame ($T$ represents the total number of frames in one video). In particular, $(x, y)_l^t$ is the position of the $l-$th point in the $t-$th frame and $c_l^t$ is a number in the interval $[0, 1]$ and represents the corresponding likelihood (confidence map value in the position $(x, y)$). With $c_l^t$ we are able to quantify the uncertainty behind the detection of each point in each frame.

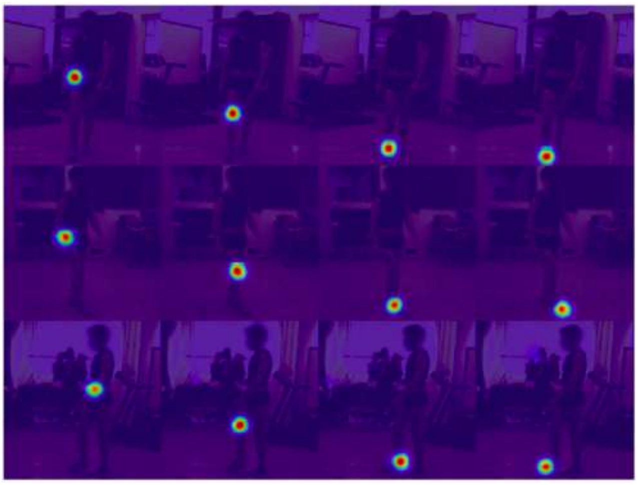


Figure 4.1: Example of probability maps for different body landmarks. Red and blue correspond to high and low probability respectively.

## 4.3 Landmark points filtering

To improve the stability across time of the estimated points and reduce localization errors, we add a temporal processing. This is possible because we are working with videos and not with single unrelated images and we can rely on temporal consistency. This step is necessary in order to: (i) correct the **mispredictions** of the semantic features detector/pose estimator, which occasionally detects points in a wrong position; (ii) manage **occlusions**.

Errors due to mispredictions are easily recognizable because they involve a characteristic spike in the trajectory of the point's coordinates. Commonly, it is possible to overcome this problem with two different approaches:

- A median filter applied to the time sequences of the individual positions, $\{x_t^l\}_{t=0}^T$ and $\{y_t^l\}_{t=0}^T$ respectively. The filter replaces each point $p_t^l$ of the trajectory, with the median value computed on a neighbourhood of $p_t^l$ of size $F$ (an odd value, usually $F = 5$ in our studies). The median filter discards outlier values, corresponding to the above-mentioned spikes.

- The detection of evident peaks in the speed profile of the coordinates themselves.

In the case of occlusions, the neural network will find it hard to identify the position of the occluded point as it is hidden; this situation is easy to identify as the detection likelihood $\{c_t^l\}_{t=0}^T$ of the occluded points $l$ at a fixed time instant $t$ drops to values close to zero. A simple idea would be to discard from the motion study the points that are detected with a small likelihood, but this approach could drastically reduce the amount of useful information. To overcome this problem and control information loss, we drop the points with the likelihood below a threshold ($thr_{lik}$, usually we set $thr_{lik} = 0.75$) and then, we interpolate the trajectories in order to reconstruct the movement of each point in the temporal interval between their occlusion and their reappearance. In this way, if an anatomical point of interest is temporarily occluded while it is moving, it is possible anyway to reconstruct its movement. With the interpolation we also smooth the signal and solve small localization errors.

Finally, in order to smooth the resulting signals, we apply a low pass filter (*e.g.*, Butterworth 4th order $f_c$ cut off frequency) to each time-series corresponding to the $x$ and $y$ coordinates of the keypoints. Since we work with human motion, the cut off frequency $f_c$ commonly adopted is $12Hz$ [Whittle, 2014].

## 4.4 Landmark points 3D reconstruction

As mentioned in Chapter 3, in the literature there are algorithms for 3D reconstruction specific for the human pose. There are methods relying on single images (*e.g.*, [Tome, Russell, and Agapito, 2017]); others that work under the hypothesis that calibration is not available, and generally require very large datasets (*e.g.*, [Burenius, Sullivan, and Carlsson, 2013; Elhayek et al., 2016]). Other works approach the problem using a similar prior as calibrated reconstruction (multi-view calibrated inputs during training) but in a data-driven fashion (*e.g.*, [Kocabas, Karagoz, and Akbas, 2019]). Unfortunately, the previously mentioned approaches do not appear to be appropriate for our purpose. In fact, we need a method as precise and accurate as possible. For this reason we opt for a geometric approach relying on stereo vision [Hartley and Zisserman, 2004]. The choice of a general purpose geometric approach is also motivated by its simplicity and high generalization potential. For these reasons, we included in our analysis the 3D reconstruction step only when a multi-view camera system (two or more cameras) is available.

GEOMETRIC APPROACH. The semantic features extracted $\{(x,y)_t^l\}_v$ from different viewpoints $v$ (where $v = 1, ..., V$ is the index for the different viewpoints and $V$ is the total number of viewpoints available) in each time instant $t$ are combined to compute their corresponding points in the 3D space $(X,Y,Z)_t^l$ by means of multi-view geometric reconstruction. In order to perform geometric 3D reconstruction, it is necessary to perform camera calibration. Among the possible calibration techniques, we rely on Zhang calibration [Zhang, 2000]. Thus, we estimate the intrinsic matrices $K_n$ of each camera and the extrinsic parameters between camera pairs [Zhang, 2000]: $(R_{ij}, t_{ij}),\ i,j = 1, ..., V \quad i \neq j$. Then, if more than two viewpoints are acquired, in order to register them and to have the same reference system, we apply rotation averaging [Hartley et al., 2013]. This technique takes the relative rotations $R_{ij}$ and computes the absolute rotations $R_i$ in order to satisfy the compatibility constraint

$$R_{ij} * R_i = R_j. \tag{4.1}$$

In the presence of noise the problem can be solved through the minimization of:

$$\min_{R_1,...,R_V} \sum_{(i,j)} ||R_{ij} - R_i * R_j^T||^2. \tag{4.2}$$

If the first view is chosen as reference, we have that $R_1 = I$. Similarly, it is possible to synchronize the translation vectors obtaining the absolute translations $t_i$ starting from the $t_{ij}$ and satisfying the compatibility constraint

$$t_{ij} = t_i - R_{ij} * t_j. \tag{4.3}$$

Once rotations and translations are synchronized, considering $\tilde{p}_i^V$ the 2D landmarks expressed in $mm$ ($\tilde{p}_i^V = K_V p_i^V$), we apply a linear triangulation algorithm followed by a non-linear refinement based on the Gauss-Newton method [Hartley and Zisserman, 2004], obtaining $P_i$ in the 3D space.

ADAFUSE. To reduce the impact of occlusions and, consequently, to reduce the 3D keypoints localisation error, it is possible to add a step before the 3D reconstruction consisting in the refinements of the probability maps of the 2D detections in the image planes. To do that, we rely on AdaFuse [Zhang et al., 2021], a deep learning-based algorithm that is mainly divided into the three following parts:

- A 2D pose estimator backbone.
- A fusing deep learning architecture (the main innovative contribution of Adafuse) that refines the probability maps of each view generated in the first step. To accomplish this, the algorithm takes into account the information from neighboring views and it leverages epipolar geometry [Andrew, 2001]. In this way, it is possible to augment the information of each probability map at any point $x$ by adding the information of the probability maps of its neighboring viewpoints.
- A geometric 3D reconstruction part as described above.

In the work presented in the following Chapters, we adopt the methods that best fit the requirements of the application task we addressed.

## 4.5 Motion characterization

Once we obtain the trajectories of each keypoint (2D or 3D depending on the number of available view-points), we analyze them in order to extract motion parameters or characterize different motion patterns. In this section we report general purpose methods. More specific approaches will be described in Part II.

### 4.5.1 *Quantitative parameters extraction*

Starting from the filtered 2D or 3D signals that represent the evolution over time of the positions of each landmark, it is possible to extract quantitative parameters that characterize the motion patterns.

Given a dataset, oftentimes videos can have different duration and they can be also quite long. For these reasons we usually split them in temporal windows of length $W$, and we compute the motion features for each time window composing each video. In this way it is possible to capture different characteristics of the motion depending on the temporal scale $W$. In order to lose as little information as possible and to increase the number of time windows for each video, we may also consider a stride $S$ ($S \leq W$) between the starting point of a time window and the starting point of the consecutive one (see Figure 4.2).

Depending on the application domain, the parameters extracted to characterize the motion are different. However, we identify certain motor parameters that describe general aspects of the human motion [Ahmedt-Aristizabal et al., 2019] and that can be divided in (i) kinematic parameters and (ii) parameters in the frequency domain (see Table 4.1 for a complete list).

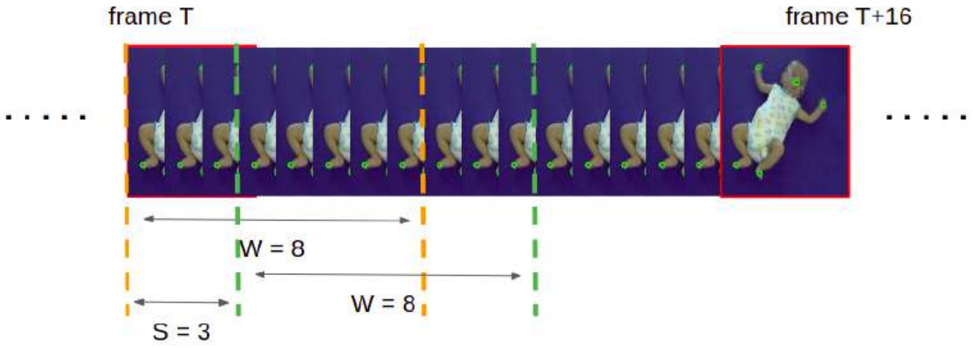


Figure 4.2: The role of $W$ and $S$, an example: we show 16 frames of a video and we highlight two time windows of $W = 8$ frames and a stride $S = 3$ between the two windows. The green circles in each frame are the detected landmarks points.

### 4.5.2 *Classification*

The quantitative parameters extracted can be used to characterize and to distinguish different motion patterns. In the medical domain, this procedure is commonly adopted to predict between normal and abnormal behaviours. In the work presented in the second part of the thesis, we focus on the detection of normal and abnormal motion patterns and we consider mainly 5 different well-established classifiers:

- Support Vector Machine with polynomial kernel (**SVM-poly**) [Vapnik, 2013];
- Support Vector Machine with Gaussian kernel (**SVM-rbf**) [Vapnik, 2013];
- Random Forest (**RF**) [Breiman, 2001];
- Fully connected neural networks (**NN**);
- Architectures based on Long Short Term Memory (**LSTM**) [Hochreiter and Schmidhuber, 1997].

A common practice in the medical domain – where the amount of data available is small – is to train and to test binary classifiers with a $k$-fold cross-validation (usually $k = 5$ in our work). In our proposed procedure, the parameters are computed for each time windows of length $W$. This means that during test, for a given video $j$, we obtain one prediction for each $i$-th time window (*i.e.*, for a binary classification problem we obtain $Pred_j^i = 0$ and $Pred_j^i = 1$ respectively for the two classes, in our work usually representing the absence or the presence of abnormal motion patterns). Since we are usually interested in one outcome for each video, the predictions of the time windows are then grouped (following the steps explained below in Subsection 5.4.1) obtaining the final prediction $Pred_j$ (one for each infant).

| | Signal | Parameter |
|---|---|---|
| Kinematic parameters | position | total covered distance |
| | | path length |
| | speed | mean |
| | | standard deviation |
| | | median |
| | | maximum |
| | | minimum |
| | acceleration | mean |
| | | standard deviation |
| | | median |
| | | maximum |
| | | minimum |
| | jerk | mean |
| | | standard deviation |
| | | median |
| | | maximum |
| | | minimum |
| Frequency parameters | total covered distance spectral density | entropy |
| | | peak magnitude |
| | | sum of the spectrum |
| | | spectral half point |
| | speed spectral density | entropy |
| | | peak magnitude |
| | | sum of the spectrum |
| | | spectral half point |

Table 4.1: List of general motion parameters (last column). They are usually computed for human keypoints / body parts. The *total covered distance* is the sum of the Euclidean distances of the position of a same point / body part in consecutive frames; the *path length* is the Euclidean distance between the starting and the final position of each landmark point / body part.

### 4.5.3 *Interpretable tools*

As we reason on the design of methods to assist medical diagnosis, an important requirement is the interpretability of the results. In this case, this means to highlight which motor parameters are more relevant in the classification step. Among the available tools to provide interpretability for a classifier, we focus on two different approaches:

1. *Feature importance based on permutation*, providing a score for each input feature and representing how much a classifier bases its decision on that specific attribute. Among the different approaches to identify sets of important features, we consider a permutation-based importance algorithm

[Breiman, 2001]. This method randomly shuffles each input parameter and compute the change in the model's performance during test. The meaningful parameters are those impacting the performance the most.

2. *Rule extraction*, an algorithm specifically designed to support interpretability of neural networks [Augasta and Kathirvalavakumar, 2012].

RXREN. Neural network models are inherently a black box, since it is not trivial to determine the exact reconstruction of the set of operations and input values that cause the network to classify an input sample into a specific output class. Such black box nature may represent a significant drawback when dealing with medical diagnosis. A possible step towards interpretability consists in associating an inherently explainable white box model to the *black box* neural network by extracting classification rules. In our work, we adopt a customized version of RxREN [Augasta and Kathirvalavakumar, 2012], a pedagogical rule extraction algorithm providing: (i) a subset of significant input parameters that are more important for the neural network's classification; (ii) ranges of significant input features (in form of minimum-maximum value) causing the neural network to assign an input sample to each of the output classes; (iii) a transparent and explainable set of rules that can be used to actually classify the input data instead of the original neural network architecture. Figure 4.3 shows a schematic representation of RxREN's workflow.

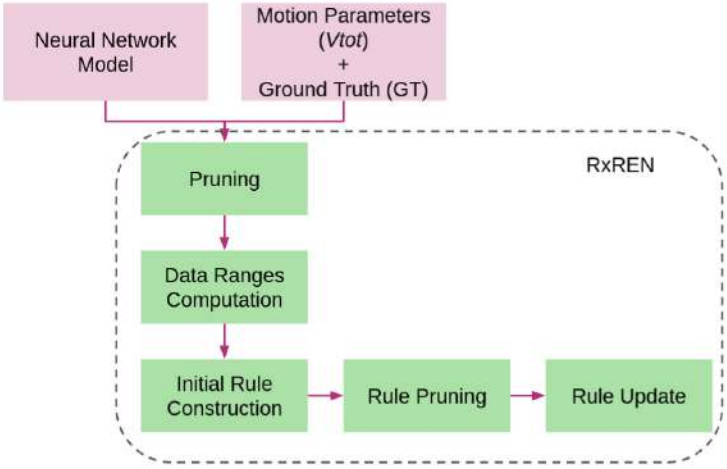


Figure 4.3: RxREN's algorithm workflow.

We report below the main steps of the customized version of RxREN:

1. **Pruning.** For each input neuron $I_i$, find the number of incorrectly classified testing samples $Err_i$, corresponding to the neural network prediction after removing the neuron $I_i$ from the input set. The neurons associated with the minimum error are considered insignificant and removed from the input. The procedure is iterated until the network accuracy at iteration *it* ($Pacc_{it}$) surpasses the original testing accuracy ($Acc$): $Pacc_{it} > Acc$.

2. **Data range computation.** After the subset of significant input neurons (*i.e.*, input features) is obtained, the next step consists in identifying the data range for both the correctly classified and the misclassified samples for each of the significant neurons $I_i$. The result is a data length matrix, where each element ($mc_{ic}$) is associated with the $i$-th neuron and $c$-th class. A threshold $\alpha$ is defined to only consider input features significantly important to classify a certain class $c$. The corresponding data range matrix is

$$DM_{ic} = \begin{cases} L_{ic} - U_{ic} & \text{if} \quad mc_{ic} > \alpha * mp_i \\ 0 & \text{otherwise} \end{cases} \tag{4.4}$$

   with $mp_i$ being the average among all the classes for neuron $i$, and $L_{ic}$ and $U_{ic}$ being the minimum and the maximum values of correctly and misclassified samples in output class $c$ for neuron $i$ .

3. **Rules definition.** The method defines rules for each target class $c_p$ by using the nonzero data ranges available in the corresponding column $p$ of the data range matrix. One rule is defined for each output class, representing the subset of significant input features and corresponding ranges that cause the original neural network to predict an input sample as belonging to that specific output class. Each of these ranges represents a condition in the rule. The rules are extracted in descending order according to the number of required input attributes.

4. **Rules pruning.** Pruning can be applied to the constructed set of rules to further improve the classification accuracy. Let $Racc$ be the accuracy of the initial set of rules on the test set. Iteratively, starting from the more restrictive rule (*i.e.*, the rule with more conditions), each condition $cn_{it}$ is removed (*i.e.*, an input attribute data range is removed), and on the remaining ones testing accuracy is estimated, let it be $Rnewacc$. The condition $cn_{it}$ is considered redundant and removed, if $Rnewacc \geq Racc$.

5. **Rules update.** Finally, the data range is updated considering the data range matrix for the correctly classified and misclassified samples obtained by using the constructed set of rules. In descending order, similarly to pruning, each condition $cn_{it}$ is updated in terms of data ranges, and the updated set is used to perform classification, obtaining an accuracy equal to $Rupacc$. The update is considered significant and maintained if $Rupacc \geq Raccpruned$, where $Raccpruned$ is the classification accuracy obtained using the pruned set of rules at previous step.

The final set of rules may be used as an explanation of the original classifier or as a new rule-based classifier, that approximates the behavior of the original model.

### 4.5.4 *Graph-based analysis*

A new approach to study and characterize human motion has been implemented in the field of action recognition and classification. In particular, the 2D (or 3D) human keypoints are considered as nodes of the graph-based structure and analyzed trough Graph Neural Networks (GNN) Scarselli et al., 2008.

In our work, to highlight different characteristics of the motion not necessarily perceptible with quantitative motion parameters, we implement a procedure based on the analysis of graphs. This idea and its corresponding implementation particularly fit the analysis of infants' motion (see Chapter 5), but it can be extended also to different application domains. In the following, we present the main steps of the procedure.

NETWORKS DEFINITION The first step is the definition of the graph structures. We start from the detection of landmark points in the image plane, as described in Section 4.2. Considering one video, we build a temporal sequence of networks (one per frame) describing the relation among the landmark points of interest in the image plane, used as nodes, that are connected through edges depending on their relative proximity. More specifically, we obtain the edges by computing the Euclidean distance between every pair of landmark points in all the images composing our dataset. The distances distributions are adopted to identify which points are close to or far from each other in each frame. Starting from the distances distributions, the selection of the threshold to consider two nodes close one to another depends on the application domain and on the aspects of the motion we are interested in.

In this way, each layer in a network represents a configuration at a specific timepoint (*e.g.*, at a specific frame) and it is defined as a $L$-nodes graph, which we exchangeably call *attributed graphette* or *configuration*, where $L$ is the number of keypoints detected.

ATTRIBUTED GRAPHETTES-BASED REPRESENTATION Each temporal network can be represented as a sequence of configurations. More formally, the $t$-th layer, corresponding to the $t$-th frame of a temporal network $G$, is represented by a graph $g_t = (V, E_t, S)$, where $V = \{1, ..., L\}$ is the set of nodes, $E_t$ is the set of edges and $S$ represents the set of node attributes. It is important to note that at each timepoint $t$, the map assigning a node $n \in V$ to a label $l \in S$ is a bijection. These configurations representing each frame are called *graphettes* [Hasan, Chung, and Hayes, 2017], defined as not necessarily connected, non-isomorphic induced subgraphs of a larger graph (see Figure 4.4 for an example). Similarly to graphlets [Pržulj, Corneil, and Jurisica, 2004] and motifs [Milo et al., 2002], graphettes are a suitable tool to give a local and global description of large complex networks. Indeed, by computing node-level graphettes concentrations in a network we are able to describe local wiring patterns [Tu et al., 2019] and, at the same time, by aggregating this local information, we get a global description of the network based on the occurrences of these substructures [Pržulj, Corneil, and Jurisica, 2004]. The number of possible configurations depends on the number of keypoints $L$.

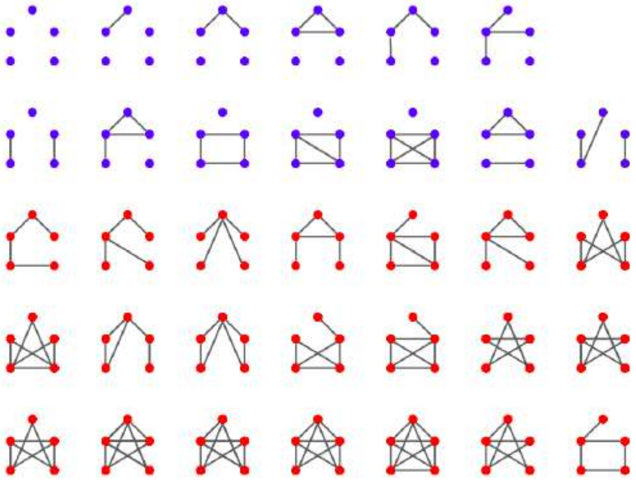


Figure 4.4: Example of 5-nodes ($L = 5$) graphettes (blue and red) and graphlets (red).

To have a local and global description of graphettes distributions, we leverage Natural Language Processing (NLP) techniques. To do that, we need to associate each attributed graphettes to a *word*, *i.e.*, a string composed by the letter $g$ and an integer number obtained by concatenating the elements of the adjacency matrix and then interpreting that as a binary number (convention we randomly decided to adopt for simplicity). A practical example of this process is shown in Figure 4.5.

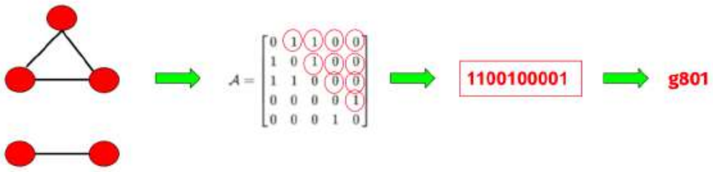


Figure 4.5: Canonical representation of one instance of a 5-node attributed graphette. Starting from an attributed graphettes and its related adjacency matrices ($A$) we generate a word composed by the letter $g$ and an integer number obtained by concatenating the elements of the adjacency matrix and then interpreting that as a binary number.

Given an infant video $G$, we sequentially associate each frame $t$ of the video with an attributed graphette, represented by the corresponding string $g_n$. For instance, the attributed graphette 0000000000 (the first in Figure 4.4) corresponds to the string $g_0$. A network is then defined as an ordered sequence of words, whose length is equal to the number of frames of the corresponding video. Indeed, $G$ results as a collection of configuration names that we treat as *text*, resorting to NLP methods for text representation in order to enumerate attributed graphettes and describe their occurrences. From a practical point of view, in this way we are studying and characterizing the evolution of human keypoints (and, consequently, of human poses) across time.

**NLP METHODS** In this regard, the Bag-Of-Words (BOW) [Goldberg, 2017] model is a histogram representation that transforms any text into fixed-length vectors by counting how many times each word appears in the document. This vectorization process is performed by fixing or inferring a *vocabulary*, which is contained in or equal to the set of all words found in the documents. In our case, the vocabulary of all configurations appearing in the dataset consists of all the possible $N_{tot}$ attributed graphettes found in the dataset. Therefore, after fitting a BOW model, every network turns out to be a vector of size $1 \times N_{tot}$. Figure 4.6 (left panel) offers a visual representation of a video as a BOW vector.

In order to identify those configurations that are discriminative for networks in the dataset, we need to normalize raw counts in BOW vectors properly. For this purpose, we leverage Term Frequency - Inverse Document Frequency (tf-idf), a common algorithm to transform word counts into meaningful real numbers [Salton and Harman, 2003]. More specifically, given a configuration $g_n$ and a network $G$, tf-idf measures the originality of $g_n$ by comparing the number of times $g_n$ appears in $G$ (*i.e., term frequency*) to the number of networks $g_n$ appears in (*i.e., document frequency*). Formally,

$$\text{tf-idf}(g_n, G) = \text{tf}(g_n, G) \times \left( \log \frac{1+N}{1+\text{df}(g_n)} + 1 \right) \tag{4.5}$$

where $\text{tf}(g_n, G)$ (*i.e.*, term frequency) is the component of $G$'s BOW representation corresponding to $g_n$, $N$ is the number of networks in the dataset and $\text{df}(g_n)$ (*i.e.*, document frequency) is the number of networks $g_n$ appears in. To reduce the dimensionality of these representations, we set a threshold on the minimum and maximum document frequency of configurations. In the tf-idf case, we also retain the ability of weighting graphettes based on their commonality in the dataset. Figure 4.6 (right panel) illustrates a tf-idf transform of the network in the left panel.

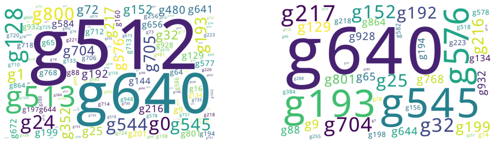


Figure 4.6: BOW (left) and tf-idf (right) *word cloud* visualization of an infant's temporal network. The size of configuration names is proportional to their weights in the corresponding representation. Note that the configuration $g512$ is either very frequent or rare in the collection of infants networks and therefore it has weight equal to 0 in the tf-idf representation.

Even if the tf-idf approach provides an arbitrary amount of reduction in description length, it does not reveal any information on intra-networks distribution over all attributed graphettes. To overcome this limitation, we resort to topic modeling [Alghamdi and Alfalqi, 2015] to define an interpretable low-dimensional representation of videos, able to describe the distribution of attributed graphettes for each video. Topic models [Alghamdi and Alfalqi, 2015] are probabilistic generative models for large collections of textual data (*i.e.*, text corpora). A notable topic model is Latent Dirichlet Allocation (LDA) [Blei, Ng, and Jordan, 2003] defined as a 3-level hierarchical Bayesian model, in which every item in a corpus is modelled as a mixture over an underlying set of topics, which are, in turn, described by a probability distribution over words. Topic probabilities offer an explicit low-dimensional representation of texts which has been recently adopted to analyse large social networks [Long et al., 2020].

As mentioned before, we adopt this procedure to characterize the evolution across time of body configurations in the study of infants' motion (Chapter 5)

## 4.6 Strengths and weaknesses of our markerless approach

First of all, we highlight the main benefits that markerless systems allow to reach with respect to marker-based ones. Here we highlight the main advantages of markerless methods for human motion analysis.

- In general, the overall process to perform markerless analysis requires less preparation and it is not operator dependent. While the practitioner during marker-based data acquisition needs to place markers very carefully on the body skin in order to reduce as much as possible the bias, markerless systems are fully automatic, and they are independent of any human performance.

- The naturalness of movements is not affected by cumbersome markers placed all over the body skin. Thanks to the non-invasive nature of markerless systems, it is possible to adopt them to study human motion in application fields where the invasiveness of markers do not allow or reduce the possibilities to perform analysis (such as with infants, music players and athletes).

- It is less expensive. We estimated that the cost for markerless analysis is almost 20 times lower with respect to marker-based systems.

On the other side, they present some limitations that should be further explored.

- If from one side there are few and well defined marker-based system for human motion analysis, on the other side there are many different RGB cameras with different resolutions, sensors type and frame rates. For these reasons and for the variability of the application environment, it is difficult to create a standard reproducible system based on RGB cameras.

- Due to intrinsic physical limitation of camera sensors (spatial limitation related to pixel size), the level of detail achievable with this type of analysis is limited.

- A too low temporal resolution (frame rate) of the camera combined with fast movements may lead to motion blur and, consequently, to small errors in the detection of the keypoints in the image plane. One immediate way to reduce the motion blur is to adopt RGB cameras with a high temporal resolution (*i.e.*, high acquisition rate).
- The proposed pipeline, at this stage, requires only one person in the scene and it has been tested in controlled environments.

In conclusion, markerless analysis represent an interesting alternative to standard marker-based techniques. Unfortunately, its application has not been fully explored and the studies analyzing the differences with well defined gold standard analysis are still limited.

PART II

# Applications

This part of the document contains all the applications of our markerless pipeline to characterize human motion. With these examples we show the potential of our algorithm in terms of accuracy and versatility with respect to gold standard marker-based systems.

# 5

# Infants Motion Analysis

In this chapter we present the application of our proposed markerless approach to study the motion of preterm infants. This work is carried out in collaboration with the Istituto Giannina Gaslini (Genova, Italy).

## 5.1 Introduction

The analysis of infants' spontaneous movements is essential for an early diagnosis of neuro-motor disorders, especially for preterm birth. In fact, the 5-15% of the premature babies born with a birth weight of less than 1500g have motor alterations and 25-50% of them have developed behavioural and/or learning deficits [Bax et al., 2005]. Thus, they can face a lifetime of disability [Allen, 2008]. Moreover, thanks to the development of intensive care techniques, in the last years, preterm survival rates have increased in high-income countries.

Common neurological disorders that could occur in the early stage of life are grouped under the term of 'Cerebral Palsy' (CP): permanent neuronal disorders due to lesions that primarily involve the areas of the brain intended for the control of movement and posture. Therefore, infants with CP may present problems in motor skills, muscle weakness, rigidity, slowness and difficulty in balance and coordination [Beckung and Hagberg, 2002]. An early diagnosis of pathological cases would allow the start of early rehabilitation treatment that could significantly increase the likelihood and extent of recovery.

Starting in the 1990s, the study of infants motion provide evidences of a qualitative correspondence between anomalies in the motion patterns and neurological dysfunctions [Bos et al., 1997]. This is the starting point for the development of Prechtl's General Movement Assessment (GMA) [Prechtl, 1997]. General Movements (GMs) are spontaneous movements of variable amplitude and speed involving different parts of the body that could reflect the state of neuro-motor development [Prechtl, 1990; Prechtl, 1997]. Common neurological evaluation to better understand and estimate the infants' neurological status include traditional neurological examination [Palmer, 2004] and neurological examination based on the observation of spontaneous motor behavior. The latter can include also the GMA [Prechtl, 1990]. Unfortunately, the visual analysis and the recognition of GMs and of abnormal movements involves highly specialized personnel for a long period of time, it is often operator dependent [Adde et al., 2009] and its reliability increases starting from 12 months after the conception, making difficult an early diagnosis. For these reasons, there is a need for objective computer-aided methodologies able to extract quantitative parameters that characterize infants' motion pattern.

In this context, we address the problem of the **automatic** markerless analysis and characterization of infants' spontaneous movements. An important requirement to take into account is the need for interpretable methods. With this thesis we present two different interpretable approaches for the characterization of infants motion relying only on a single RGB camera (see Figure 5.1). In both of them, our goal is to identify early signs of neurological disorders. This is essential because, despite the GMA, a fully reliable clinical evaluation of infants' neuro-motor status is performed starting from 24 months after birth. We study the problem with two different approaches because, with each of them, we could highlight different characteristics about infants motion patterns.

The long term goal of our research is to provide an easy to use methodology which could also be applied at home by care givers. For this reason we focus on single RGB videos which can be easily acquired by a mobile phone.

***Approach 1: parameters-based***. In this first study, we design a pipeline organized in three independent steps: (i) *video representation* based on 2D landmark points detection, (ii) *motion parameters extraction* and (iii) *classification*. Firstly, we adopt a semantic feature detector [Mathis et al., 2018] to automatically detect the positions of relevant landmark points (nose, hands and feet) in the image plane and we filter them to add spatio-temporal consistency. Then, we compute quantitative parameters inspired by the neuro-motor literature to describe infant's motion [Ahmedt-Aristizabal et al., 2019; Meinecke et al., 2006]. Lastly, to better understand the discriminative power of our computed parameters, we train and test different binary classifiers in order to identify infants likely to manifest neuro-motor disorders. To increase the interpretability of our analysis, this stage also includes a features importance procedure, to highlight the most meaningful parameters among the ones computed [Augasta and Kathirvalavakumar, 2012]. For the classification task we consider different alternatives: Support Vector Machine (SVM) [Vapnik, 2013], Random Forest (RF) [Breiman, 2001], a fully connected Neural Network (NN) and a deep learning architecture based on Long Short Term Memory (LSTM) [Gers, Schmidhuber, and Cummins, 2000]. The output of our pipeline is an automatic data-driven evaluation for infants' motion (machine learning (ML) evaluation in Figure 5.4), that can be adopted to support the clinical evaluation performed in the first weeks and/or months of infants life, when it is difficult to detect signs of abnormal motion patterns. In this way it is possible to increase the reliability of an early detection of infants at risk of neuro-motor disability and, consequently, timely plan an intervention, increasing the potential for recovery.

***Approach 2: graph-based***. In this second case, we approach the problem of representing spontaneous movement of preterm infants by studying it as a *temporal network analysis problem*. More precisely, we map each frame of a video to a 5-nodes graph whose nodes are detected landmark points (nose, hands and feet) and edges are inserted based on the Euclidean distance of the landmark points on the image plane. In this way, each video become a temporal series of graphs. We model the networks as sequences of 5-nodes *attributed graphettes* [Hasan, Chung, and Hayes, 2017], defined as not necessarily connected, non-isomorphic induced subgraphs of a larger graph, whose nodes are equipped with attributes. We exploit this modelling choice in order to obtain an interpretable, low-dimensional representation of each video, able to convey information about the local dynamics of each infant. In this sense, in [Long et al., 2020] the authors present a work in which they define a representation of a large social network by using methods of topic modelling [Alghamdi and Alfalqi, 2015]. In our problem, we leverage the method described in [Long et al., 2020], instantiated with a Latent Dirichlet Allocation [Blei, Ng, and Jordan, 2003] model, to identify local motion patterns able to characterize infants spontaneous movements. This method allows us to highlight insightful differences between the classes of infants with normal and abnormal motion patterns. As far as we know, this is an original approach, never used before.

## 5.2 Related work

We report here significant papers in the field of the analysis of 2D markerless infants' motion patterns. Since our work is based on single RGB videos, we review papers that adopt the same input data and we do not consider works that: (1) analyze infants' motion starting from signals acquired with wearable devices or different sensors (*e.g.*, depth cameras); (2) are based on motion capture systems and infrared cameras and markers; (3) exploit 3D information. On these approaches the reader is referred to [Burger and Louw, 2009; Hesse et al., 2018]. The remaining works in the literature dealing with this problem are not many.

One of the first video-based approaches for infants' motion analysis is introduced by [Adde et al., 2010, 2009]. The authors extract information about infants' motion relying on change detection: they extract the difference image between two consecutive frames and compute quantitative parameters based on pixels change between frames. The same method is adopted also by [Tacchino et al., 2021] and by [Tsuji et al., 2020]. [Baccinelli et al., 2020] provide a graphical user interface in order to help tracking the movement of hands and feet. This work also provides a software for motion parameters extraction. [Das, Fry, and Howard, 2018] focus specifically on the analysis of kicking movements for determining neuro-motor risk-level. Another computer vision technique adopted in this field is optical flow: [Stahl et al., 2012] compute motion parameters by tracking the body parts with a method based on optical flow. With this method they are able to underline specific motor patterns presented in the GMs theory and to classify with high accuracy infants with and without neuro-motor disorders. [Rahmati et al., 2014, 2015] propose an approach to segment and track the infants' body using optical flow fields initialized with a manual labeling, then from this information they compute motion parameters.

Recently, a lot of effort has been put into the study of algorithms for human pose estimation from RGB images (see [Colyer et al., 2018] for a review). One of them (*i.e.*, Openpose [Cao et al., 2017]) has been adapted in two different works - one by [Reich et al., 2021] and the other by [Chambers et al., 2020] - to study GMs directly from the infants' pose.

We conclude by observing that with respect to the papers cited in the current section, our approaches are fully data driven and require limited user intervention.

## 5.3 Dataset

Data acquisition was performed in collaboration with the Giannina Gaslini Hospital in Genova and it included different acquisition sessions and video and clinical evaluations.

ACQUISITION SESSION 1: 40TH WEEK OF GESTATIONAL AGE. During the first acquisition session, the spontaneous movements of 142 preterm infants, 90 females, born at (mean ± standard deviation) 29 ± 2 weeks and with weight at birth of 1212 ± 307g, were acquired at 40 weeks of gestational age. For inclusion in the study, all infants had to be preterm, in stable clinical conditions, in absence of pharmacological sedative treatment or respiratory support in the previous 4 weeks, with birth not beyond the 32th gestational week and/or weight at birth less than 1700g. The acquisition setup was composed by a single RGB camera (Canon Legria HF R37, acquiring at 25 frames per second (fps) with a resolution of $1080 \times 1920$ pixels) mounted on a stable support above a cradle or a physiotherapy treatment table where the infants could move freely facing the camera (see Figure 5.1). For each infant, one video (mean duration and standard deviation in minutes: 8 ± 2) was acquired during the wake condition. We excluded from the analysis video sequences where unintentional interventions of the operators obscured part of the scene and where the infants were crying or using the pacifier (a total of more than 1000 minutes remaining).

ACQUISITION SESSION 2: 3 MONTHS AFTER BIRTH. During the second session of acquisition, the movements of 118 infants were acquired 3 months after birth (78 females, born at 29 ± 2 weeks and weighting 1150 ± 303g). Among them, 95 are the same infants acquired also at the 40th week of gestational age. The inclusion criteria and the setup were the same as for *acquisition session 1*. Also in this case, one video was acquired for each infant (mean duration and standard deviation in minutes: 7 ± 2, for a total of more than 800 minutes)

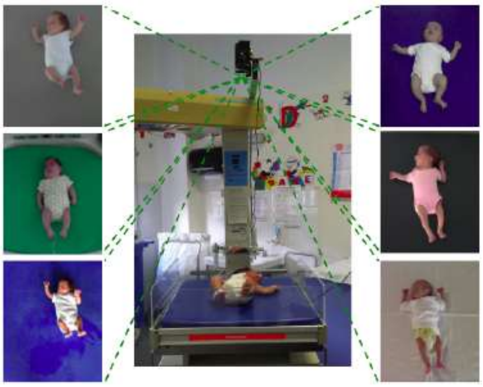


Figure 5.1: Data acquisition setup with the camera viewpoint. We show different examples of acquisition with background of different colors. The images are cropped to improve visibility and faces are anonymized for privacy.

EXPERTS VIDEO EVALUATIONS (VE40 AND VE3). For all the infants involved in the study we have also the Video Evaluations ($VE40$ and $VE3$ respectively for acquisitions done at the 40th week of gestational age and 3 months after birth) performed by two trained physicians relying only on the videos recorded during the different acquisition sessions and performed according to the General Movements Assessment (GMA). The evaluations done at the 40th week of gestational age ($VE40$) are less reliable than the ones performed 3 months after birth ($VE3$) since GMs and abnormal motion patterns are less evident at early stages even for trained physicians. The two experts, first assessed independently the videos and then, they reached a common evaluation (by discussing the ambiguous cases and agreeing on a decision) $VE40_j$, with $j = 1, 2, ....., 142$, $VE40_j = 0$ and $VE40_j = 1$ indicate respectively the absence and the presence in the recording of frames with abnormal motion patterns. Same notation for $VE3$ (but in this case $j = 1, 2, ....., 118$).

GROUND TRUTH (GT): CLINICAL EVALUATION. Among the infants involved in the study, some of them (59 acquired at the 40th week of gestational age and 53 acquired 3 months after birth) had a clinical diagnosis of neuro-motor disorders, while the others had not. The neuro-motor status assessments were done according to different evaluations and tests. Among them, Magnetic Resonance Imaging (MRI) was performed at birth and it was adopted to verify possible brain injuries. The other clinical neuro-motor evaluations were performed *30 months after the video recording* and for the majority of the infants the evaluation included the Bayley test [Bayley, 2006]. We considered these evaluations as Ground Truth (GT) for our analysis ($GT_j$ with $j = 1, 2, ....., N$, with $GT_j = 0$ and $GT_j = 1$ respectively for infants without and with neuro-motor disorders and with $N$ corresponding to the total number of infants). The infants involved in the study present a wide spectrum of intensity of motor disorders and most of them present minor impairments. Figure 5.2 shows a time line of infants neuro-motor evaluation and acquisitions.

The study and the consent form signed by parents were approved by the Giannina Gaslini Hospital Institutional Review Board on 20/06/2013 (protocol number: IGGPM01).

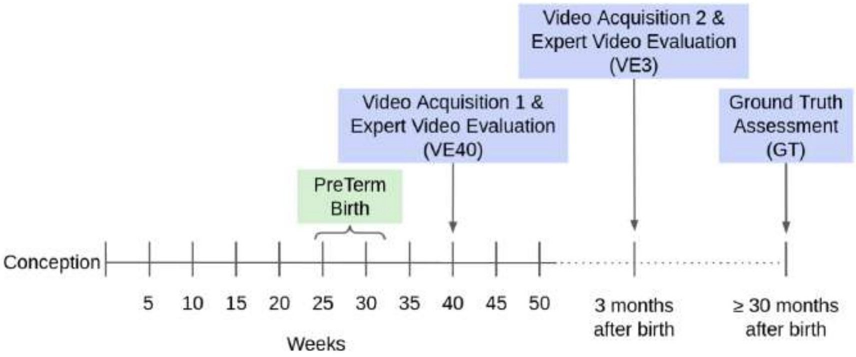


Figure 5.2: Time line of infants birth and clinical assessments.

## 5.4 Methods

In this section we summarize the methods adopted for data analysis from Chapter 4. The two proposed approaches (parameters- and graph-based) have the potential to highlight different characteristics about infants motion patterns. In particular, with the parameters-based analysis we are interested in the characterization of motion features able to distinguish among normal and abnormal motion patterns as soon as possible after birth. Thus, in this case, we rely on **acquisition session 1**. Conversely, with the graph-based analysis we are interested in the automatic recognition of body configurations similar to the ones observed with GMA, that are more evident 3 months after birth [Prechtl, 1997]. Thus, we rely on **acquisition session 2**. In future works, we are planning to extend both approaches to the whole dataset.

The two approaches share the first step of the analysis, consisting on the detection of interesting keypoints in the image plane, described in the following paragraph.

pre-processing: landmark points detection and filtering. We train DeepLabCut (DLC) [Mathis et al., 2018] to detect a small set of meaningful landmark points on the infants' body. The motivations behind this choice, instead than full body pose methods (*e.g.*, [Cao et al., 2017]) are: (i) classical full body pose estimation algorithms, if not fine tuned on infants poses, have proven to be not always appropriate for infants since they are trained and implemented for the detection of adults' poses [Chambers et al., 2020; Hesse et al., 2018], and they would require a significant amount of data for the fine tuning; (ii) the possibility to focus only on some key points that provide meaningful information regarding infants' motion and to guarantee a higher per-point accuracy and a higher control on the interpretability of the results; (iii) semantic feature detectors normally require a limited number of annotated examples in the training phase (see [Mathis et al., 2018]).

For the **parameters-based approach**, we train DLC with labelled examples from videos of **acquisition session 1** of our dataset. To extract quantitative information related to different body parts, we detect the positions in the image plane of the following landmark points: nose, hands and feet. We randomly select 10 frames from 120 videos and we manually label the points of interest. Then, we adopt that model to extract the positions of the five landmarks in each frame of each video. The outputs are $\{(x, y, c)^l_t\}^T_{t=0}$, with $l = \{\text{nose (N), left hand (LH), right hand (RH), left foot (LF), right foot (RF)}\}$. As mention in Chapter 4, $(x, y)^l_t$ is the position of the $l-$th point in the $t-$th frame (examples are shown in Figure 5.3) and $c^t_l$ –a number in the interval $[0, 1]$– is the corresponding likelihood. With $c^t_l$ we were able to quantify the uncertainty behind the detection of each point in each frame and filter them as described in Section 4.3. Furthermore, the trajectories were normalized in order to compensate for the possible differences in size of the infants' body and distances between the camera and the acquisition plane (see Section 4.5).

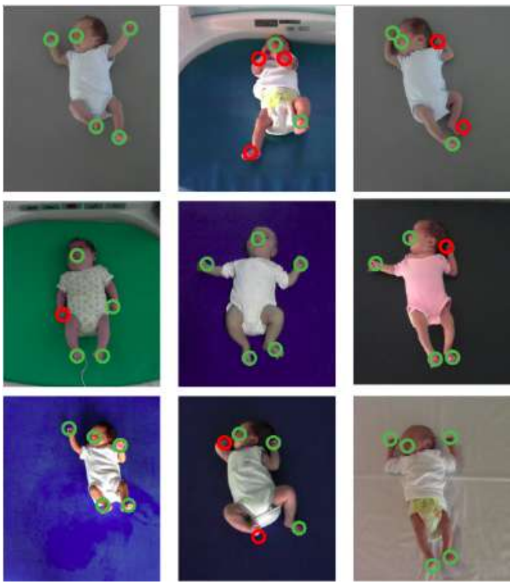


Figure 5.3: Examples of detected landmarks in the image plane. The green circles are the positions of some landmarks directly after the detection performed with our fine tuned DLC model. The red circles are the landmark points that were wrongly detected by the network (mispredicted or occluded) and whose positions were retrieved thanks to the filtering step.

As a final remark, in order to confirm that full body pose estimation are less accurate than *ad hoc* fine tuned semantic feature detectors, in Table 5.1 we report the mean error in pixels for the same landmark points obtained with DLC and the full pose estimator Openpose [Cao et al., 2017] with respect to manually annotated ground truth. To solve this task we randomly selected 680 images and we manually labeled them.

For the **graph-based approach**, we trained DLC with labelled examples from videos of **acquisition session 2**. In particular, we randomly select 10 frames

| *Point* | *DeepLabCut* | *Openpose* |
|---|---|---|
| Nose | 3.73 ± 2.43 | 7.68 ± 3.14 |
| Right Hand | 5.12 ± 3.35 | 8.73 ± 3.97 |
| Left Hand | 5.34 ± 3.58 | 8.82 ± 4.73 |
| Right Foot | 6.57 ± 4.13 | 9.74 ± 5.04 |
| Left Foot | 6.18 ± 4.28 | 9.98 ± 5.01 |

Table 5.1: Mean error ± standard deviation (SD) for each point in pixels computed considering a manually labeled ground truth in 680 images.

from 100 videos and we manually label the points of interest and we fine tune the ResNet-50 within the DLC framework.

Before further analysis, we need to compensate for the possible differences in *(i)* size of infants' body and *(ii)* distances between the camera and the acquisition plane. Therefore, we normalize the landmarks' coordinates within each frame with the maximum distance in the image plane between the nose and the virtual middle point between the feet across the whole video.

## 5.4.1 *Parameters-based approach*

Figure 5.4 summarizes all the steps performed in this approach.

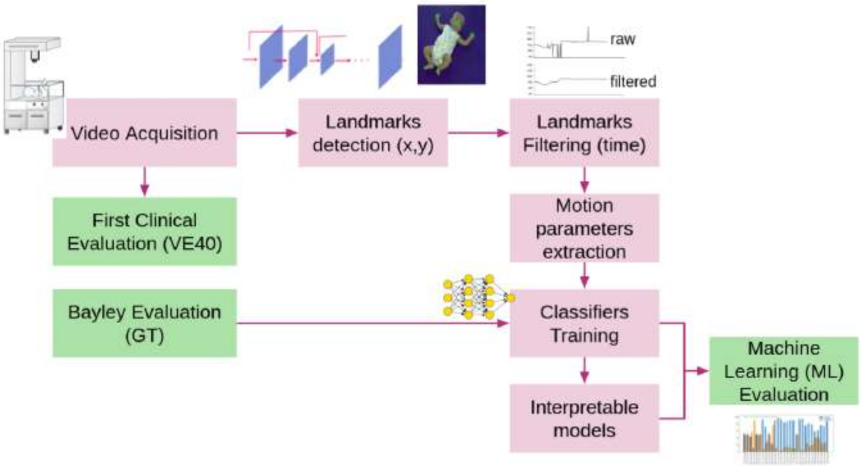

raw
filtered
Video Acquisition
Landmarks detection (x,y)
Landmarks Filtering (time)
First Clinical Evaluation (VE40)
Motion parameters extraction
Bayley Evaluation (GT)
Classifiers Training
Machine Learning (ML) Evaluation
Interpretable models

Figure 5.4: Summary of the steps implemented for parameters-based approach.

motion parameters extraction. Starting from the filtered and normalized signals that represent the evolution over time of the positions in the image plane of each landmark, we extract quantitative parameters that represent infants' motion patterns. Since the videos last on average 8 ± 2 minutes and since abnormal motion patterns are usually localized in time, we divide each video in time windows of length $W$ frames and we extract motion parameters in each time window (see Section 4.5.1 for further details). Among time windows we consider a stride of $S$ frames. The values of $W$ and $S$ we adopt expressed in number of frames are: $W = \{50, 100, 250, 500, 1000\}$ and $S = \{50, 100, 250, 500, 1000\}$, corresponding to $\{2, 4, 10, 20, 40\}$ seconds. In this way it is possible to highlight motion patterns of different granularity. Finally, we extract two different types of parameters: general and specific motor parameters.

***General motor parameters.*** Following the pipeline implemented in [Ahmedt-Aristizabal et al., 2019], in each window we extract 125 features – that we call $Vgen$ – (25 parameters for each detected landmark point - nose, right hand, left hand, right foot, left foot) described in Section 4.5.

***Specific motor parameters.*** Following [Meinecke et al., 2006] we compute a set of specific parameters for our application field – $Vspec$. These are derived from qualitative factors that are commonly identified by the physicians for evaluating infants' spontaneous motor activity and that can be summarized in: variability, smoothness and complexity of the motion [Sival, Visser, and Prechtl, 1992]. Corresponding quantitative parameters, as presented in [Garello et al., 2021; Meinecke et al., 2006], are:

- *Cross-Correlation*: a measurement of variability, to determine whether the movements of upper and lower limbs are correlated [Prechtl, 1990].
- *Skewness*: a statistical parameter that highlight asymmetry in the distribution of the speed profile of each landmark point; it is an indicator of smoothness [Meinecke et al., 2006].
- *Area out of moving average*: the area across time between a point's coordinate ($x$ or $y$) and its moving average. Moreover, we consider also the *area out of standard deviation of moving average*: the area between the trajectory of a point's coordinate and its moving average plus the standard deviation [Meinecke et al., 2006]. These are other descriptors for smoothness.
- *Periodicity*: an indirect measure of the complexity of the motion. Considering the time course of a certain landmark point's coordinate ($x$ or $y$), the periodicity is the number of intersections between the coordinate trajectory and its temporal average [Meinecke et al., 2006].

Thus, $Vspec$ comprises 24 parameters: cross-correlation of hands and feet speed; cross-correlation of hands and feet acceleration; the skewness, the area differing from moving average, the area out of standard deviation of moving average and the periodicity for each landmark point (nose, right hand, left hand, right foot, left foot).

In conclusion, we obtain a $149 - dimensional$ feature vector summarizing the two sets of parameters:

$$Vtot_j^k = Vgen_j^k \cup Vspec_j^k$$

with $j = 1, ..., 142$ the index for each infant and $k = 1, ..., N_j$ the index for the time windows – $N_j$ total number of time windows for the $j$-th infant.

CLASSIFICATION AND INTERPRETABLE MODELS. We adopt the computed parameters ($Vtot_j^k$) to train and test different binary classifiers as described in Section 4.5.2 in order to distinguish between infants with and without neuromotor disorders. To reason on how critical the choice of a classifier is, we consider all the 5 different classifiers presented in Subsection 4.5.2: (i) **SVM-poly**, with 3rd degree polynomial kernel; (ii) **SVM-rbf**; **RF** with 100 trees; **NN** with three hidden layers (with 64, 32, 4 units respectively and relu activation function); **LSTM** with two layers (64 units each) and a last dense layer with a sigmoid activation function.

During testing, we obtain one prediction for each time window ($Pred_j^k = 0$ and $Pred_j^k = 1$ respectively represent the absence or the presence of abnormal motion patterns) that are then grouped in order to have one predicted label for each infant $Pred_j$ (see Subsection 4.5.2). To compute the final $Pred_j$ we set a threshold $\tau$ on the percentage of windows classified as with neuromotor disorders:

$$PercImp_j = 100 * (\sum_{i=1}^{N_j} Pred_j^i)/N_j \tag{5.1}$$

If $PercImp_j \geq \tau$, the $j$-th infant is classified as with neuromotor disorders $Pred_j = 1$, otherwise the infant is classified as without neuromotor disorders $Pred_j = 0$. To evaluate the predictive power of the classifiers we compute the mean accuracy and the sensitivity (*i.e.*, the percentage of infants correctly identify as with neuromotor disorders with respect to the ground truth, *GT*). Focusing on $PercImp_j$, it is also possible to have an interesting measure related with the uncertainty of the prediction. For instance, if the number of time windows correctly detected is a high percentage of the their total number (*e.g.*, $PercImp_j \geq 75$ for an infant with neuromotor disorders) we could be more confident about the final prediction with respect to a lower percentage.

We start following the usual procedure for a binary classifier and we set $\tau = 50$: an infant is assigned to the majority class of the predicted labels. As a consequence, if the class corresponds to the ground truth, then the infant is correctly classified. Then, we verify the effect of selecting different thresholds. In the following, unless otherwise stated, we refer to $\tau = 50$.

As we reason on the design of methods to assist medical diagnosis, we ask ourselves which motor parameters are more relevant in the classification step. Among the available tools to provide interpretability for a classifier, we apply the two different approaches described in Subsection 4.5.2: (i) feature importance based on permutation, providing a score for each input feature and representing how much a classifier bases its decision on that specific attribute and (ii) rule extraction, specifically designed to support interpretability of neural networks.

## 5.4.2 *Graphs-based approach*

Figure 5.5 summarizes all the steps performed in this approach.

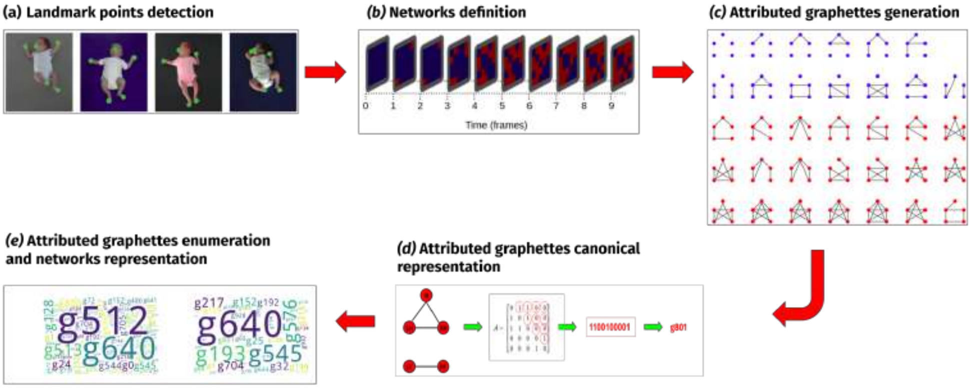

(a) Landmark points detection
(b) Networks definition
0 1 2 3 4 5 6 7 8 9
Time (frames)
(c) Attributed graphettes generation
(d) Attributed graphettes canonical representation
1100100001
g801
(e) Attributed graphettes enumeration and networks representation
g512
g640
g217
g193
g545

Figure 5.5: Summary of the steps implemented for graph-based approach.

**NETWORKS DEFINITION: ATTRIBUTED GRAPHETTES.** For each video, we build a temporal sequence of networks (one per frame). We consider the detected landmark points (N, RH, LH, RF, LF) as nodes of each network and we connect them through edges depending on their relative proximity as described in Section 4.5.4. In this way, each network represents a configuration at a specific frame and it is defined as a 5-nodes graph, which we exchangeably call *attributed graphette* or *configuration*. Edges are obtained by thresholding the Euclidean distance distributions between every pair of landmark points in all the images composing our dataset. The empirical distributions of normalized distances are shown in Figure 5.6.

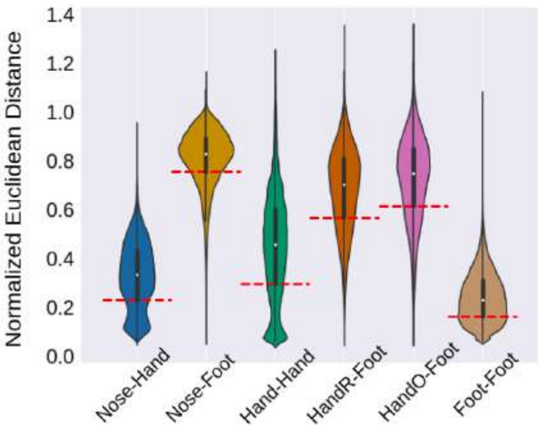


Figure 5.6: Plot of the empirical distributions of normalized intra-frame distances between landmark points. 'HandR' is the hand relative (R) to the same side of the body (right or left) as the foot; 'HandO' is the opposite (O) side with respect to the foot. The red dotted lines correspond to the 25th empirical quartiles of the distributions.

We state that if the distance between two landmark points is greater than the 25th quartile of the corresponding empirical distribution (see the red dotted line in Figure 5.6), then they are far from each other, and we do not connect them with an edge.

attributed graphettes representation and enumeration. Each video (*i.e.*, each infant) is now represented as a series of attributed graphettes. As mentioned in Subsection 4.5.4, graphettes are a suitable tool to give local and global descriptions of large complex networks. In order to do that, we need to exploit Natural Language Processing (NLP) methods that work with *words* and *texts*. For this reason, we associate each configuration with a *word*, *i.e.*, a string composed by the letter $g$ and an integer number. Since the number of possible configurations representing one frame is $2^{10}$, this integer number can range from 0 to 1023. Some of these configurations are not present in our dataset. Thus, the vocabulary of all configurations appearing in our dataset consists of 650 attributed graphettes (and, consequently, words). The final result of this step is the association of each frame $t$ of a video with an attributed graphette, represented by the corresponding string. Then, we leverage NLP methods (*e.g.*, Bag-Of-Words (BOW) and Latent Dirichlet Allocation (LDA)) to describe infants motion in terms of configuration occurrences.

nlp methods. Specifically, we use BOW to convert each infant's temporal network into a vector of size $1 \times 650$ containing the occurrences of each word. In order to identify meaningful configurations, we need to normalize raw counts in BOW vectors. For this purpose, we leverage Term Frequency - Inverse Document Frequency (tf-idf), a common algorithm to transform word counts into meaningful real numbers [Salton and Harman, 2003] and to reduce the dimensionality of the representations. Unfortunately, the dimensionality reduction performed with the tf-idf is still not enough to allow an interpretable characterization of motion patterns. Thus, starting from the representations obtained with the previous steps, we rely on topic modeling [Alghamdi and Alfalqi, 2015] and, in particular, on Latent Dirichlet Allocation (LDA) [Blei, Ng, and Jordan, 2003]. The result is an interpretable low-dimensional representation of videos, able to describe the distribution of attributed graphettes for each infant and also able to give local information on the dynamic of infants by considering co-occurrences of configurations.

***Data Augmentation.*** Typically, in order to obtain reliable and stable topics, LDA needs to be trained on a large amount of data. Our dataset is composed of 118 infants (65 with normal and 53 with abnormal motion patterns) which is too small to infer meaningful topics from LDA. Thus, in order to augment the dataset, we simulate videos from the two classes (*i.e.*, infants with normal (N) and abnormal (Ab) motion patterns) until we obtain a balanced dataset of 1130 videos (118 original videos, 500 and 512 simulated videos from the classes N and Ab respectively). Simulated networks are composed of 10710 consecutive configurations, which is the average number of frames composing original infants videos. We simulate temporal networks by leveraging normalized bigrams (*i.e.*, couples of adjacent configurations) counts from the original dataset. More specifically, given a temporal network $G$ we compute bigram frequencies and associate every configuration $g_n$ with a vector $v_{g_n} = (bf_i)_{i=1}^{650}$, where $bf_i$ corresponds to the normalized frequency of the bigram $(g_n \, g_{n(i)})$ in $G$, $g_{n(i)}$ identifying the $i$-th configuration in the vocabulary. Thus, for every infant, we obtain a matrix $X_G$ (650 × 650) describing an infant-specific conditional distribution over configurations. We generate networks by first picking at random an infant from a chosen class and a starting configuration, then we iteratively sample configurations from the probability distribution identified by $X_G$.

***Number of topics selection.*** One of the most crucial LDA hyperparameters that needs to be tuned is the number of topics. In the literature, many metrics have been defined in order to find an optimal number of topics [Blei, Ng, and Jordan, 2003; Röder, Both, and Hinneburg, 2015]. We focus on the maximization of the *Intrinsic Topic Coherence Measure* (ITCM) [Mimno et al., 2011], which is a metric based on the co-occurrence of words within the documents being modeled. For every topic $p$, ITCM is defined as

$$ITCM(p, V^p) = \sum_{m=2}^{M} \sum_{h=1}^{m-1} \log \frac{df(v_m^p, v_h^p) + 1}{df(v_h^p)} \tag{5.2}$$

where $V^p = (v_1^p, \ldots, v_M^p)$ is a list of the $M$ most probable configurations in the topic $p$, $df(v_m^p, v_h^p)$ is the number of documents where the configurations pair $v_m^p$ and $v_h^p$ appear together in, and $df(v_h^p)$ is the document frequency of the configuration $v_h^p$. For each topic, co-occurrence frequencies of the $M$ most probable configurations ($df(v_m^p, v_h^p)$ in Equation 5.2) are computed within fixed-size temporal windows for every network. We consider $M = 10$ and a temporal window size equal to 110 frames. We select an optimal number of topics ($NoT$) by studying how ITCM varies as $NoT$ ranges in $\{2, 3, 4, 5, 6, 7\}$ when applying LDA to the tf-idf transform of the augmented dataset for different settings of maximum and minimum document frequencies. As shown in Figure 5.7, we determine that an optimal ITCM is reached at $NoT = 5$, maximum and minimum document frequency equal to 70% and 45%, respectively.

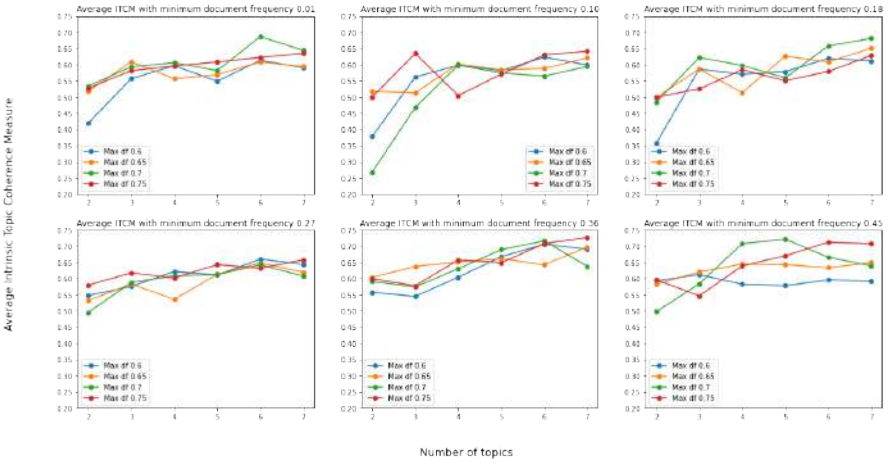


Figure 5.7: Average ITCM evaluated for number of topics ($NoT$) ranging in $\{2, 3, 4, 5, 6, 7\}$ and for different values of maximum and minimum document frequencies in the tf-idf representation. As shown in the bottom right corner, the optimal choice is $NoT = 5$, maximum df equal to 70% and minimum df equal to 45%.

## 5.5 Results: parameters-based approach

In this section, we report the main results obtained with the parameters-based approach. The results are reported following four main experiments: (1) first, the performances of the 5 different classifiers are computed, considering as ground truth the evaluation at 30 months after birth ($GT$); their accuracy is also compared with the video analysis provided by the experts ($VE40$). This experiment provide us an understanding of the complexity of the early diagnosis task at 40th weeks of gestational age. (2) Then, we apply features importance and rule extraction algorithm focusing on the models with the highest classification accuracy. These algorithms allow us to reason on the predictive power and the interpretability of the results. (3) On the same models as in (2), we investigate the role of the threshold $\tau$ in Equation 5.1. (4) Lastly, we assess the potential of our method as a tool to support the clinical evaluation.

### 5.5.1 *Normal vs abnormal motion patterns classification accuracy*

We start by comparing the different classifiers listed in Subsection 5.4.1 on different choices of $W$ ($W = \{50, 100, 250, 500, 1000\}$) and $S$ ($S = \{50, 100, 250, 500, 1000\}$). We evaluate the accuracy of each classifier performing a 5-fold cross-validation in order to have also a measure of the stability across different sets of test infants. Figure 5.8 shows the results for each pair of $W$-$S$ for each classifier. We report the mean accuracy (colored dots), the standard deviation and also the maximum value (text numbers) across the 5-fold. It can be noticed that there are not significant differences in terms of mean accuracy across the different classifiers and for different pairs of $W$ and $S$. However, RF and LSTM lead to slightly better performances, with maximum accuracy of 78.5% and 75% respectively. This can be better appreciated in Table 5.2 that shows, for each classifier, the overall mean (mean across different pairs of $W$-$S$), the mean of maximum values and the absolute maximum values.

The overall mean accuracy reported in Table 5.2 (around 60%) highlights the complexity of the task and it is confirmed by the accuracy obtained by the experts video evaluation $VE40$ (52.1%). Then, we focus on the models that allowed to reach the maximum accuracy, *i.e.,* that better reflected the differences between the two classes, and we analyze which parameters are more important for the classification task. In this way we are able to address the need of this application field to provide results as interpretable as possible, that physicians could adopt as a tool to support their diagnosis.

### 5.5.2 *Models interpretability assessment*

In the previous subsection, in Table 5.2, we highlight the fact that RF and LSTM allowed to reach the best results. Following this direction, we focus on

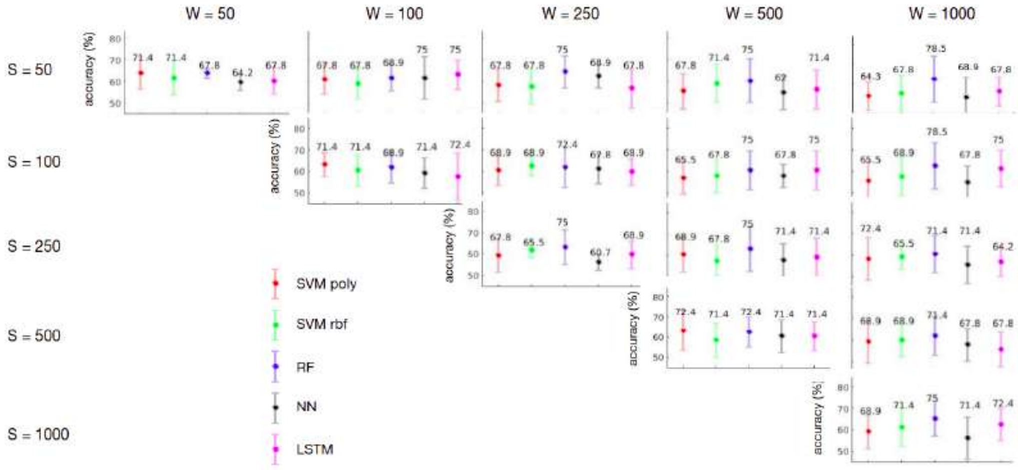


Figure 5.8: Classification accuracy for all the possible pairs of $W$ - $S$ for all classifiers. In each plot we show the results for all the five different classifiers, fixing $W$ and $S$. Each color refers to a different classifier (as reported in the legend - red: SVM with polynomial kernel; green: SVM with rbf kernel; blue: random forest; black: fully connected neural network; magenta: architecture LSTM-based) and for each case we report the mean (colored dot) with the correspondent standard deviation and the maximum value (text numbers) for the 5-fold cross-validation.

| | Overall Mean (SD) | Mean Max (SD) | Max |
|---|---|---|---|
| **SVM poly** | 59.3 (2.8) | 68.6 (2.4) | 72.4 |
| **SVM rbf** | 59.4 (2.0) | 68.9 (2.0) | 71.4 |
| **RF** | **62.6** (1.4) | **73.3** (3.2) | **78.5** |
| **NN** | 58.0 (2.8) | 68.5 (3.8) | 75.0 |
| **LSTM** | 59.5 (2.1) | 70.5 (3.2) | 75.0 |

Table 5.2: Here we report: (i) the mean values of accuracy across all the pairs $W$-$S$ for each classifier and the related standard deviation (SD); (ii) the mean of maximum values across all the pairs $W$-$S$ for each classifier and the related standard deviation (SD); (iii) the absolute maximum classification accuracy values

these two choices, selecting the trials that allow to reach the best classification accuracy and we explore the importance of each parameter to highlight those that are more important in the classification task.

#### 5.5.2.1 *Feature importance*

Considering the RF model with $W = 1000$ and $S = 100$, we apply the feature importance algorithm; in Figure 5.9 we report the first 25 parameters with higher importance coefficient. In this case there appear to be a single, meaningful, parameter: the median value of the acceleration profile for the right foot.

All the others are distributed across different types of parameters, and no specific motion quality seems to emerge. Notice that the parameters set we use is highly redundant so it is likely that different combinations of parameters can lead to similar classification performance.

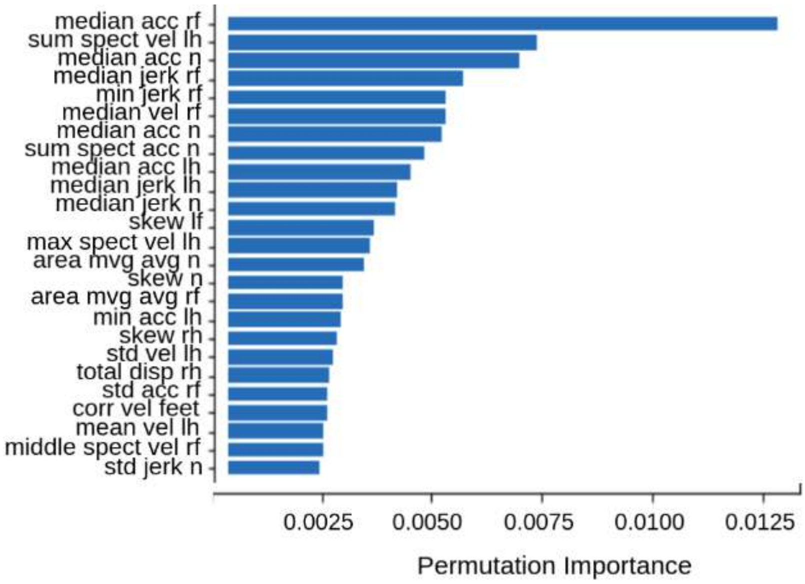


Figure 5.9: First 25 parameters with higher feature importance coefficient. Abbreviations – acc: acceleration, vel: speed, spect: spectral, mvg: moving, avg: average, displ. displacement, std: standard deviation, rf: right foot, lf: left foot, rh: right hand, lh: left hand, n: nose.

#### 5.5.2.2 *Rule extraction algorithm*

On the LSTM model with parameters $W = 500$ and $S = 100$, we apply the rule extraction algorithm described in Subsection 4.5.2: a subset of 23 input features is found significant after the pruning step (Figure 5.10 shows the detailed list). The accuracy of the LSTM model using such significant input attributes is equal to the one on the entire input set (75%). The 23 input features are used to extract the related rules [Augasta and Kathirvalavakumar, 2012]. The classification accuracy using the constructed set of rules is 71.4%. Figure 5.10 represents the data-ranges corresponding to the extracted set of rules. While the same comment on the parameter redundancy would apply also in this case, the extracted rules present some interesting intepretability insights. In particular, we can notice that, for the majority of the normalized input data, the ranges that characterize infants with neuro-motor disorders are close to zero, also for the last 7 parameters that describe the smoothness and the complexity of the motion patterns. This is in line with prior knowledge from the medical domain.

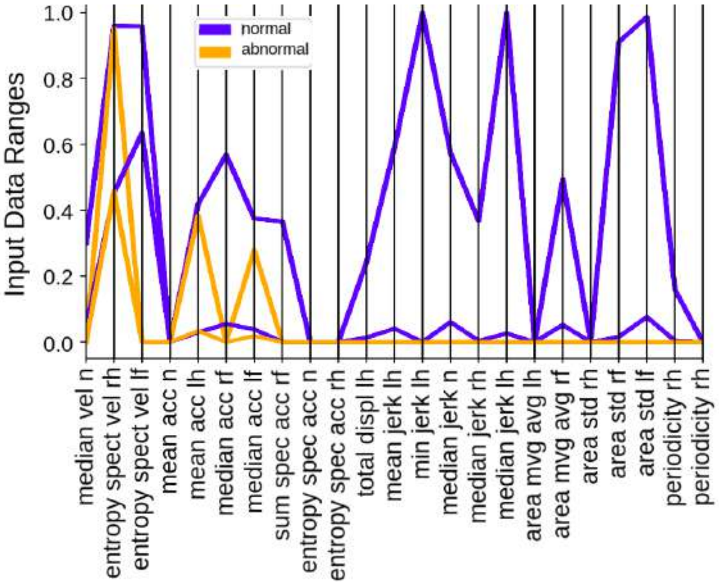


Figure 5.10: Normalized input data ranges for the extracted rules. The blue curves represent the input ranges that caused the LSTM model to classify a sample as without neuro-motor disorders, while the orange curves correspond to the infants with neuro-motor disorders. As it is possible to notice, most of the normalized parameters related with the characterization of infants with neuro-motor disorders have values close to 0. Same abbreviations as in Figure 5.9

### 5.5.3 *Choice of the threshold $\tau$*

The threshold $\tau$ in Equation 5.1 for the previous experiments is selected according to the standard threshold for a binary classification problem ($\tau = 50$). For the nature of our application task, a threshold $\tau = 50$ is not necessarily the best option due to the fact that many infants in our dataset present minor motor impairment. Also, abnormal motion patterns can be present only during few time instants of the video recording and, consequently, in few time windows. Following these considerations, we investigate the influence of the threshold $\tau$ in the classification accuracy for both models explained in the previous subsection. The lower the values of $\tau$ the higher the probability of an infant being classified as impaired (more importance given to $PercImp$), and vice versa. Thus, the choice to select lower $\tau$ accounts for the importance of the detection of pathological cases. Specifically, in this application, it is preferable to have a detection of false positive that will result on an unnecessary clinical exam, than a false negative, *i.e.*, failing to detect an infant at risk. Thus, it is essential to give more importance to the correct detection of abnormal motion patterns ($\tau < 50$) than the normal patterns.

Table 5.3 reports the results of this experiment in terms of overall classification accuracy. We varied $\tau$ between 20 and 50 (steps of 5%) for both the two best original models and the extracted equivalent set of rules. As it is possible to notice from Table 5.3, the accuracy, in all the cases, increases and reaches a maximum value around $\tau = 30$ or $\tau = 35$ and then decreases again. The extracted set of rules provides interpretability to the trained neural networks predictions, indicating the exact parameters and their ranges that cause the network to predict if an infant has a neuro-motor disorder. More importantly it also improves the classification accuracy that, as a consequence of the pruning procedure, reached a maximum of 85.7%.

| | **20** | **25** | **30** | **35** | **40** | **45** | **50** |
|---|---|---|---|---|---|---|---|
| **RF** | 71.4 | 78.5 | 78.5 | 82.1 | 78.5 | 78.5 | 78.5 |
| **LSTM** | 64.2 | 67.8 | 67.8 | 82.1 | 78.5 | 78.5 | 75.0 |
| **Extracted rules** | 82.1 | **85.7** | **85.7** | 82.1 | 78.5 | 78.5 | 71.4 |

Table 5.3: Overall classification accuracy with different threshold $\tau$ values (20, 25, 30, 35, 40, 45, 50) for the selected best models (RF with $W = 1000$ and $S = 100$ and LSTM with $W = 500$ and $S = 100$) and for the set of extracted rules.

To evaluate the potential of our models specifically in the early detection of infants with abnormal motion patterns (true positives), in Table 5.4 we report the sensitivity of our algorithm, *i.e.*, the ratio between the true positives and the positives, depending on the threshold $\tau$, in the recognition of infants with neuro-motor disorders. As expected, the lower the threshold the higher is the sensitivity of the algorithm in the classification of infants with neuro-motor disorder. Combining the information in Table 5.3 and in Table 5.4 we can highlight that decreasing - until a certain value - the threshold $\tau$ helps the correct classification of a higher number of infants with neuro-motor disorders, not affecting the number of infants without neuro-motor disorders correctly classified.

| | **20** | **25** | **30** | **35** | **40** | **45** | **50** |
|---|---|---|---|---|---|---|---|
| **RF** | 72.7 | 72.7 | 72.7 | 72.7 | 63.6 | 63.6 | 63.6 |
| **LSTM** | 72.7 | 63.6 | 63.6 | 63.6 | 54.5 | 54.5 | 45.4 |
| **Extracted rules** | **81.8** | 72.7 | 72.7 | 54.5 | 45.4 | 45.4 | 27.2 |

Table 5.4: Sensitivity with different threshold $\tau$ values for the selected best models (RF with $W = 1000$ and $S = 100$ and LSTM with $W = 500$ and $S = 100$) and for the set of extracted rules.

### 5.5.4 *Early diagnosis assessment*

We focus on the models adopted in the previous subsections and we provide more details regarding the prediction potential as an early diagnosis tool.

For this purpose, we first report in Figure 5.11 more details on the percentage of time windows classified as with (orange, $PercImp$) or without (blue, $100 - PercImp$) neuro-motor disorders for each test infant. The first 11 infants

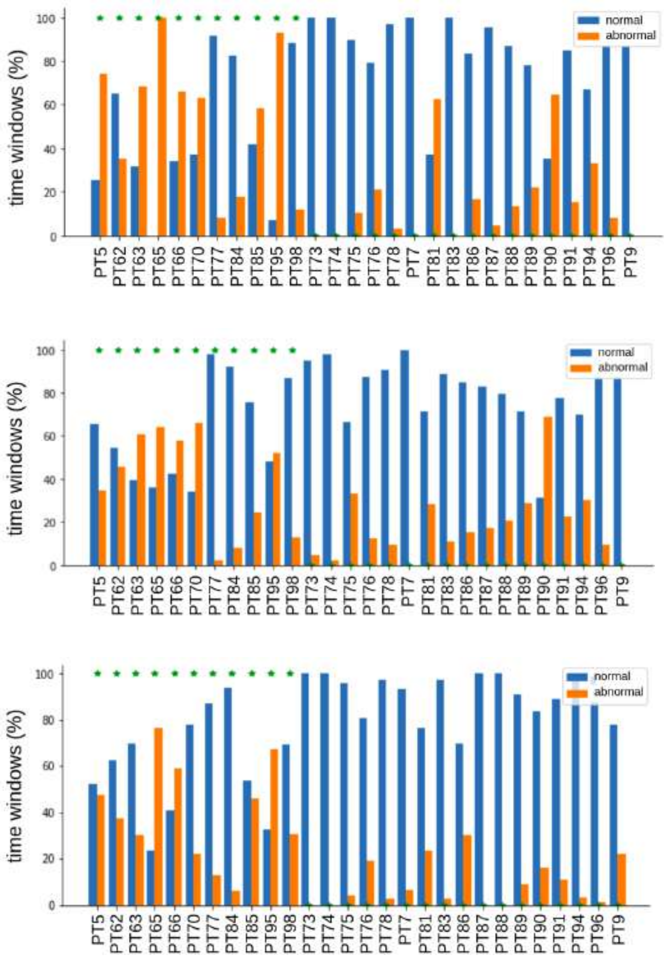


Figure 5.11: We report the % of windows predicted as with normal (blue) and abnormal (orange, *PercImp*) motion patterns for test infants of the same fold (x axis). The green stars for the first subjects indicate infants with neuromotor disorders (according to *GT*). At the top we report the results for the random forest with $W = 1000$ and $S = 100$; in the middle the architecture LSTM-based with $W = 500$ and $S = 100$; at the bottom the results obtained adopting the extracted set of rules.

are the subset of cases with neuro-motor disorders (according to *GT*). The top panel of Figure 5.11 reports the details for the random forest with $W = 1000$ and $S = 100$. The middle panel reports the results for one of the fold for the architecture based on LSTM with $W = 500$ and $S = 100$. The bottom panels shows the percentage or windows classified with the extracted set of rules. From these plots it is possible to notice that the percentage of time windows classified as impaired is on average higher (even if not $> 50\%$) for infants with neuro-motor disorders.

Then, we compare the sensitivity and specificity of our predictions in association with $VE40$. Figure 5.12 shows the agreement between the computer-aided predictions with the set of rules and $\tau = 30$ (accuracy of 85.7%) and

$VE40$ (accuracy of 50%). In this case, our pipeline can correctly detect 8 out of 11 infants at risk and 16 out of 17 infants without neuro-motor disorders.

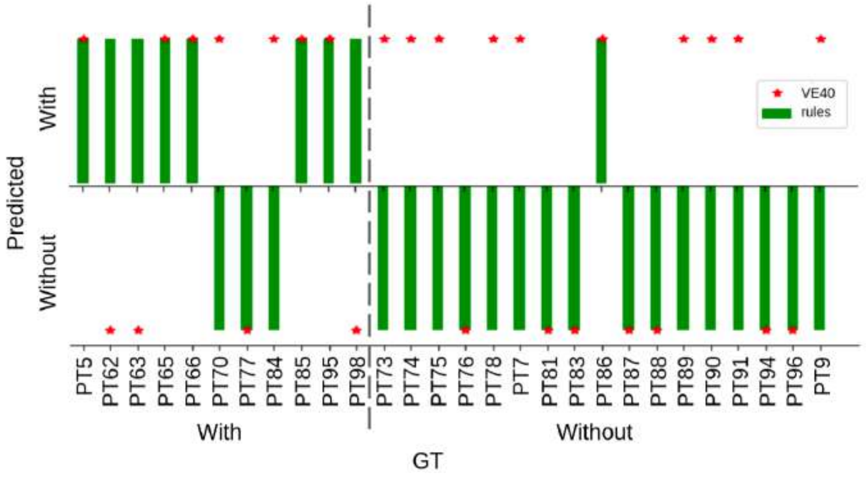


Figure 5.12: Agreement of our predictions with $VE40$. In this way we highlight the potential of our pipeline as a tool to support the experts in the detection of normal/anomalous motion patterns. On the x axis there are the infants divided in with and without neuro-motor disorders according to $GT$ and on the y axis the predicted label ($VE40$ the red stars and our machine learning based prediction - extracted rules with $\tau = 30$ - in green).

## 5.6 Results: graph-based approach

We fit the LDA model with the hyperparameters retrieved following the steps described in Subsection 5.4.2 to the augmented infants dataset and we obtain 5 topics describing local motion patterns. Figure 5.13 shows the topics summarized by their 5 most probable configurations. Topic-specific most probable configurations differ from each other only by a few edges and also appear as little modifications of a basic configuration. This is evident by looking at the first 2 most probable configurations in Figure 5.13. For instance Topic 2 is well summarized by the configuration in which the only present edges are the ones which connect a hand with the corresponding foot, meaning **LH-LF** and **RH-RF**. Indeed the 2 most probable configurations appear as slight deviations from this basic configuration.

Then, we study topic proportions for every network in the original dataset in order to look for differences between the networks representation of infants with normal and abnormal motion patterns. Topic proportions of networks provide us with a global description of infants movement. Indeed, for each network in the dataset, larger mixture components correspond to topics whose most probable configurations are peculiar to the corresponding infant's motion sequence. Furthermore, topic proportions are suitable to be interpreted as probabilistic assignments to clusters, which are identified by the corresponding topics.

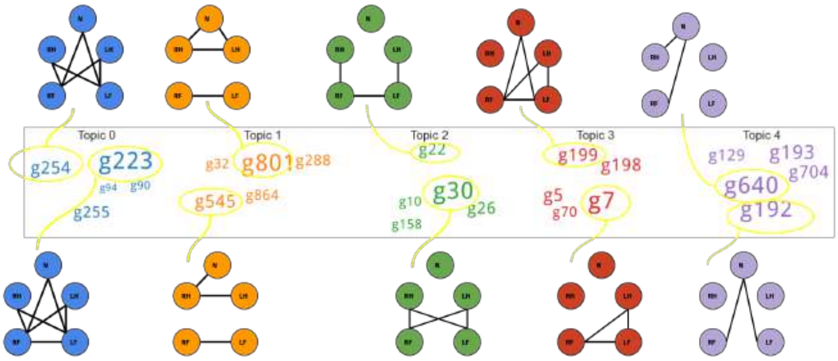


Figure 5.13: Visual representation of the five obtained topics described by their 5 most probable configurations: the top 2 are depicted as graphs whereas the last 3 are synthesized by their encoding (text). The size of a configuration encoding is proportional to its weight in topic-configurations probability distribution.

We perform class-specific topic proportions analysis, as reported in Table 5.5. In particular, for each network in the dataset, we observe the largest mixture component in its topic representation, that tells us the confidence in assigning the network itself to the corresponding topic. Once assigned the infants to the corresponding prevalent topic, we compute intra-topic, class-specific mean, minimum and maximum probability assignments. We claim that such statistics are good descriptors of the variety of intra-class motion. Also, for each topic, we compute the concentrations of infants in normal and abnormal motion patterns classes assigned to it. Differences in such concentrations would indicate different global motion patterns between the two classes. Furthermore, for each topic, we evaluate the mean global symmetry and density of the 5 most probable configurations as well as the mean symmetry of hands and feet neighborhood. In general, from Table 5.5 we can observe that:

1. no significant differences are detected in the concentrations of infants assigned to each topic.
2. Infants with normal motion patterns are more uniformly distributed among the 5 different topics meaning that they present a higher variability in terms of motion patterns.
3. Infants with abnormal motion patterns are well represented in Topic 0 and Topic 4 (considering the minimum and the mean probability assignments respectively).

## 5.7 Discussion

The common aim of the two methods presented was to apply our implemented markerless pipeline in order to study, characterize and classify infants' spon-

| | **Intra-class probability** | | | | | | | | **Symmetry** | | | **Density** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **class N** | | | | **class Ab** | | | | | | | |
| | Mean | Min | Max | Conc | Mean | Min | Max | Conc | Global | Hands | Feet | |
| **T0** | 0.59 | 0.34 | 0.9 | 0.15 | 0.67 | 0.6 | 0.91 | 0.15 | 0.94 | 0.95 | 0.78 | 0.62 |
| **T1** | 0.8 | 0.41 | 1.0 | 0.28 | 0.74 | 0.42 | 1.0 | 0.28 | 0.94 | 0.87 | 0.60 | 0.28 |
| **T2** | 0.73 | 0.46 | 1.0 | 0.17 | 0.74 | 0.36 | 1.0 | 0.13 | 0.90 | 0.80 | 0.75 | 0.34 |
| **T3** | 0.6 | 0.4 | 0.99 | 0.23 | 0.71 | 0.33 | 0.93 | 0.17 | 0.80 | 0.50 | 0.58 | 0.34 |
| **T4** | 0.7 | 0.44 | 0.98 | 0.17 | 0.82 | 0.47 | 1.0 | 0.26 | 0.92 | 0.20 | 0.68 | 0.24 |

Table 5.5: Results of topic analysis. For each topic, we report statistics on: Intra-class assignment probability (mean, minimum, maximum, and concentrations), Symmetry (global, hands, and feet), and Density of the 5 most probable configurations. Class N stands for normal motion patterns and class Ab for abnormal motion patterns.

taneous movements. Both approaches do not require expensive and obtrusive technologies and they are based only on RGB video analysis. We opted for a 2D analysis to provide an easy to use system that - in our long term plan - could be adopted also by non-expert users at home.

With the parameters-based approach, we evaluated the power of our implemented pipeline to discriminate between infants with and without neuromotor disorders as soon as possible after birth. Indeed, we relied on the videos recorded in the first weeks after birth (40th week of gestational age). We obtained encouraging results in terms of mean overall accuracy, especially considering the task complexity. In fact, the majority of infants with neuromotor disorders presented minor motor impairments and identifying abnormal motion patterns in infants before the 12th months after the conception is difficult also for trained experts. Furthermore, adopting features selection techniques, we highlighted the most meaningful parameters considered during the classification task and we further increased the overall accuracy (reaching 85.7%). In terms of future developments, we will explore the role of different parameters and adopt a dense motion estimation [Stahl et al., 2012]. Also, it would be interesting to divide infants with neuromotor disorder in more classes depending on their impairment.

With the graph-based approach, we attempted to reflect qualitative aspects of infants motion patterns considered by experts physician during the motor evaluation, for example the detection of body configurations similar to the ones observed with GMA. For this reason we focused on data acquired 3 months after birth, when these configurations are more easily detectable [Prechtl, 1990]. Also with this approach, we could highlight correspondences between our analysis and the qualitative aspects of the motion usually considered during visual evaluation of expert physicians ($VE3$). In particular, we highlighted higher motion variability associated with infants with normal motion patterns and dense configurations and with a higher level of symmetry in infants with abnormal motion patterns.

6

# Gait Analysis

In this chapter we present the application of our proposed markerless approach for 2D and 3D gait analysis. Part of the work presented in this chapter was founded by Fondazione Italiana Sclerosi Multipla (FISM – 2019/PR-single050) and carried out in collaboration with San Martino Hospital (Genova, Italy), Dipartimento di Neuroscienze, Riabilitazione, Oftalmologia, Genetica e Scienze Materno-Infantili (DINOGMI).

## 6.1 Introduction

Gait analysis is an essential functional evaluation in clinics [Fritz et al., 2015; Langhorne, Coupar, and Pollock, 2009]. It consists on the extraction of quantitative parameters that can describe the quality of gait. It allows to better understand the evolution of specific neurological diseases and how they affect lower limbs. It helps also physiotherapists and physicians in the decision of the best treatment that can improve the quality of the gait and, consequently, the quality of life. Quantitative assessments ensure repeatability and objectivity of the analysis with respect to visual observations [Wren et al., 2020]. Indeed, this kinematic quantification has been a major technical challenge for many years in the mid 90's [Whittle, 2014]. Important applications of gait analysis are for example the motion analysis of stroke survivors and of people with Multiple Sclerosis (MS), where the recovery of the walking abilities is one of the primary goals [Langhorne, Coupar, and Pollock, 2009].

Our goal is to evaluate the appropriateness of our markerless pipeline as an alternative to classical marker-based systems. To do that, we perform gait analysis with gold standard systems and with our proposed markerless approach on the same dataset and we statistically compare the results obtained. We compare the analysis both in 2D and in 3D in two different works.

- Firstly, we focus on the 2D experimental assessment and we compare the quality of our estimated parameters with the ones obtained from the study [De Luca et al., 2018] and report promising results: most of the 2D parameters may be computed by our markerless method at a comparable precision. We do not find statistically significant differences between the elevation angles computed with marker-based system and markerless one. Furthermore, we succeed in highlighting the differences between the parameters of the impaired leg and the unimpaired one in hemiparetic stroke survivors, similarly to analysis carried out by conventional methods. For this work, we adopt a subset of the dataset presented in [De Luca et al., 2018], where RGB videos were acquired with the aim of visual assessment and not for an organized computer-aided analysis. In fact, in the clinical practice, videos are recorded for manual inspection, but they are never used for automatic analysis. In this direction, our work opens the possibility for a massive data analysis campaign.

- Then, we extend our analysis and we perform a comparison in 3D. In this direction, we acquire an *ad hoc* multimodal dataset with a gold standard marker-based motion capture system and a multi-view RGB cameras system to be able to perform a geometric 3D reconstruction. Also in this case we do not observe major differences among the two approaches.

## 6.2 Related works

Many efforts have been done in the last few years to implement and test video-based systems able to characterize human gait without using cumbersome and intrusive markers placed on the body skin. In this section, we present works that addressed this problem by following approaches that differ for: the dimensionality of the considered space (2D or 3D analysis), type of cameras, *e.g.*, depth cameras (RGBD) or RGB cameras, and type of algorithms (deep learning or classical approaches).

[Rodrigues et al., 2020] developed a markerless multimodal motion capture system using multiple RGBD cameras to determine spatio-temporal gait parameters. However, additional IMUs were mounted to the lower limbs of the participants to determine the joint angles. [Corazza et al., 2006] managed to extract the walking people's silhouettes from 16 RGB camera views. These silhouettes from different perspectives allowed the researchers to reconstruct the visual hull of the subject as a 3D model. By post-processing this model, the relevant joint angles could be determined. The authors could achieve good performance determining the angles on the sagittal plane, but with larger errors on smaller angles such as the knee adduction angle. Examples of similar approaches that used one or more RGB cameras and extracted silhouettes or used RGBD cameras can be found in [Castelli et al., 2015; Clark et al., 2013; Gabel et al., 2012; Kwolek et al., 2019; Saboune and Charpillet, 2007].

Recently, due to the continuous progress in terms of accuracy and computational costs of pose estimation algorithms based on deep learning architectures, there is an increasing interest in the study of video-based systems to perform gait analysis. [Kidziński et al., 2020] performed 2D gait analysis starting from the detection of keypoints in the image plane and, then, analyzing their trajectories extracting the joint angles and their changes on the gait cycle. They analyzed data from 1792 videos of 1026 patients with cerebral palsy. This approach has the potential to assess early symptoms of neurological disorders by using inexpensive and readily available technology. This work succeeded in performing a quantitative movement analysis using single camera videos in a stable way with results comparable to standard marker-based methods. Unfortunately, they have limitations due to the pure 2D nature of the images, limiting the analysis to elevation angles [Borghese, Bianchi, and Lacquaniti, 1996] and only a subset of spatio-temporal parameters.

[Vafadar et al., 2021] performed markerless gait analysis by first reconstructing an accurate human pose in 3D from multiple camera views. To this aim, they collected a gait-specific dataset: 31 participants, 22 with normal gait and 9 with pathological gait participated in the data collection. The researchers recorded the gait of the participants with a standard marker-based system and with 4 RGB cameras. For 3D pose estimation they relied on the approach proposed by [Iskakov et al., 2019]. They were successfully able to reconstruct the human pose while walking in 3D. However, they did not include feet on the keypoints detected and, consequently, they were not able to extract all the spatio-temporal parameters and the joint angles usually computed in gait analysis.

## 6.3 Datasets

stroke survivors (2d). For this part of the study, we analyze a subset of the data presented in [De Luca et al., 2018]: considering only subjects for which RGB videos is present.

Ten chronic stroke survivors (mean age ± standard deviation: 62.75 ± 12.29 years old) volunteered to participate and provided written informed consent (participants' information are reported in Table 6.1. Seven of them are female and three of them male; one has a left hemiparesis and the other nine have a right hemiparesis.

| ID | Age | DD | Gender | Paretic Leg | Walking Aids |
|---|---|---|---|---|---|
| S1 | 46 | 6 | Female | Left | No |
| S2 | 43 | 2 | Female | Right | Cane |
| S3 | 60 | 8 | Female | Right | Cane |
| S4 | 69 | 5 | Female | Right | Cane |
| S5 | 41 | 6 | Female | Right | No |
| S6 | 75 | 8 | Male | Right | Cane |
| S7 | 69 | 17 | Male | Right | No |
| S8 | 73 | 5 | Female | Right | Cane |
| S9 | 72 | 9 | Male | Right | Cane |
| S10 | 69 | 9 | Female | Right | No |

Table 6.1: Subjects information. DD means disease duration.

All the data were recorded in a gait analysis laboratory. Body motion was recorded by a stereophotogrammetic system (SMART DX, BTSBioengineering, Milan, Italy). The system consisted of eight infrared cameras (SMART- DX 5000 BTSBioengineering) that allow measuring body movement. We recorded twenty-two reflective spherical markers (15 mm diameter) positioned on anatomical landmark points according with the DAVIS protocol [Davis, 1988].

Two RGB cameras (BTS VIXTA) acquiring at 25 frame per second (dimensions: 640x480 pixels) were present in this setup. The first one was positioned in order to acquire the sagittal plane of the subject (lateral view) and the other one the frontal plane (frontal view). Important kinematic parameters are computed from the information obtained from the sagittal plane, so we focused on the first type of videos [Whittle, 1996]. It is worth noticing that cameras were included in the set-up with the sole purpose of providing a source of information for future visual inspection to be carried out by clinicians. For this reason the quality of the signal is quite low and the the positions of the cameras are not optimal for an automatic analysis task. Also, no calibration procedure took place at the time of data acquisition. Hence, we do not have access to extrinsic nor intrinsic parameters.

Subjects walked multiple times on an eight-meter long pathway, inside the acquisition volume of the infrared cameras, and stepped on two force platforms located half way. Both walking directions were considered. They were instructed to walk as naturally as possible and at their preferred speed and to walk straight from one side to the other of the pathway.

healthy participants (3d). In this case, we acquired 16 unimpaired participants (6 females, mean age ± standard deviation: 27 ± 2 years old) without known history of orthopaedic injuries or neurological diseases. We asked the participant to walk naturally in a straight lines from one side of a room to the opposite. The path was 6 meters long. Each participant performed 20 trials, 10 for each direction.

The setup for data acquisition (see Figure 6.1) included (i) a calibrated multi-view camera system consisting of 3 RGB Mako G125 GigE cameras with Sony ICX445 CCD sensor, resolution 1292 X 964, 30 frames per second (fps) for markerless analysis and (ii) a calibrated motion capture system, the Optitrack Flex 13 Motion Capture system, 1.3 MP, 56° Horizontal FOV, 46° Vertical FOV, 28 LEDs, 8.33 ms latency, with 8 cameras acquiring at 100 Hz. With the motion capture system we acquired the 3D position of 22 infrared passive markers placed on the body of the participants following the Davis protocol [Davis III et al., 1991].

The study was conducted according to the guidelines of the Declaration of Helsinki, and approved by the Institutional Review Board of the Department of Informatics, Bioengineering, Robotics and Systems Engineering (DIBRIS), University of Genoa, Genova, Italy (protocol code CE DIBRIS - 008/2020 approved on 18/05/2020). All the participants involved in the study signed an informed consent form.

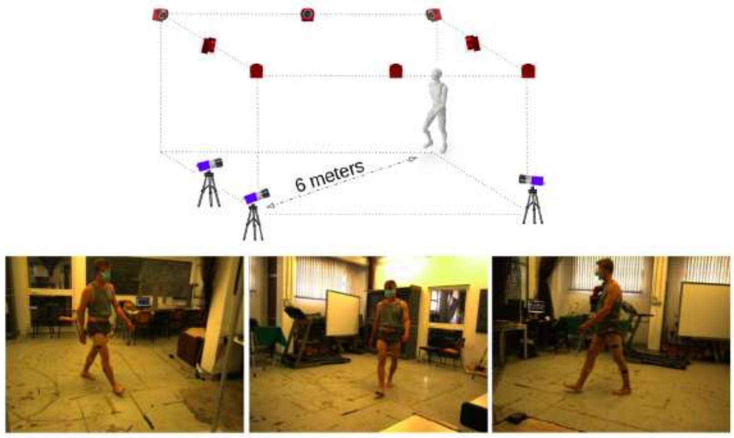


Figure 6.1: Setup adopted for data acquisition. In the upper panel the sketch of the setup with the position of the 8 infrared (red) and 3 RGB (blue) cameras. In the lower panel the three view points of the RGB cameras.

people with multiple sclerosis (3d). We acquired the gait of 30 Multiple Sclerosis (MS) participants (22 females, 8 males, mean age ± standard deviation: 39 ± 11 yeas old, mean disease duration ± standard deviation: 8.7 ± 7.5 years, mean Expanded Disability Status Scale score (EDSS) [minimum, maximum]: 2.6 [0, 6]). The set up included only the multi-view camera system composed by 3 synchronized RGB Mako G125 GigE cameras with Sony ICX445 CCD sensor, resolution 1292 X 964, 30 frames per second (shown in blue in Figure 6.1 . We did not include the motion capture system to reduce the obtrusiveness of data acquisition for MS participants. In the same conditions, we acquired also the gait of 30 healthy participants (age and sex matched with the distributions of MS patients).

Due to time limits, we have not analyzed this dataset yet. Nonetheless, we are planning to do that in the next months.

## 6.4 Methods

### 6.4.1 *2D analysis*

In this case our analysis is based on the lateral views, as they contain all the landmark points needed to describe the motion on the sagittal plane: hip, knee, ankle and foot. For this analysis we adopt the dataset of stroke survivors (only one lateral viewpoint). Each video of a single walk lasts about 3 seconds. We consider at least three trials for each leg. This allows our markerless approach to incorporate information of both the impaired and the unimpaired leg. Here we summarize the methods we adopt to extract gait parameters.

1. **Landmark points detection and filtering**. In this case, we train two networks: one for the left leg and one for the right one, depending on the direction of the gait in the video. We proceed in this way in order to

minimize the DeepLabCut (DLC) detection error. To train the networks we randomly select 5 frames for two out of three videos of each subject and we manually label the points of interest. During inference, in each video frame, we first detect 2D key-points corresponding to the anatomic joints considered in the reference study (see the example in Figure 6.2). Once the key-points are detected on all the video frames, we track them over time computing key-point sequences and we filter them (following the steps described in Section 4.3).

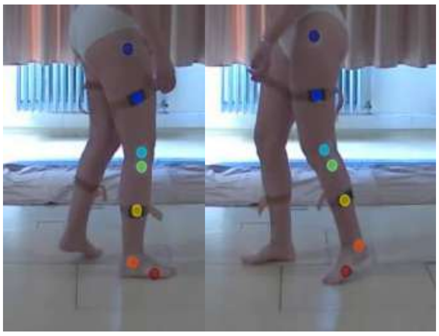


Figure 6.2: Example of the extraction of landmark points coordinates (x and y) in the image plane for left and right leg.

Once we obtain the 2D points of interest in the image plane reference system, with a one camera system we can not derive the original 3D information. For this we rely on an approximation, observing that the joints we are considering live, with a good approximation, on a 3D virtual plane. Lacking an appropriate calibration procedure, we estimate the plane-to-plane transformation (or homography) relating the detected image points $(x, y)$ with the $(X, Y)$ coordinates of the corresponding marker-based 3D locations (where we discard the third component corresponding to depth information which we cannot determine in the image). This transformation also takes care of the change of coordinates (pixels to mm) [Hartley and Zisserman, 2004]. In this way we are compensating for the absence of an appropriate calibration procedure during data acquisition.

2. **Gait cycle detection.** The first step before computing the gait spatio-temporal and kinematic parameters is the detection of the gait cycles. One gait cycle is defined as the period that starts with the heel strike (first instant when the heel hits the ground) of one foot and ends with the following heel strike of the same foot. A typical approach for automatic gait cycle detection is to analyze the speed of the heel (or a foot keypoint). The cycle starts when the heel hits the ground; in this time instant the speed of the heel is close to zero. It remains close to zero for the entire stance phase (the phase starting with the heel strike and ending when the foot leaves the ground) and it goes up in the swing phase (complementary to the stance phase). Then, the swing phase ends

| Parameters | Description |
|---|---|
| Stride Length | Distance (in meters) walked during a gait cycle. |
| Stride Time | Time (in seconds) necessary to walk one gait cycle. |
| Stance Phase | Percentage of the gait cycle during which the foot of interest is touching the ground (from heel strike to toe off). In healthy subjects it accounts for 60% of a single gait cycle for both legs. |
| Swing Phase | Percentage of the cycle during which the foot that we are considering is not touching the ground. In healthy subjects it accounts for 40% of a single gait cycle for both legs (complementary of the stance). |

Table 6.2: Spatio-temporal parameters for 2D gait analysis.

and the heel speed goes close to zero again. This first time instant where the speed is close to zero is the one representing the end of the current gait cycle and also the start of the following one. From a practical point of view, since in this case we do not have the position of the heel, for each participant, we select the ankle trajectories and we filter the $(x, y)$ coordinates with a low pass filter (Butterworth, 4-th order, $3Hz$ cut off frequency) in order to get rid of any noise in the signal. This is specially important because we then compute the derivative of the position and we need the result to be smooth (derivative generally increases noise, if the input signal is noisy). It is worth mentioning here that this filtered signal with 3 Hz cut off frequency is only created for gait cycle detection. To further process the signal in later steps, we go back the original signals and proceed with a different filter. Starting from the filtered signals, we combine the different coordinates and we obtain the total displacement.

3. **Parameters extraction.** We computed the spatio-temporal parameters described in Table 6.2 as reported in [O'Connor et al., 2007].

   Furthermore, we compute the relative joint angles from the spatial coordinates of each joints [Borghese, Bianchi, and Lacquaniti, 1996]. The elevation angle of a limb segment is defined in the sagittal plane as the orientation of the segment with respect to the vertical and to the walking directions, and it is positive for the forward direction:

$$\sigma_i = arctan(\frac{x_d - x_p}{y_p - y_d}) \tag{6.1}$$

   where $x$ is the forward direction, $y$ is the vertical direction, $d$ and $p$ denote the distal and proximal endpoint of the segment. We compute the elevation angle of thigh, shank and foot.

4. **Statistical analysis.** For the analysis of the changes of the elevation angles as function of the gait cycle we used the statistical parametric mapping (SPM) approach [Friston et al., 2007] which is used to analyse statistical differences among continuous curves [Pataky, Robinson, and Vanrenterghem, 2013]. We perform 1D paired t-test with $\alpha$=0.05 [Pataky, Robinson, and Vanrenterghem, 2013] to compare curves of the elevation angles computed with the two different techniques (marker-based and our markerless approach) both for impaired and unimpaired legs.

### 6.4.2 *3D analysis*

Here we summarize the methods to analyze the 16 healthy participants dataset. Given the two different types of data (marker and video), some of the steps are different for the two approaches.

3D KEYPOINTS EXTRACTION: MARKER DATA The motion capture system reconstructs the trajectories of the markers in the 3D reference system, starting from 8 infrared cameras. To perform the motion analysis, we need to add a feature matching and tracking step. The process of *sorting and tracking* the markers is a standard procedure performed after data acquisition with a motion capture system. The software Motive [*Motive: optical motion capture software*] provided with the Optitrack motion capture system automatically perform this procedure by applying a model of the human body indicating the position of the markers (Figure 6.3 A), defined by the user. However, in cases of markers occlusions or presence of disturbances as reflexes, this procedure required the manual intervention of the operator, resulting in a time consuming procedure. This workload emphasizes one drawback of the marker-based motion capture system. At the end of this process, we obtained 16 matrices $Pmarker^j$ with $j = 1, ..., 16$ indicating the index for each participant, of shape $22 \times 3 \times M_j$ (22 representing the number of markers, 3 the $(X, Y, Z)_m$ markers' coordinates in the 3D space in the markers reference system ($_m$) and $M_j$ the number of samples for the acquisition of the $j$-th participant).

3D KEYPOINTS EXTRACTION: VIDEO DATA In the markerless approach, the RGB cameras produce video streams acquired from multiple-views. To obtain the 3D points, we need to detect semantic features in 2D and then triangulate them in 3D. The resulting 3D points are in this way already tracked, since each one of them is associated with a semantic meaning. Thus, the aim of this step is the detection of the 3D positions of keypoints that represent the analogous of markers and that can be adopted to perform gait analysis. To perform this step we rely on a 2D pose estimator to detect the positions of the keypoints in the image planes of each viewpoint and then we reconstruct the positions of each keypoint in the 3D space with geometric reconstruction.

For this task we rely on AdaFuse [Zhang et al., 2021], described in Section 4.4. AdaFuse is a deep learning-based algorithm that allows to accurately detect the positions of specific keypoints in the image plane and leverages classical stereo vision algorithms [Andrew, 2001] to reconstruct the 3D positions of the detected keypoints in 2D image planes. The pretrained 2D backbone models provided by AdaFuse authors [Zhang et al., 2021] do not consider keypoints on the feet. Unfortunately, these keypoints are necessary for gait analysis to compute the kinematic parameters related with the ankle joint (*i.e.*, ankle dorsi-/plantar-flexion). For this reason it is necessary to train the model with new data that included also keypoints on the feet. To effectively train our model, we need a dataset with the 3D ground truth positions of each keypoint. Among the public available datasets (well summarized in [Kwolek et al., 2019]), we rely on the Human3.6m dataset [Ionescu et al., 2013].

This dataset includes both a multi-view RGB camera system (with 4 cameras) and a motion capture system with infrared cameras and 32 markers (see [Ionescu et al., 2013] for further details). Leveraging *Vicon's* skeleton fitting procedure [*Vicon*] and by applying forward kinematics [Ionescu et al., 2013], the 32 markers are projected into the 2D image planes of the different viewpoints resulting in 2D keypoints ground truth (see Figure 6.3B). Human3.6m is our best option, even if it presented drawbacks for our main goal. For example, the feet sometimes get rather blurry, mainly in the swing phase where one foot moves quickly. Additionally, the background carpet, under the lighting condition during the recordings, has color similar to the skin, so contrast decreases to a low level, where even for human observers would be hard to detect the keypoints precisely.

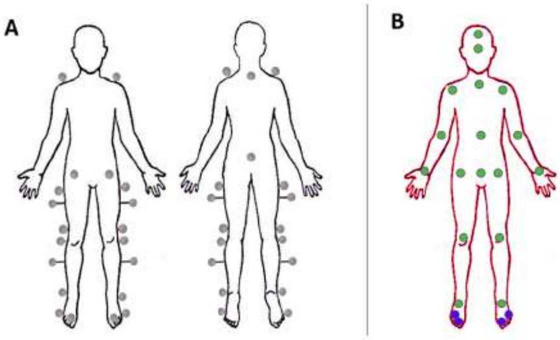


Figure 6.3: (A) Frontal and back views of the positions of the 22 markers positioned in this study according to the Davis protocol [Davis III et al., 1991]. Specifically they were placed on the spinal process of C7 and on the spinal process of the sacrum (both visible in the back view) and bilaterally on: the acromion, the Anterior Superior Iliac Spine (ASIS), the greater trochanter, the middle between the greater trochanter and the lateral epicondyle of the femur (with bars 5*cm* long), the lateral epicondyle of the femur, the fibula head, the middle between the fibula head and the lateral malleolus (with bars 5*cm* long), the lateral malleolus, the first metatarsal phalangeal joint, the fifth metatarsal phalangeal joint on the lateral aspect of the foot. (B) 2D keypoints (green and blue dots) considered in this work from the Human3.6 dataset. The two blue keypoints in each foot are highlighted because they are those not included in [Zhang et al., 2021] and that we added them in our training.

We fine tune the Adafuse architecture in two steps:

1. **2D backbone**. We first focus on the 2D backbone network creating independent probability maps of the keypoints in Figure 6.3B for each separate input image and we fine tune the Pose ResNet-152 [Xiao, Wu, and Wei, 2018] pretrained on the COCO dataset [Lin et al., 2014b]. We fine tune the network adopting a subset of the Human3.6m training images, *i.e.*, we considered one image every 20 frames. This allow us to have

a training set with a reasonable number of frames sufficiently different from one another.

2. **Full architecture**. Then we focused on the fusing network which refines the maps with the help of the neighboring views. This second part of the AdaFuse architecture should not be trained separately (as mentioned in [Zhang et al., 2021]), but jointly with the 2D backbone. Thus, we initialize the first part (2D backbone) with the weights obtained with the fine tuning described above and the fusion network with random weights. In this case, the inputs of the process are not just single images (as for the previous step), but a group of images representing the same time instant but coming from different viewpoints. Additionally, we input the calibration information for the group of images containing intrinsic and extrinsic parameters. These parameters are not used by the neural network itself, but in an immediate post-processing step which computes the 3D poses at the end. The target and output for the neural network is a group of probability maps corresponding to the input images.

The motivation behind the choice of Adafuse is its high level of accuracy reached thanks to the refinement of the 2D keypoints estimates performed before the geometric 3D reconstruction step.

After training, we apply the model to our dataset for retrieving the 3D positions of the Human3.6m keypoints highlighted in Figure 6.3B. Since Pose ResNet-152 requires as input also a bounding box localising the person in the image plane for each frame composing the videos, we rely on CenterNet [Zhou, Wang, and Krähenbühl, 2019] – a state-of-the-art object detector – to create these bounding boxes for our dataset. Thus, we input to the model the 3 images coming from the 3 different viewpoints at the same time instant $t$, the bounding boxes and the intrinsic and extrinsic parameters retrieved with cameras calibration. Firstly, we obtain the probability maps for different keypoints at the same time instant (Figure 6.4A) and, then, the final 2D locations of each keypoint (Figure 6.4B). At the end, the final output is a vector of shape $21 \times 3$ (21 keypoints with the corresponding $(X, Y, Z)_v$ coordinates in the 3D space in the camera reference system $_v$), with $j = 1, ..., 16$ representing the number of videos (*i.e.*, the number of participants) and $t = 1, ..., N_j$ the index for the number of frames composing the $j$-th video ($N_j$ is the total number of frame for the video of the $j$-th participant). At the end of this step, we end with 16 matrices $Pmarkerless^j$ of shape $21 \times 3 \times N_j$ (see Figure 6.4C for examples of 3D poses).

3d trajectories analysis. The 3D trajectories of the keypoints obtained with marker-based and markerless systems ($Pmarker^j$ and $Pmarkerless^j$ respectively) are processed in the same way to extract quantitative parameters describing the gait of each participant. In particular, we perform the following steps.

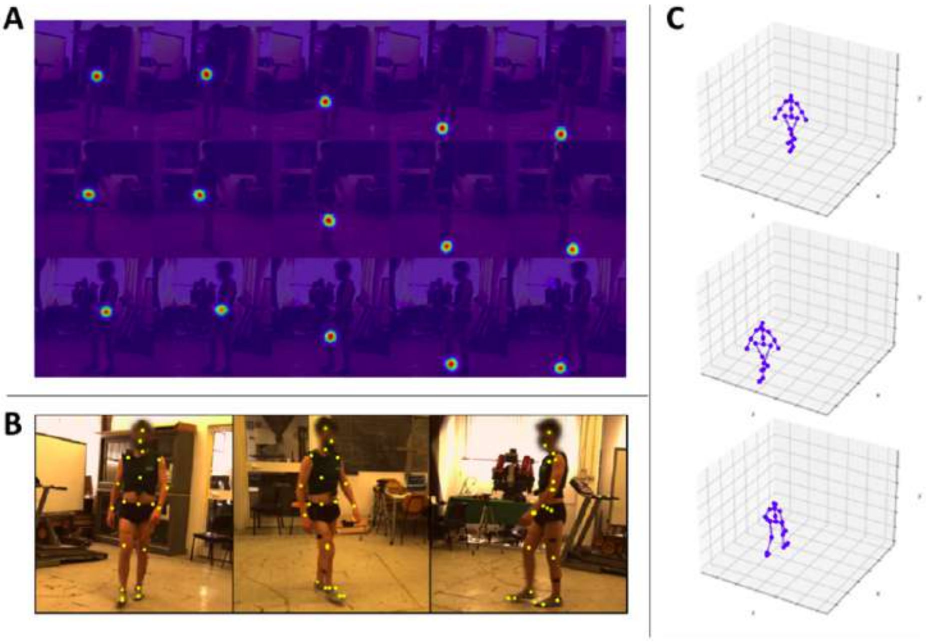


Figure 6.4: (A) Examples of the detected probability maps ($U_t^{j,i,l}$) for the $j$-th participant at a specific time instant $t$. The rows represent the 3 different viewpoints $i$. Each column represents a different keypoint $l$ detected on the right leg (from left to right: hip, knee, heel, toe). (B) Examples of the detected keypoints (yellow dots) on the three views composing our dataset. (C) Examples of the final 3D skeleton of the video pre-processing.

1. **Gait cycle detection.** We follow exactly the same procedure as for the 2D analysis, but in this case we use the heel trajectories to detect the gait cycles.

2. **Spatio-temporal and kinematic parameters extraction.** The 3D coordinates trajectories of each keypoint during the gait cycles are low pass filtered (Butterworth, 4-th order, $12Hz$ cut off frequency) [Whittle, 2014]. Starting from the heels' markers trajectories, we extract the spatio-temporal parameters that characterize the human gait. In particular, similarly to the 2D analysis, we compute the parameters reported in Table 6.2 and we add: (i) *stride width*: the distance (in meters) between the right and the left foot across the cycle; (ii) *speed*: mean speed of the center of mass of the body during the cycle. To estimate the joint angles during the gait cycle we rely on the open source software Opensim [Delp et al., 2007]. Opensim is commonly adopted to estimate joint angles during gait analysis because it allows associating the detected keypoints/markers to human biomechanical skeleton models and analyze the kinematics and the relative muscular activation. In this work we adopted the Rajagopal Model [Rajagopal et al., 2016] (shown in Figure 6.5), a full body musculoskeletal model for dynamic simulations of human movements, widely used in gait analysis applications. In Opensim, two tools are specifically designed to solve our problem, *Scaling* and *Inverse Kinematics*. The first is

adopted to scale a generic skeleton model to fit the input markers/keypoints data. The latter is used to simulate the motion of the skeleton and to estimate the joint angles for each gait cycle for each subject. Following the steps explained above, we extract the joint angles for the central gait cycle of each trial (for a total of 20 gait cycle) for each participant involved in the study both with marker-based and markerless systems.

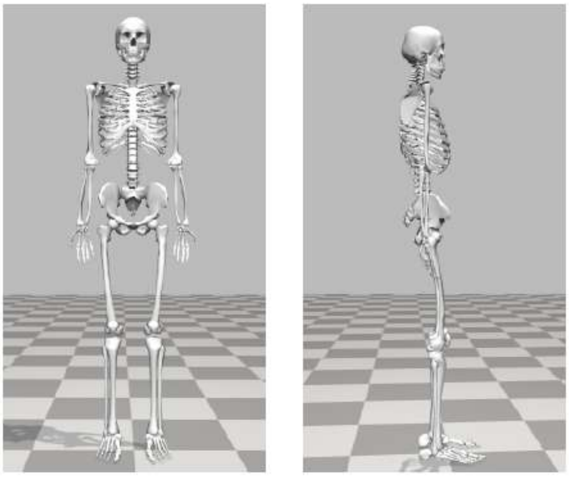


Figure 6.5: Rajagopal skeleton model [Rajagopal et al., 2016].

3. **Statistical analysis.** As in the 2D analysis, to compare the time profile of the joint angles during the gait cycle obtained with the markerless and the marker based gait analysis we use the statistical parametric mapping method, which is specifically designed for continuous field analysis [Pataky, Vanrenterghem, and Robinson, 2015]. We apply this method to the 1-D spatio-temporal variables describing the variations of the joint angles during the gait cycle by using the open source software spm1d [Pataky, Vanrenterghem, and Robinson, 2015]. Specifically, we perform a one dimensional paired t-test. We test the following null hypothesis: "there is no statistical significant differences between the gait angles obtained with our markerless approach and the gait angles obtained with the gold standard marker-based system". The alpha level indicating the probability of incorrectly rejecting the null hypothesis is set at 0.05. To follow a conservative approach, *i.e.*, to maximise the possibility of finding statistically significant difference between the results obtained with the two methods, we do not apply Bonferroni corrections. Notice that the application of corrections for multiple comparison would decrease the probability to find significant differences between the single point curves. Furthermore, we compare the spatio-temporal parameters obtained with the two methods with a paired t-test. Again, statistical significance is set for all statistics at the family-wise error rate of $\alpha = 0.05$.

## 6.5 Results: 2D analysis

We first conduct a quantitative analysis with the aim of verifying the similarity between measures obtained with marker-based technique and our markerless method. In Figure 6.6 we can see the coordinates for a gait cycle of each landmark point for one subject randomly selected. We report both legs to show that the method is robust for both the impaired and the unimpaired leg.

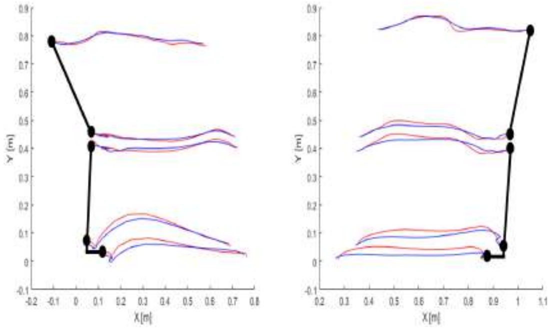


Figure 6.6: Coordinates in meters for landmark points for a randomly selected subject during a gait cycle. On the right we show the left leg (impaired) and on the left the right leg (unimpaired). In red we can see the evolution of each coordinate computed with the marker-based techniques, in blue those obtained with our markerless approach.

In Figure 6.7 we show the mean error of our estimates expressed in centimeters and computed as the euclidean distance of each point estimated by our markerless approach with respect to the marker-based ones that we use as a ground truth. The mean is computed over all the available data. We notice that overall the distances are small, in the order of centi/millimeters. They are sensibly lower for the first three landmark points (hip and the two points on the knee) and higher for the ankle and the foot.

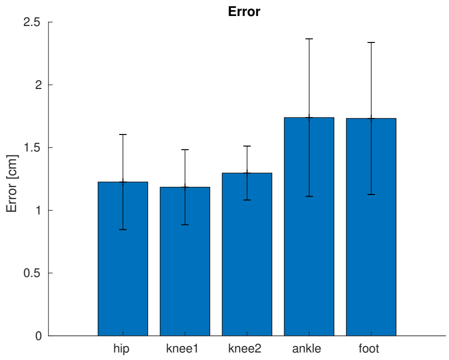


Figure 6.7: Mean euclidean distance between positions estimated by a marker-based system and our markerless approach. Error-bars indicate the standard deviation.

Then, we compute the parameters described in the Table 6.2 and we verify that the parameters computed with our method are coherent with the marker-based ones, they are included in the standard deviation of the marker-based measures as shown in Figures 6.8 and 6.9.

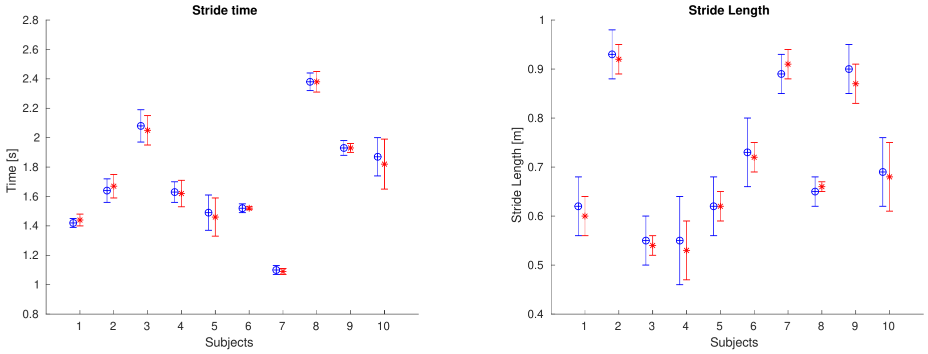


Figure 6.8: Mean and standard deviation for stride time on the left and for stride length on the right computed with marker-based technique (red) and markerless one (blue).

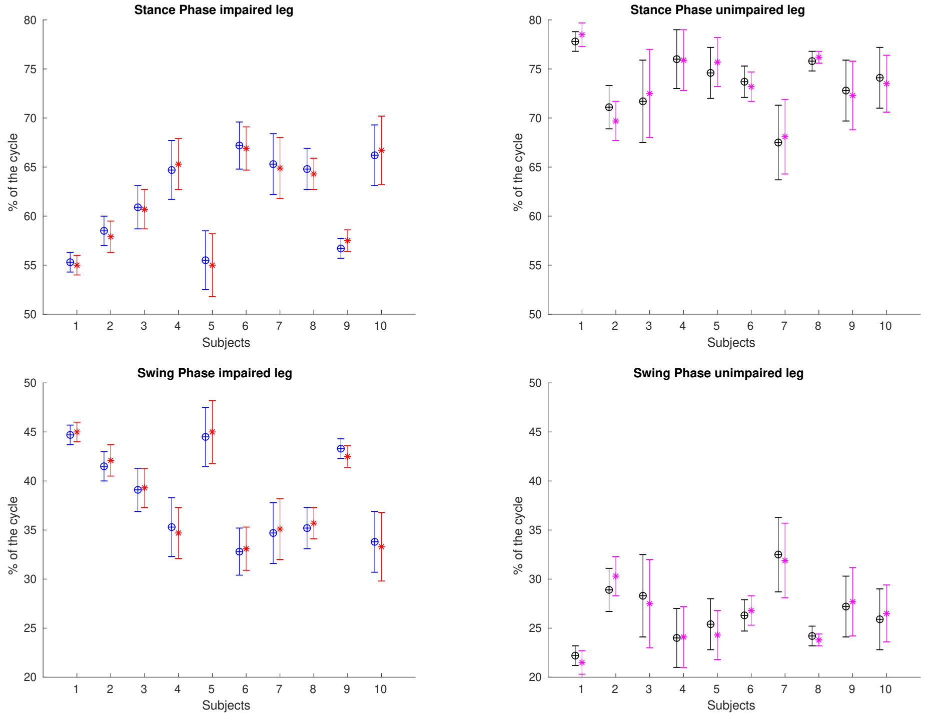


Figure 6.9: In the upper part is reported the stance phase (% of the gait cycle) for each subject and in the lower part the swing phase. In red (impaired leg) and in magenta (unimpaired leg) we show the mean and the standard deviation computed using the marker-based technique taking in account three trials; in blue (impaired) and black (unimpaired) we show the mean and the standard deviation computed with our method.

In Figures 6.10, 6.11 and 6.12 we show the results of the statistical analysis. First of all, in the upper part of Figure 6.10, we can see thigh, shank and foot elevation angles computed for the impaired leg with the marker-based technique (red) and our system (blue). The colored area represents the Standard Error (SE). In the lower part, we show the t statistic as a function of the percentage of gait cycle. As we can see there are not significant statistical differences between the curves. Figure 6.11 displays the same analysis for the unimpaired leg. In both cases the biggest difference although not significant is in the foot elevation angles. This difference is coherent with the result shown in Figure 6.7: a higher error in the coordinates of ankles and feet causes a bigger difference between the elevation angles. Finally, Figure 6.12 highlights the differences between the elevation angles of impaired and unimpaired leg computed with our markerless system.

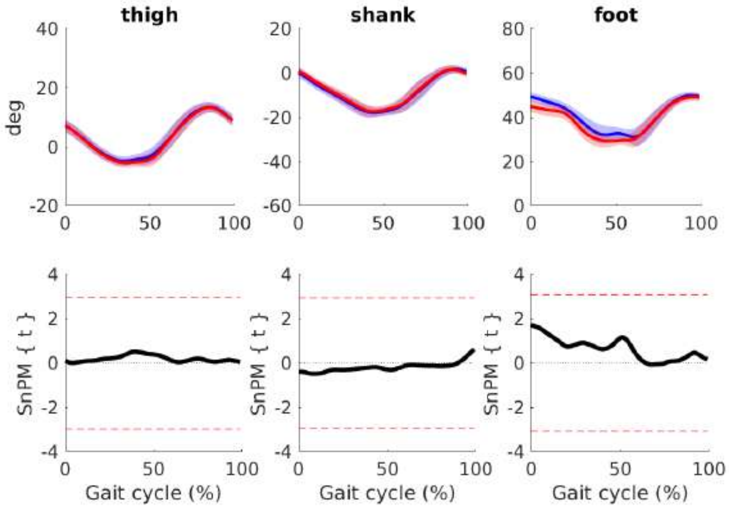


Figure 6.10: The first row shows the elevation angles for the impaired leg computed with the marker-based system (red) and our method (blue) averaged across all the subjects. The colored area is the corresponding standard error (SE). The second row shows the t statistic for the comparison of marker-based and markerless technique as a function of the percentage of gait cycle. No statistical significant differences are reported.

## 6.6 Results: 3D analysis

### 6.6.1 *Architecture evaluation*

To evaluate the accuracy of our trained 2D backbone, we compute the *PCKh* for each keypoint (see Figure 6.13 for a qualitative result). As threshold value $r_{thr}$ we select a percentage of the head bone link for each participant (indicated by the *h* in *PCKh*). The following multiplication factors are chosen: 1 (*PCKh*@1), 0.75 (*PCKh*@0.75) and 0.5 (*PCKh*@0.5). Table 6.3 summarizes the obtained results.

The neural network indeed learns to detect also the new keypoints (toes and heels) with high accuracy. The *PCKh* for these keypoints is comparable to the

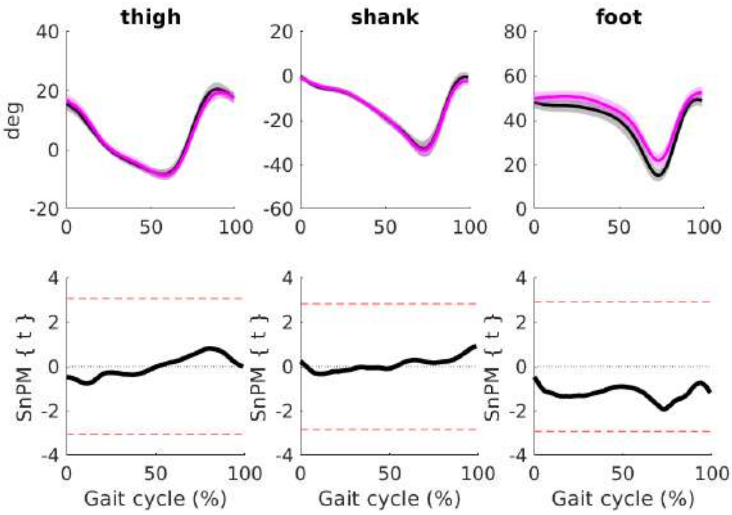


Figure 6.11: The first row shows the elevation angles for the unimpaired leg computed with marker signal (violet) and our method (black) averaged across all the subjects. The colored area is the corresponding standard error (SE). The second row shows the t statistic for the comparison of marker-based and markerless technique as a function of the percentage of gait cycle. No statistical significant differences are reported.

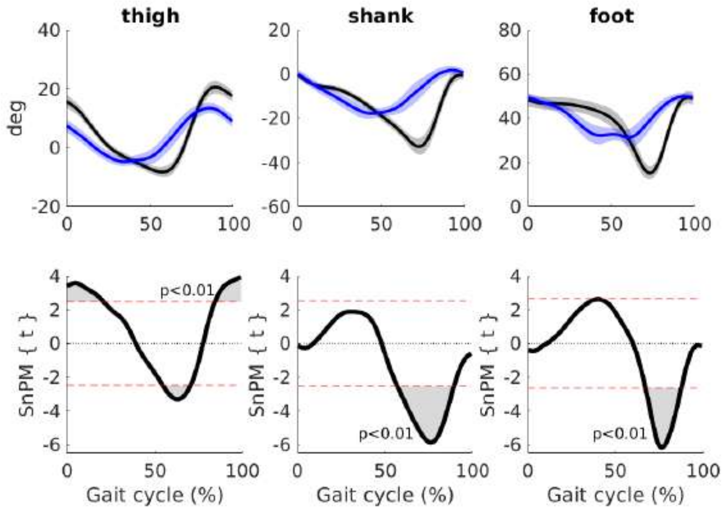


Figure 6.12: The first row shows the elevation angles computed with markerless technique for the impaired (blue) and the unimpaired (black) leg. The colored area is the corresponding standard error (SE). The second row shows the t statistic as a function of the percentage of gait cycle. Here we have significant differences (gray area) and the correspondent p value.

one of the other keypoints, and also to the results presented in other works (see for instance [Zhang et al., 2021]).

To evaluate the accuracy of the full architecture we compute the MPJPE across all the detected keypoints and we obtain an error of 23.65 millimeters, again comparable to the one obtained in Zhang et al., 2021 (*e.g.*, 19.5 millimeters on the same dataset, but with fewer keypoints – the feet were excluded)

| Keypoints | PCKh@1 | PCKh@0.75 | PCKh@0.5 |
|---|---|---|---|
| head | 96.3 | 95.8 | 95.2 |
| root | 96.6 | 95.6 | 94.8 |
| nose | 96.1 | 94.3 | 87.2 |
| neck | 96.1 | 89.3 | 77.2 |
| right shoulder | 93.4 | 87.4 | 66.7 |
| right elbow | 89.1 | 79.8 | 70.7 |
| right wrist | 85.5 | 78.6 | 67.8 |
| left shoulder | 95.2 | 88.9 | 72.7 |
| left elbow | 90.6 | 82.2 | 77.1 |
| left wrist | 85.0 | 78.7 | 70.0 |
| belly | 94.2 | 80.7 | 72.0 |
| right hip | 96.0 | 87.6 | 73.2 |
| right knee | 93.4 | 85.5 | 76.2 |
| right foot1 | 91.6 | 79.7 | 61.4 |
| right foot2 | 92.3 | 84.5 | 68.6 |
| right foot3 | 89.2 | 77.3 | 63.0 |
| left hip | 95.8 | 85.1 | 72.1 |
| left knee | 92.4 | 79.9 | 66.7 |
| left foot1 | 90.3 | 75.9 | 52.8 |
| left foot2 | 91.7 | 83.4 | 67.7 |
| left foot3 | 88.7 | 78.4 | 64.4 |

Table 6.3: Accuracy (%) of the 2D backbone, *i.e.*, the percentage of corrected keypoints (PCKh) considering different threshold values: 1, 0.75 and 0.5 times the head bone link (*PCKh*@1, *PCKh*@0.75 and *PCKh*@0.5 respectively).

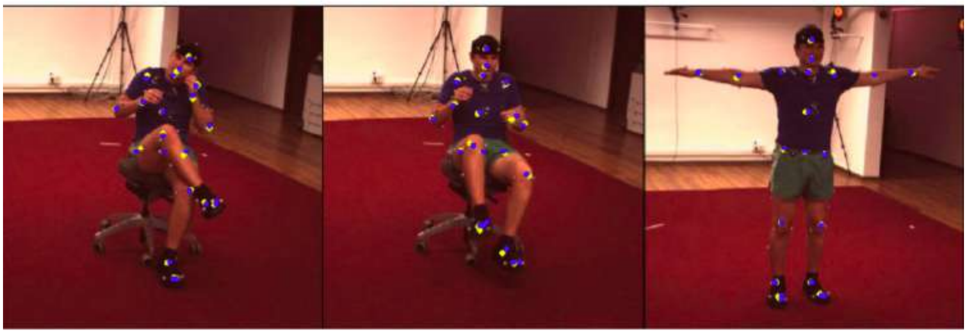


Figure 6.13: Examples of the keypoints detected with our model (yellow dots) with respect to the ground truth (blue dots).

and also comparable with the error obtained in the best performing recent works about 3D pose estimation (between 19 and 30 millimeters) [He et al., 2020; Li et al., 2021; Reddy et al., 2021; Shan et al., 2021].

### 6.6.2 *Joint angles and spatio-temporal parameters*

We compute the spatio-temporal parameters described in the previous section for each gait cycle for every participants and we compare the results obtained with the two different techniques. In Table 6.4 we report the mean and the standard deviation across all the subjects. As we can notice the parameters obtained with our markerless pipeline are similar to the ones extracted with the gold standard marker-based technique (as highlighted by the statistical comparisons: all *p-values* $> 0.050$, see Table 6.4 for more details).

| | **Stance phase (%)** | **Swing phase (%)** | **Stride length (m)** | **Step width (m)** | **Stride time (s)** | **Speed (m/s)** |
|---|---|---|---|---|---|---|
| **Marker** | 59.2<br>± 2.6 | 40.8<br>± 2.6 | 1.35<br>± 0.11 | 0.10<br>± 0.02 | 1.13<br>± 0.02 | 1.31<br>± 0.10 |
| **Markerless** | 59.6<br>± 3.1 | 40.4<br>± 3.1 | 1.40<br>± 0.21 | 0.12<br>± 0.02 | 1.11<br>± 0.04 | 1.35<br>± 0.16 |
| **p-values** | 0.644 | 0.644 | 0.474 | 0.132 | 0.291 | 0.341 |

Table 6.4: Spatio-temporal parameters computed with marker-based and markerless systems, and statistical results of the comparison between the two methods (last row). We report the mean ± the standard deviation of each parameter. The stance and swing phases are reported in % with respect to the whole gait cycle; stride length and step width and expressed in meters (m); stride time in seconds (s) and the speed in meters per second (m/s).

We compare the joint angles obtained by our markerless approach to those obtained with the marker-based method. We select the following meaningful angles: hip flexion/extension, knee flexion/extension, ankle dorsi-/planta-flexion, hip ab-/ad-duction and pelvis tilt. Figure 6.14 shows the mean and the standard deviation of the angles previously mentioned across all the participants (black: marker-based, red: markerless) and the results of the paired t-test. No statistical differences are found between the two techniques with the exception of a slight underestimation of the knee flexion and the ankle dorsiflexion angle between the 70% and the 80% of the gait cycle (during the swing phase, see gray areas in the paired t-tests in the right column of Figure 6.14 in correspondence of these two angles). Note that those statistical differences are not robust to multiple comparison, *i.e.*, applying a Bonferroni correction the differences are not below the threshold for significance.

## 6.7 Discussion

The results obtained with our markerless system present differences with respect to the ones obtained with the gold standard, especially during the swing phase in knee and the ankle elevation and joint angles.

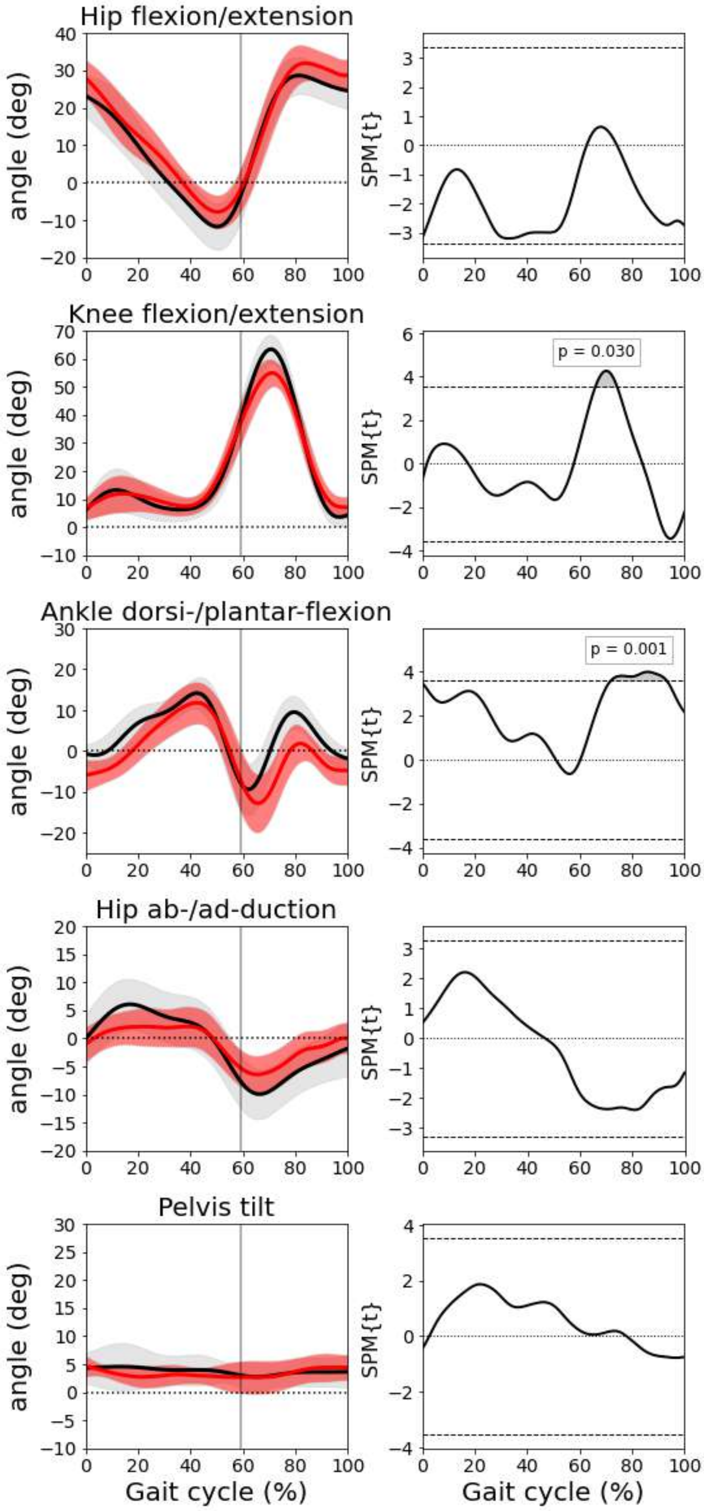


Figure 6.14: Left column: joint angles (mean and std). From top to bottom: hip flexion/extension, knee flexion/extension, ankle dorsi-/planta-flexion, hip ab-/ad-duction and pelvis tilt. In black the results obtained with the marker-based system and in red with the markerless pipeline. Right column: results of the correspondent paired t-tests.

In the 2D analysis, these differences are not statistically significant. While, in the 3D analysis, these differences are statistically significant, but they seem to be small. Nonetheless, this limitation should be accounted and further investigated when adopting this markerless pipeline to detect and monitor abnormal motion patterns in people with orthopaedic injury or neurological diseases.

If we focus on the errors related to the knee and the ankle elevation and joint angles during the swing phase, we can observe that they are mainly due to small errors in the detection of the feet keypoints. In fact, during the swing phase the foot moves quickly and the image tends to get blurry and it is difficult also for human beings to detect keypoints with high confidence. The immediate way to reduce the motion blur is to adopt RGB cameras with higher temporal resolution, meaning higher acquisition rate (fps) [Pueo, 2016]. In this way the motion blur will be reduced and, consequently, also the detection error will be lower.

The results highlighted with the 2D analysis open the possibility for a massive data analysis campaign, as often, in the clinical practice, videos are recorded for manual inspection, but they are never used for automatic analysis. The available considerable set of data has the potential to be analysed offline, with the goal of obtaining additional and statistically relevant information about the medical condition and how it evolves over time.

# 7

# Other Applications

In this chapter we present two additional applications of markerless human motion characterization. In particular, the first part of the chapter is related with the analysis of the motion of violin players. The work is done in collaboration with Marquette University (Milwaukee, WI, USA), Michigan State University (East Lansing, MI, USA) and the Music Institute of Chicago (Chicago, IL, USA). The aim of this part of the thesis is to quantitatively evaluate the error produced with our markerless pipeline with respect to a gold standard marker-based system. In the second part of the chapter we report the implementation of a video-based Body Machine Interface (BoMI) that makes it possible to move the computer cursor by moving body parts in front of a webcam.

## 7.1 Motion analysis of violin players

The characterization of motion patterns of violin players is a project carried out in collaboration with the NeuroMotor Control Laboratory of the Marquette University (Milwaukee, WI), the Department of Kinesiology of the Michigan State University (East Lansing, MI) and the Music Institute of Chicago and it was supported by the US National Science Foundation (Grant 1823889). The long term goal of this project is the study of motor learning and motor relearning: how people acquire motor skills and how these skills change with practice and experience. The specific aim within this thesis is the application of our implemented markerless pipeline and its quantitative comparison with a gold standard marker-based procedure in the analysis of a complex task like playing a music instrument. Adopting markerless methods in this application domain may provide significant benefits with respect to participant setup time and reduced invasiveness.

In order to evaluate the performance of our markerless approach, we acquire the kinematics of 58 violinists repeatedly playing a G scale arpeggio. We rely on a 3-view camera system. The choice of three viewpoints allows us to geometrically reconstruct the 3D information while reducing the numbers of self-occlusions, which are quite frequent in moving human bodies. As a gold standard reference, we employ a motion capture system (Optotrak 3020) with active markers placed on the violin and on the bow. To compare our implemented pipeline with the gold standard, since the two systems are not mutually calibrated, we compare Euclidean distances between pairs of 3D landmark points in the marker-based and the markerless approach respectively. The distributions of distances show that the measures computed with our markerless pipeline are very close to the one computed with the marker-based system (with an error on the pair-wise distances below 6 mm in at least 70% of the cases).

### 7.1.1 *Dataset*

The motions of 58 violinists of a wide range of ages and capabilities has been acquired during 5 days of the 2019 Summer Suzuki Institute organized by the Music Institute of Chicago. The violinists signed an informed consent form approved by the Marquette University Institutional review board.

The setup includes a multi-view camera system (3 RGB Mako G125 GigE cameras with Sony ICX445 CCD sensor, resolution 1292 X 964, 30 frames per second) and a motion capture system (Optotrak 3020, Northern Digital Inc., 100 Hz) with 6 active markers on the violin and 4 on the bow - see Figure 7.1. The RGB cameras have been calibrated in order to obtain intrinsic and extrinsic parameters.

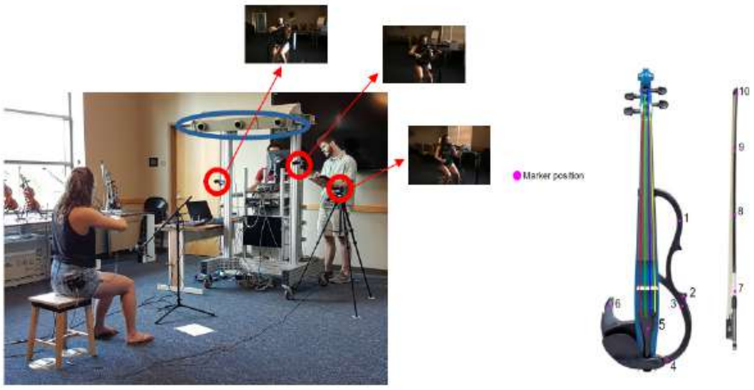


Figure 7.1: Data acquisition setup (video cameras marked in red, motion capture system marked in blue) and markers location on the violin and the bow.

Each violinist was asked to sit on a chair, at a fixed distance from the acquisition sensors, and to perform 50 repetitions of a 13-note arpeggio (G-scale arpeggio). Apart from that, no other instructions were given to the violinists: their pose is variable and clothing differs across participants. They all played the same instrument. To reduce the acquisition time and limit the discomfort of the volunteers, no markers were attached to the players.

### 7.1.2 *Methods*

We apply our proposed pipeline (described in Chapter 4 summarized by the following steps:

- **Landmark points detection**. In order to be able to compare the markerless analysis with the marker-based one, we focus on the positions of the infrared markers on the violin and on the bow. Moreover, since the long term goal of the work is the analysis of human motion, we consider also some human joints (the right shoulder, elbow, and wrist). Also in this case, we adopt a CNN-based semantic features detector and not a classical human pose estimation algorithm because we are not interested in the full-body pose, but we are instead interested in including semantic features that belong to objects (violin and bow). To fine tune our model, we consider 45 subjects and we randomly select 15 frames for each viewpoint (45 frames for each subject) and we manually label the position in the image plane of the landmarks: 4 markers on the bow, 5 markers on the violin (the 6th one is excluded because it is almost always occluded), 3 anatomic landmarks on the body - see Figure 7.2.

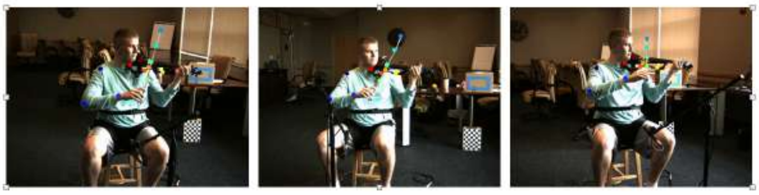


Figure 7.2: Landmark points detected in the image plane.

- **Filtering**. Once the network is trained, for each test frame $t$ it provides a set of 2D landmarks $p_{l,t}^{V} = (x, y, c)_{l,t}^{V}$, where $l \in \{$*shoulder, elbow, wrist, violin1, . . . , violin5, bow7, . . . , bow10*$\}$ characterises the semantic features and $V = \{1, 2, 3\}$ describes the view-point. For each video, the trajectories $(x, y)_{l,t}^{V}$ are filtered (first depending on the value of $c_{l,t}^{V}$ and then low pass filtered — see Section 4.3 for details).

- **3D reconstruction**. The semantic features extracted from the three viewpoints in each time instant, are combined to compute their corresponding points in the 3D space by means of multi-view geometric reconstruction (see Section 4.4 for details and Figure 7.3 for an example). In this case, we do not include the probability maps refinement step (*i.e.*, Adafuse) for the following reasons: (i) the landmark points we consider in this application are rarely occluded and their image texture variability is low (*i.e.*, it is easier to recognize them across different images); (ii) the position of the cameras (one close to another) and the position of the participants with respect to the cameras (see Figure 7.1) allow to increase the stability of the 3D reconstruction algorithm.

To show that recent 3D algorithms based on deep neural networks are not accurate enough for our general aim, we adopt also the method described in [Kocabas, Karagoz, and Akbas, 2019] and we compared the results obtained with the ones retrieved with geometric 3D reconstruction. The method described in [Kocabas, Karagoz, and Akbas, 2019] is a self-supervised learning method for 3D human pose estimation, which does not need any 3D ground truth and makes use of multiple viewpoints and epipolar geometry.

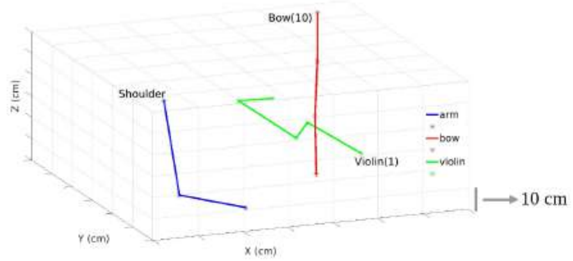


Figure 7.3: The reconstructed landmark points: the right arm in blue (shoulder, elbow and wrist), the 4 markers on the bow in red and the 5 ones on the violin in green.

### 7.1.3 *Results*

landmarks detection evaluation. Firstly, we process all 58 videos acquired through our trained model. In order to evaluate the quality of the detection of each landmark $l$ in the image plane, we analyze the confidence value $c_{l,t}^{V} \forall V$ returned by the model for each frame $t$. Both for training (45) and test (13) subjects we count for each landmark the number of frames where confidence is lower than 0.75; to identify the number of times that we can not trust the detection. Figure 7.4 shows the percentage - with respect to the total number of frames for each video - of cases detected with $c < 0.75$. Occlusions are included in this analysis.

As we can see from Figure 7.4 the % of frames with points in the bow and in the violin detected with low confidence is balanced in the test and training videos; these cases are mainly due to occlusions that can occur during the performance depending on the pose of the violinist with respect to the violin itself. Different considerations can be done for shoulder, elbow and wrist where the % of cases detected with low confidence is higher in test subjects. This is mainly due to the high variability of body landmarks, as confirmed in Figure 7.5: the figure compares the appearance variability of the *shoulder* landmark with *violin1*. The higher variability of shoulder, mainly due to different clothes worn by the volunteers, is apparent. Because of that, we may conclude body landmark points' detection would need to be trained on a larger dataset [Perez and Wang, 2017]. These points are not considered in our comparative analysis, as we do not possess a 3D gold standard for them.

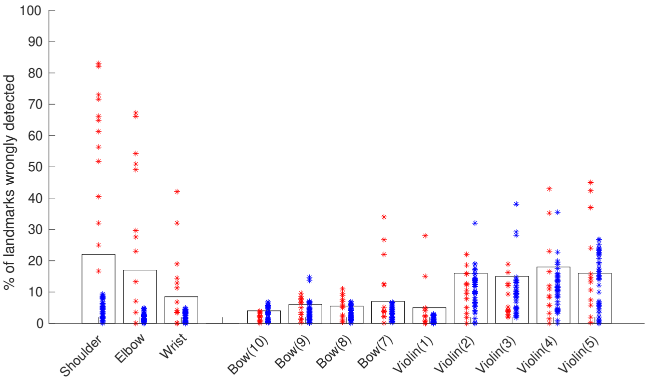


Figure 7.4: % of frames (y axis) in which the landmarks (x axis) are detected with a confidence < 0.75. In blue we show the results for the training subjects (45), in red the test ones (13). These results show that the the number of points detected with $c < 0.75$ in the violin and in the bow is balanced in training and test subjects.

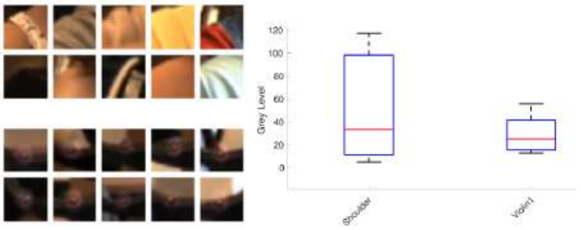


Figure 7.5: Left: examples of textures for *shoulder* (top) and *violin 1* (bottom). Right: average grey level variability.

marker vs markerless performance comparison. We evaluate the precision of the reconstructed 3D landmarks with the geometric approach. Since we do not posses the relative position between the cameras and the motion capture reference systems, we compare Euclidean distances in the 3D space between pairs of landmarks estimated by the marker-based method and the markerless one. Let $dM_t^k$ be the distances computed with the marker-based system for each $t$-th frame and for each $k$-th pair of markers, with $k = \{violin1 - violin2, violin2 - violin3, violin3 - violin4, violin4 - violin5\}$ as numbered in Figure 7.1. $dML_t^k$ are the corresponding markerless distances. We then evaluate the difference between the measures computed with the two techniques: $(dM_t^k - dML_t^k)$. A difference close to 0 mm means that our markerless measure is close to the gold standard. In Figure 7.6 we report the errors for 4 different pairs of points. As we can notice the majority of the samples has a very small difference. The distributions of the errors are approximately Gaussian centered in 0 and with a mean standard deviation of 6 mm.

markerless 3d reconstruction comparative analysis. As a final evaluation of the 3D reconstruction algorithm adopted, we compare its accuracy with the 3D reconstruction performed following [Kocabas, Karagoz, and Akbas, 2019]. Figure 7.7 reports a consistently larger error with respect to the

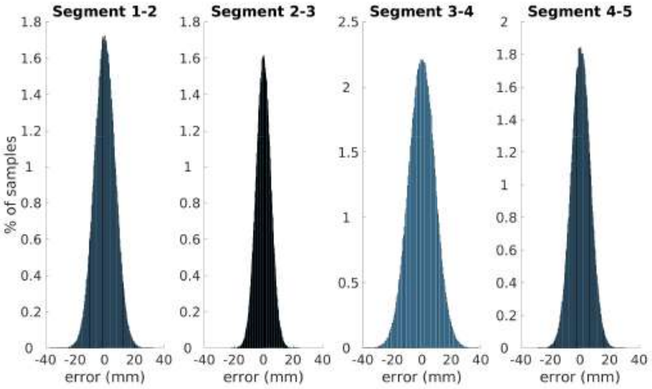


Figure 7.6: Difference in mm (x axis) between Euclidean distances computed with marker-based signal (gold standard) and the geometric 3D reconstructed markerless one. The 4 plots refer to 4 different distances between pairs of markers on the violin numbered as in Figure 7.1. The results show that the errors distribution are approximately Gaussian centered in 0 mm and with a mean standard deviation around 6 mm, meaning that the error is very low for the majority of cases.

geometric approach. This is confirmed by Table 7.1, where we report mean and standard deviation for both techniques with respect to the gold standard.

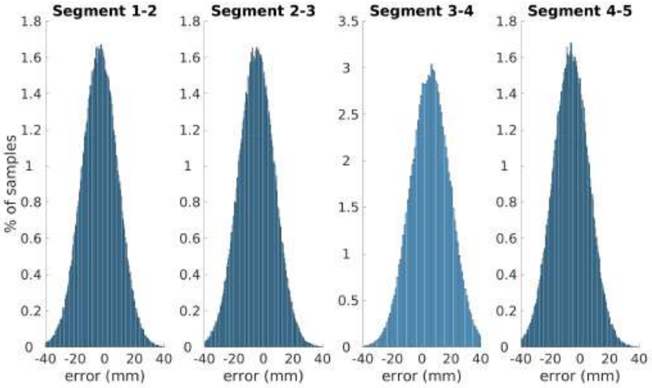


Figure 7.7: Difference in mm (x axis) between Euclidean distances computed with marker-based signal (gold standard) and the CNN-based 3D reconstructed markerless one. A comparison with Fig. 7.6 shows that the error with [Kocabas, Karagoz, and Akbas, 2019] is significantly higher.

| | 1-2 | 2-3 | 3-4 | 4-5 |
|---|---|---|---|---|
| Geom | 0.7 ± 5.7 | 0.6 ± 4.9 | 1.9 ± 8.3 | 0.8 ± 5.5 |
| CNN | 3.5 ± 9.1 | 4.8 ± 9.3 | 9.1 ± 12.2 | 5.3 ± 8.9 |

Table 7.1: Absolute value of mean ± standard deviation in *mm* of the error reported in Figure 7.6 and 7.7 for geometrical (Geom) and CNN-based (CNN) 3D reconstruction. The pairs of markers are numbered as shown in Figure 7.1.

### 7.1.4 *Discussion*

The results show that the error made by adopting the implemented markerless pipeline is in the order of a few millimeters (below 6 mm in at least 70% of the cases). This opens the possibility of adopting video-based markerless systems also in other applications fields such as motor learning, where a high level of precision is required. Furthermore, it is an example of how markerless techniques can help in the characterization of human movements in scenarios where it would be difficult (or even impossible) to adopt marker-based approaches, due to they intrusive nature.

## 7.2 Markerless body machine interface

Human motion disability is a global challenge affecting many people around the world [Cook and Polgar, 2014]. Disability can arise due to a birth condition, an accident or ageing. In this scenario, it is necessary to implement and investigate technologies that can improve the quality of life. Assistive Technologies (AT) [Cook and Polgar, 2014] and Positive Technologies (PT) [Grossi et al., 2020] emerge as promising approaches to address human disability. AT and PT are generic terms for all devices and services that enable the independence of individuals with cognitive and/or functional impairment by improving the conditions of their daily living activities and, consequently, their quality of life.

We investigate the problem of enabling individuals with motor disabilities (such as after Spinal Cord Injury - SCI) to recover their functional independence. We exploit the fact that, even after a severe injury, many individuals retain some movement, especially of their head and shoulders, that can be used to control external devices, such as a computer cursor. Our approach is based on the framework of Body-Machine Interfaces (BoMIs) [Casadio, Ranganathan, and Mussa-Ivaldi, 2012]. BoMIs convert high-dimensional body signals (*e.g.*, upper body kinematics, muscle activities) into lower-dimensional, latent, commands to operate an external device. As a result, BoMIs allow individuals with motor disabilities to overcome some of their impairments. The use of BoMIs has been tested in situations involving the control of a computer cursor [Rizzoglio et al., 2020], a powered wheelchair [Thorp et al., 2015] and quadcopters [Miehlbradt et al., 2018]. Typically, kinematic-based BoMIs rely on the use of sensors such as inertial measurement units (IMUs) [Pierella et al., 2017] or markers [Zhou and Hu, 2008] to record the body-movements of their users. For the specific task of cursor control, sensor-based techniques (electrooculargraphy (EOG), electromyography (EMG), IMU, gyro- and opto-sensors) are commonly adopted [Chen et al., 1999; Di Mattia, Curran, and Gips, 2001; Jeong, Kim, and Son, 2005; Kim et al., 2010]. However, such approaches might hinder the assistive capability of the interface, as sensors cannot be worn autonomously by the BoMI user. Moreover, in the case of training with the BoMI across multiple days, sensors need to be placed consistently so as to minimize the need of interface re-calibration.

A video-based markerless BoMI would potentially allow for a more natural user-friendly interaction with an external device, as it is less invasive and cheaper than a sensor-based one. However, applicability of interfaces that rely only on video-information has seen limited efforts. Javanovic et al. [Javanovic and MacKenzie, 2010] have proposed MarkerMouse, a computer mouse controller based on videos acquired from a webcam that detect a big marker placed on the head of the user. Fu et al. [Fu and Huang, 2007] and Betke et al. [Betke, Gips, and Fleming, 2002] have developed respectively hMouse and Camera Mouse, two video-based markerless mouse controllers that detect specific body features, enabling people with severe disabilities to comfortably access a computer, without body attachments.

With the application of our implemented markerless pipeline, we want to enhance these approaches combining new computer vision and deep learning techniques and the knowledge derived from the body machine interfaces. Specifically, we present a novel video-based markerless BoMI pipeline (available at `https://github.com/MoroMatteo/markerlessBoMI_FaMa`) to empower individuals with motor disabilities to independently control a computer cursor via shoulders and/or head movements without the needs of any sensors other than the computer webcam. Our procedure can be summarized by the following steps (see also Figure 7.8): (1) automatic acquisition of images of the user from a computer webcam; (2) detection of landmark points (*e.g.*, eyes, nose and shoulders) in the image plane; (3) encoding of the extracted signals to a lower dimensional (control) space via application of a dimensionality reduction (DR) algorithm; (4) handling of the graphic for providing BoMI users with visual feedback of the cursor via a computer monitor. We have evaluated our pipeline in terms of landmark points detection accuracy and overall speed, obtaining encouraging preliminary results. To the best of our knowledge, our method is the first involving a recent state-of-the-art pose estimation algorithm based on deep learning techniques.

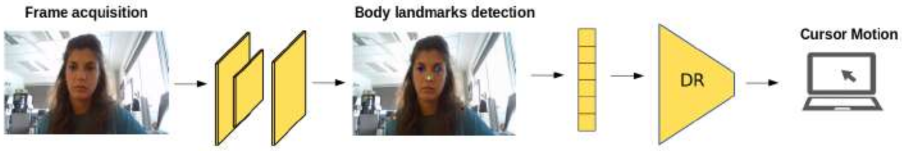


Figure 7.8: Summary of the BoMI pipeline. The image acquired by the computer webcam is fed through the trained network to detect the body landmarks. Then, a dimensionality reduction (DR) algorithm is applied to the landmarks' signal to obtain the coordinates of the computer cursor.

## 7.2.1 *Methods*

The pipeline adopted in this study slightly differs from the one described in Chapter 4. The main steps are reported below.

1. **Body landmarks detection**. The first step of the pipeline is the detection of the positions of body landmarks in the image plane. Since the long-term goal of the pipeline is to empower individuals with motor disabilities, specifically after cervical SCI (cSCI), regaining independence, we decided to focus on the tracking of the body parts whose mobility is most likely retained even after a high level cSCI - *i.e.*, shoulders and head (nose and eyes). In order to create an *ad hoc* model, for the first implementation [Moro et al., 2021b], we ask 40 healthy volunteers to freely move their head and shoulders for 30 seconds so as to comfortably explore their range of motion and to use a computer webcam or a mobile phone to capture a video of such body movements. Then, we randomly select 15 frames in 32 videos (80% of the total number of videos, for a total of 480 training samples), we manually label the points of interest for each sample and we fine tune the semantic features detector (DLC). As a result, the network learns to predict the position $(x, y, c)_t^l$ of each landmark point in the frame coordinate system, with $l \in \{righteye, lefteye, nose, rightshoulder, leftshoulder\}$. Figure 7.9 shows examples of detection result of this first step. Notice that we include videos with a wide variety of backgrounds, clothes and image dimensions so as to increase the robustness of the model. In the actual implementation presented at `https://github.com/MoroMatteo/markerlessBoMI_FaMa`, we adopt Mediapipe [Bazarevsky et al., 2020] as pose estimator in order to speed up the pipeline and allow our markerless BoMI to run in real time also in common laptop with low computational resources.

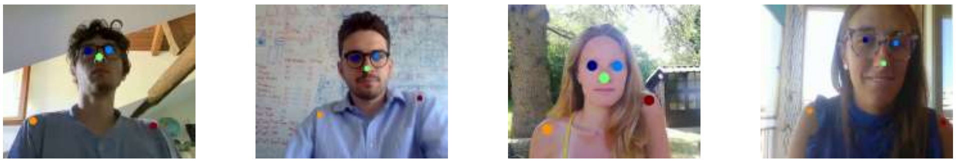


Figure 7.9: Extracted landmark points (shoulders, nose, eyes) for different subjects.

2. **Encoding body landmarks in the 2D cursor space**. After detecting the body landmarks, the second step consists of applying the BoMI forward map to obtain the $(x, y)$ coordinates of the computer cursor. Since the movements of the nose and the eyes are extremely correlated, we decide to exclude the latter. Thus, the 2D coordinates of shoulders and nose are organized as a 6D vector ($q$). The BoMI forward map is obtained by asking a volunteer to freely move his head and shoulders for 30s. Then, the DLC model previously trained is applied to the video to extract the vector of body landmarks $q$ for each frame. As a result, a matrix $Q$ containing the estimated coordinates of the landmark points for every frame is obtained. Next, we train a non-linear 2D variational autoencoder (VAE) [Kingma and Welling, 2013] on $Q$ to derive the 2D latent space in which the greatest amount of the body movements variance during calibration is explained. We choose a VAE among the possible methods

for dimensionality reduction (DR) (*e.g.*, Principal Component Analysis, *vanilla* AE) due to its ability to enforce a Gaussian distribution within its latent space. This would ensure a more uniform coverage of the 2D workspace with respect to that obtained training other DR models. To control the uniformity of the latent space, we introduce a scaling term ($\beta = 0.00025$) in the VAE cost function (see [Higgins et al., 2016] for more details). Then, we set the VAE encoder sub-network $E$ as the BoMI forward map. Thus, $E$ maps the 6D body landmark vector ($q$) into the x-y cursor vector ($p$):

$$p = E(q) + p_0 \tag{7.1}$$

The offset vector $p_0$ is chosen to match the origin of the body-space with a corresponding reference position of the cursor. Moreover, the resulting workspace is scaled to ensure full coverage of the computer monitor space [Casadio et al., 2010].

3. **Online video-based markerless BoMI**. Finally, we set up the online control of a computer cursor with the proposed BoMI. In order to set up the real-time interface, we develop a custom-coded Python script. The script has a multi-threaded architecture so as to handle three different processes: (i) capture the current frame from the computer webcam (via OpenCV library); (ii) forward pass the current frame through the DLC trained model to obtain the body-vector $q$; (iii) forward pass $q$ through the variational encoder $E$ to obtain the coordinates $p$ to control the cursor (see Figure 7.10). During the real-time pipeline, we feed the current webcam frame -read with OpenCV- and the weights of the DLC model to the Deeplabcut-live library [Kane et al., 2020], to obtain a real-time estimation of the body landmarks $q$. Finally, the encoder $E$ is applied to $q$ in order to obtain the cursor coordinates $p$ for the current frame. To speed up these online operations, we run the code in a computer with a 16 GB NVidia P5000 Quadro GPU.

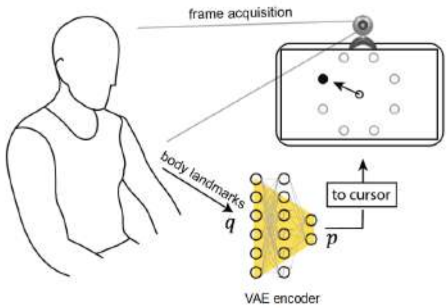


Figure 7.10: Scheme of the markerless BoMI for online cursor control.

### 7.2.2 *Results*

We present the results related to the real-time operation of the BoMI. Our goal is to assess whether our markerless BoMI could be run in real-time. Analyzing the frame recorded by the webcam with the DLC model is a computationally expensive operation, thus achieving a satisfactory frame rate during the online operation of the BoMI is not trivial. We have enrolled a naive unimpaired participant (age 27, male) to practice the online operation of the interface, informing him that he could move the cursor with the motion of his upper part of the body and nothing else. He has performed a reaching task, in which he is asked to move the cursor over a set of targets as quickly as he can. The position of the cursor and the targets are shown to the participants on a computer monitor. The targets are placed in four different locations, uniformly distributed along a circle. A reaching trial is considered successful after the cursor remains inside the target for 250ms. A total of 192 targets (48 trials per target location) are presented. The order of targets is pseudoranodmized so as each target location is not presented again before all 4 locations have been reached. The participant was immediately able to efficiently move the computer cursor over each target presented. Specifically, he completed all the 192 trials in just 13 minutes. Moreover, we completed all the analysis described before with a frame rate of 15 Hz, that allowed a continuous and efficient cursor control.

### 7.2.3 *Discussion*

This study delivered three main findings: (i) the fine-tuned DLC model was able to accurately predict the position of body landmarks on images with a variety of different backgrounds and clothes; (ii) such model can be adopted for the online detection of body landmarks; (iii) the proposed pipeline allowed a participant to efficiently and easily operate a computer cursor. For comparison with sensor-based approach, 10 healthy participants practicing cursor control with an IMU-based BoMI completed the same protocol in approximately 20 minutes. Note that this is a pilot study with the aim of exploring the feasibility of the real-time procedure, for this reason only one subject is involved.

The main goal of the study was to verify whether a video-based markerless BoMI could be used to operate a computer cursor in real-time. The step that took the most to be completed online was the application of the DLC model to estimate the landmarks position. Only a desktop computer with a powerful GPU would be able to complete this operation within an acceptable time frame during the online cursor control - as we achieved. However, our long-term goal is to improve the pipeline in order to be able to run it on any modern-day laptop, which does not have the same computational capability. This would dramatically increase the availability of the interface, thus broadening its impact as an assistive device.

The work described here is the first version of the markerless BoMI. In fact, we improved it with the following novelties (present in the software downloadable at `https://github.com/MoroMatteo/markerlessBoMI_FaMa`).

- User-friendly Graphical User Interface (GUI) that allows to select the landmark points to be used for cursor controlling.
- Custom calibration step that allows to create a **personal** BoMI forward map to obtain the $(x, y)$ coordinates of the computer cursor.
- Replacement of DLC Live with Mediapipe [Bazarevsky et al., 2020] for the landmark points detection step. In this way it was possible to increase the frame rate of the software. In fact, it is now possible to use it also in laptop without a GPU, reaching a mean frame rate of 27 Hz. We are now testing its performances on laptops and computers with different hardware characteristics.

# 8

# VisionTool

In this chapter we report the implementation of VisionTool, our custom semantic feature detector designed to provide an effective tool to non expert users. VisionTool leverages transfer learning with a large variety of deep neural networks allowing high-accuracy features detection with few training data. The toolbox offers a friendly Graphical User Interface (GUI), efficiently guiding the user through the entire process of features extraction. Moreover, it allows us to have more control on the *features detection* step of our proposed markerless pipeline. To facilitate broad usage and scientific community contribution, the code and an user guide are available at `https://github.com/Malga-Vision/VisionTool.git`.

## 8.1 Introduction

As mentioned in Chapter 3, in order to characterize human motion, it is not always necessary to retrieve the full body skeleton. In fact, there is a large variety of applications where the availability of an accurate algorithm for the detection of semantic features in the image plane may be crucial as evident from the applications described in the previous chapters.

In this context, it becomes clear that versatility is a fundamental feature for a toolbox aiming to provide general-purpose semantic features extraction. In particular it requires: (i) the possibility to define the set of high-level features to detect; (ii) no assumption on input data, which may be a video or a set of static and uncorrelated images; (iii) high accuracy with minimal training data because obtaining annotated data is not a trivial process. In fact, annotation is time-consuming and user-dependent. Moreover, the availability of training data may be intrinsically limited in some fields of application (*e.g.*, medicine and rehabilitation). With respect to the tool adopted in all our studies described in the previous chapters (*i.e.*, DeepLabCut [Mathis et al., 2018]), we need more control on the training step (*i.e.*, parameters and hyperparameters setting) and the possibility to select among an higher number of architectures depending on the complexity of the problem.

For these reasons, we implement VisionTool, a custom Python toolbox for general-purpose markerless semantic features detection. VisionTool is based on transfer learning with deep neural networks, and has been designed to give appropriate importance to the following characteristics.

1. *Versatility*: the toolbox allows the user to define the semantic features to detect and to graphically annotate a set of training data to be used for further steps.

2. *Prediction accuracy*: precision in keypoints coordinates detection is a key factor in pose estimation since high-level features are later extracted from keypoints positions in all of the applications.

3. *Annotation efforts reduction*: after a minimal training set has been annotated (*e.g.*, 5-25 frames, depending on the application), the toolbox offers the possibility to use an assisted annotation procedure. A neural network is trained on the annotated data, and used to predict the remaining frames (either the entire video or a random subset). The predictions are then automatically uploaded to the annotation tool and identified with different color maps with respect to the first set. The user can visually inspect the predictions, and correct mistakes dragging them with the mouse, adding or removing a label, in order to obtain a bigger annotated dataset, potentially improving further predictions.

4. *Simple and immediate adoption*: the toolbox is provided with an intuitive GUI that allows all the users to easily exploit all the implemented features (see Figure 8.1).

5. *Modularity and extensibility*: the toolbox is modular, meaning that new features and modules can be easily added to the package, and the adoption of customized neural networks to perform segmentation is straightforward.

As shown in Section 8.4, VisionTool can be exploited in different ways. Firstly, it can be used as an annotator, meaning that, given the frames composing one video or a set of images, a neural network can be trained on a subset of them and used to predict the remaining ones with high accuracy. In addition, the toolbox has good generalization properties. Thus, it is possible to train a model on a set of frames belonging to one video and use it to detect the analogous set of selected keypoints in frames extracted from a different video. To test VisionTool's versatility and precision, we apply the toolbox to three different domain of applications: (i) action recognition; (ii) face descriptors extraction and (iii) plankton cell tracking. We show that, with less than 50 annotated frames, VisionTool is able to provide accurate features detectors ($mAP^{0.5} > 0.95$) for all the three case studies.

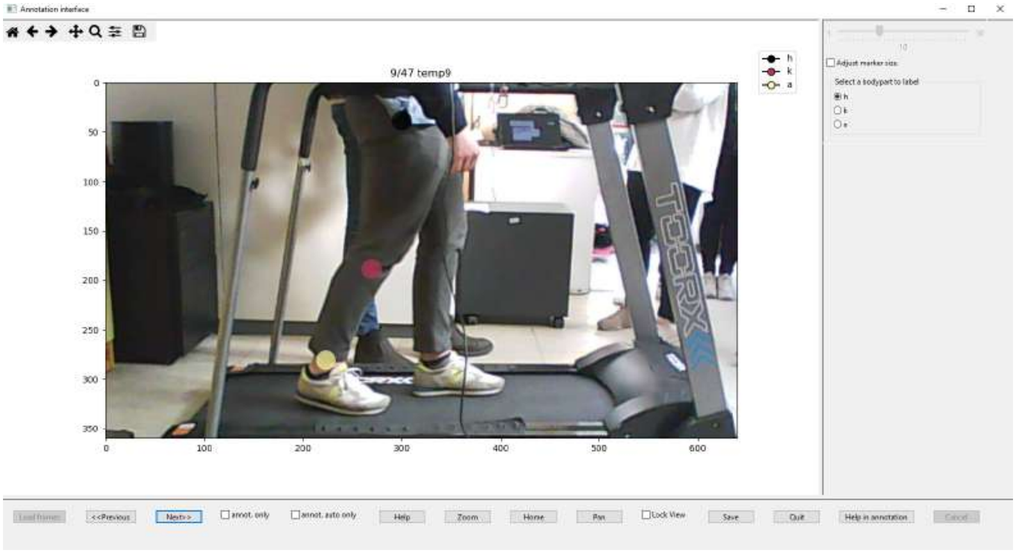


Figure 8.1: Example of VisionTool's annotation GUI. The user can annotate keypoints of interest with the mouse, visualize images and the predictions overlaid on them.

## 8.2 Methods

In this section we provide a schematic description of VisionTool's features extraction pipeline and give a detailed report on the available implemented neural networks.

### 8.2.1 *VisionTool's workflow*

Visiontool offers a user-friendly interface allowing the user to easily exploit all the implemented features. First, the user creates a new project, imports input data (videos or set of images), defines the keypoints (*i.e.*, the semantic features to detect) and selects the number of frames to be annotated, which is randomly extracted from the total available set. The number of frames to be annotated (*i.e.*, the training set size) is a fundamental parameter for the features detection task. A meaningful choice should be a compromise between annotation efforts and the quality of prediction. In general, it depends on the difficulty of the specific task (*e.g.*, number of keypoints, percentage of occlusions, average standard deviation of keypoints location, number of poses in pose estimation applications). To manually perform features annotation, the user exploits the dedicated annotation interface (see Figure 8.1), using the mouse to select the keypoints (*e.g.*, keypoints coordinates in human-pose estimation). A deep neural network is chosen among the available ones and trained on the annotations. Data augmentation based on random transformations (*i.e.*, rotation, shearing, zooming and shifting) is performed at training time to allow for better generalization ensuring high accuracy on few training data. The trained model is then ready to be used to perform features extraction in testing videos (either unseen videos, or the remaining frames of the training video). After testing, the obtained results can be visually inspected by the user; if they are not satisfactory, they can be corrected and used as a further set of annotated data in the training procedure, resulting in a human-in-the-loop framework. See Figure 8.2 for a schematic description of VisionTool's workflow. VisionTool's GUI guides the user through the entire process of semantic features extraction. More details on the main steps are reported in the next subsections.

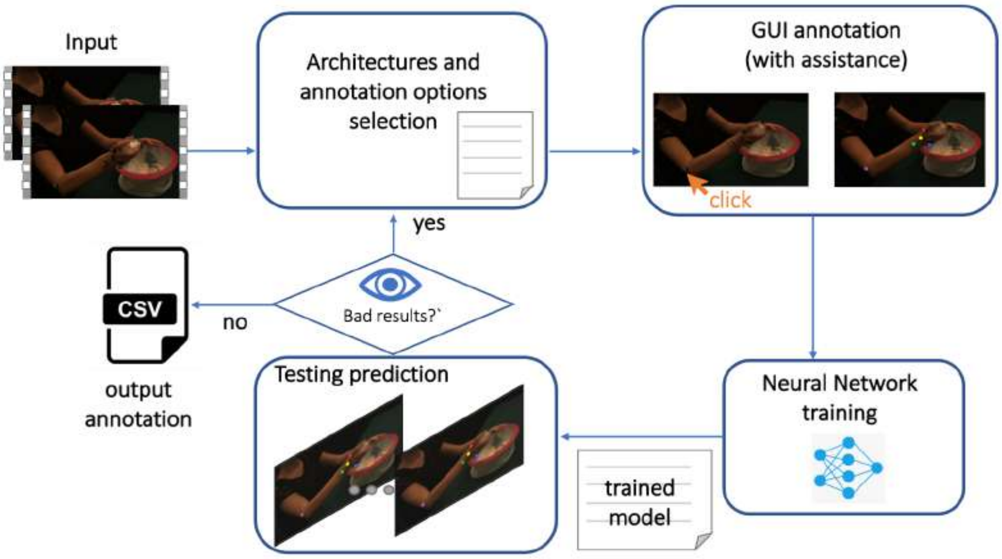


Figure 8.2: VisionTool's workflow description.

## 8.2.2 *Input data import and annotation*

After a project is created (or an existing project is opened), the user can add new videos (or process the existing ones). The videos are automatically read by the toolbox to provide the total length (in number of frames), helping the user to set a valid number of frames to annotate. After the user sets the number $j$ of frames to annotate, a random set of $j$ frames is extracted among all the available ones and annotated. When the user annotates an image, a circle with radius $r$ is drawn over the frame in the annotation tool, where $r$ can be set by the user through the annotations option GUI. Such circles are then used to form the ground truth segmentation masks.

## 8.2.3 *VisionTool as an annotation assistance tool*

The larger the training set, the higher the algorithm precision in detecting the semantic features from videos. However, the annotation procedure is time consuming, forcing a compromise between number of annotations and prediction accuracy. In order to partially solve this issue, VisionTool implements a deep neural network-based automatic annotation procedure. After at least

10 frames are manually annotated, in fact, there is an option to train a deep neural network, to provide an initial annotation estimation for a number *k* of randomly extracted frames, with *k* defined by the user. After the prediction, the automatically annotated frames are loaded in the same GUI used for manual annotation, and the user can check the results and correct potential mispredictions by dragging the points to the correct location, adding or removing a detected keypoint, with a significant saving in term of annotation efforts. The automatic and manual frames predictions are represented with different color maps in order to be clearly distinguishable in the GUI. The checked and corrected frames are added to the original set of manual annotations to increase the training set size. The automatic annotation tool is a key feature and the main novelty in VisionTool, reducing user annotation efforts while speeding up the entire features detection process, eventually leading to a higher prediction accuracy and better generalization.

### 8.2.4 *Available deep neural networks*

VisionTool includes 4 different largely used architectures for detection and segmentation: UNet [Ronneberger, Fischer, and Brox, 2015], LinkNet [Chaurasia and Culurciello, 2017], Pyramid Scene Parsing Network (PSPNet) [Zhao et al., 2017] and Feature Pyramid Network (FPN) [Lin et al., 2017]. These architectures encode the input exploiting sequential downsampling (*i.e.*, compressing the images) and then reconstruct the input by specular sequential upsamples (deconvolution) with different combinations with respect to the downsamples layers according to the specific architectures. The encoding module can be adapted from different neural networks, choosing the number of parameters and network depth according to the specific problem. VisionTool offers 30 models to be used as backbones for each of the available deep network. A key-feature in VisionTool is the possibility to obtain high accuracy in the semantic features extraction with a limited training set (*i.e.*, with limited annotations). Such feature is implemented exploiting transfer learning, providing better generalization than training from scratch. In fact, ImageNet [Deng et al., 2009] pre-trained weights are available for each of the neural network backbones. Neural network implementations are based on the library proposed in [Yakubovskiy, 2019].

### 8.2.5 *Model training and deployment*

A dedicated GUI offers the possibility to select the neural network, the optimizer, the loss function, the learning rate and the number of epochs to wait if validation loss does not decrease before stopping training, training from scratch or using transfer-learning from ImageNet pre-trained weights. The learning rate is halved at every *z* epochs to facilitate the convergence of the trained model, and *z* is again set through the dedicated architectures preferences GUI.

After training, VisionTool can be used to annotate other frames of the same video or frames of new similar videos. The final locations of the detected keypoints are obtained by thresholding the confidence maps. The confidence maps (one per keypoint and of the same size of the input image) have pixels intensities corresponding to the probability of finding the keypoint at that precise location (the higher the intensity, the higher the algorithm confidence about the pixel belonging to that specific keypoint). VisionTool's final output corresponds to a dataframe reporting the estimated locations for the detected features in each frame (pixel location with the highest value in the corresponding confidence map), stored both as a h5 and a csv file. For each video frame $t$, they include the detected coordinates $(x, y)_t^l$ in the image plane for each keypoint $l$ and the corresponding estimation confidence $c_t^l$ (*i.e.*, the value of the confidence map at the location $(x, y)_t^l$ where the keypoint is detected) . If one of the keypoints has not been detected in a certain frame, the corresponding output coordinates are automatically set to a negative number: (-1, -1). The toolbox offers the possibility to save the predicted maps for each keypoint for user visual inspection or further processing.

### 8.2.6 *Evaluation Metrics*

VisionTool's semantic features detection accuracy is evaluated in terms of mean Average Precision (mAP), as explained in Chapter 3. As commonly done in the literature and COCO challenges [Lin et al., 2014a], we compute mAP with respect to three different thresholds, defined as values of Object Keypoint Similarity (OKS): (i) 0.5; (ii) 0.75; (iii) average mAP value with OKS thresholds from 0.50 to 0.95 and steps of 0.05. See Section 3.6 for further details and formal equations. In our evaluation protocol, the standard deviation $ks$ in Equation 3.3 is computed with respect to keypoints mask area, and exploiting redundant annotations, as done in [Lin et al., 2014a]. In our experiments, keypoints circle mask radius is set accordingly to the size of the semantic features to detect: 13 pixels for MOCA dataset (*i.e.*, approximately the the size of the physical markers in the cooking videos); 2 pixels for faces, and 7 pixels for plankton dataset.

The notation mAP$^{0.5}$ corresponds to the mAP computed as in point (i); mAP$^{0.75}$ corresponds to the mAP computed as in point (ii); while mAP refers to mAP computed as in point (iii). In our evaluation metrics, the mAP at OKS=0.5 can be interpreted as the percentage of correct keypoints ($PCK@0.5$) metric [Yang and Ramanan, 2012] (*i.e.*, the fraction of predicted keypoints that fall within a threshold distance from the ground truth location) with a maximum allowed distance corresponding to an Intersection over Union (IoU) between ground truth and prediction keypoints masks equal to 0.5.

## 8.3 Datasets

A semantic features detection toolbox should be versatile with respect to semantic, number of keypoints and domains of application, as well as precise, intuitive, and easy to use. To validate VisionTool with respect to such requirements, we adopt the toolbox in the analysis of three different benchmark datasets and associated application fields: *(i)* upper-body human actions from the Multiview Cooking Actions dataset (MOCA) [Nicora et al., 2020]; *(ii)* human faces from the Facial Keypoints Detection Kaggle's dataset [Bengio, 2016]; *(iii)* videos of swimming plankton cells from the Plankton dataset [Pastore et al., 2020] (see Figure 8.3 for samples of each dataset). Each of them has specific challenges (reported in Section 8.3) that support the evaluation of different aspects of the toolbox.

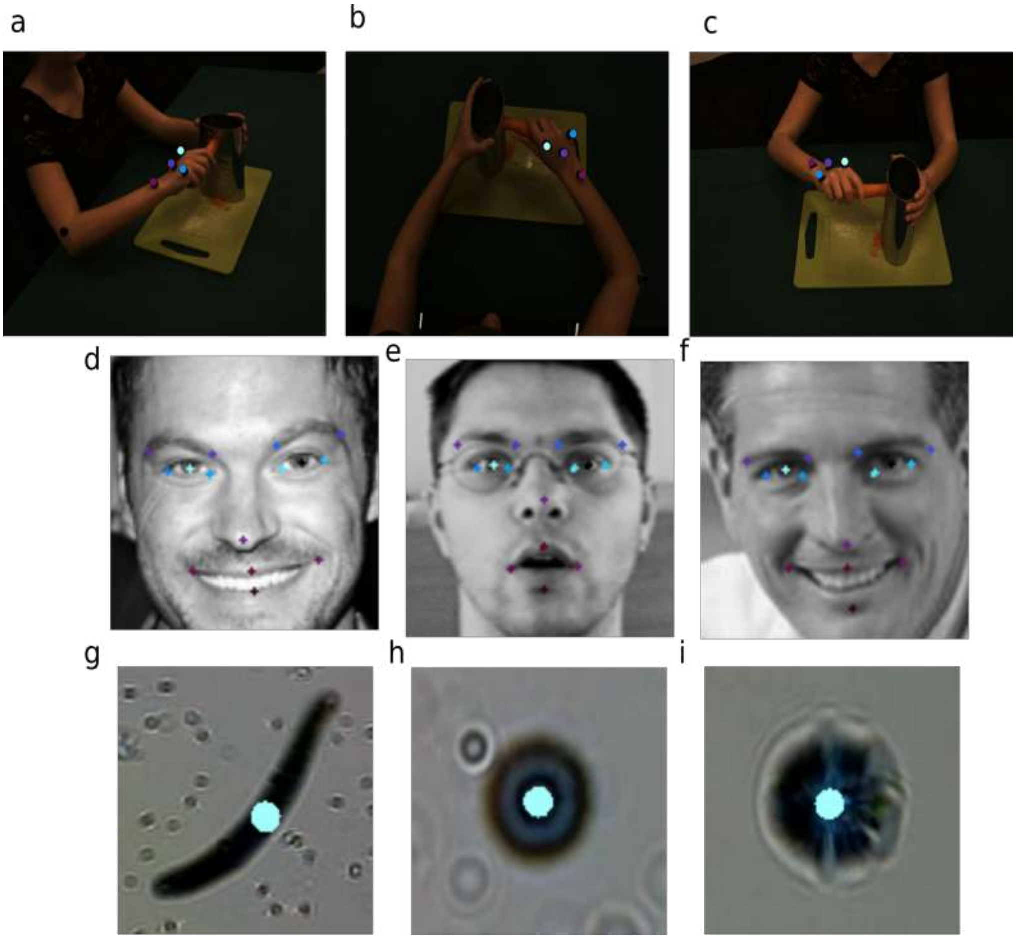


Figure 8.3: Samples for the three datasets included in the work. (a-c) Moca dataset: (a) lateral viewpoint; (b) egocentric viewpoint; (c) frontal viewpoint. (d-f) Kaggle face dataset. (g-i) Plankton dataset: (g) spirostomum ambiguum; (h) arcella vulgaris; (i) didinium nasutum.

multiview cooking actions dataset. The MOCA dataset [Nicora et al., 2020] contains video sequences acquired from multiple views of upper body actions in a cooking scenario. The purpose of MOCA is to provide a rich test bed to understand motion recognition skills and view-invariance properties of both biological and artificial perceptual systems. The dataset includes 20 cooking actions involving one or two arms of a volunteer and the tools to perform the correspondent action. Three different view-points have been considered for the acquisitions, *i.e.*, lateral, egocentric, and frontal. Each action includes a training and a testing video, each containing, on average, 25 repetitions of the action. Since the dataset is multimodal, the volunteer was wearing markers in correspondence to the five keypoints considered for the detection task: (i) index; (ii) little finger; (iii) hand; (iv) wrist and (v) elbow. However, no ground truth annotations are available with the dataset, so we needed to build a 2D ground truth for keypoints location to actually evaluate VisionTool's features detectors accuracy. Hence, ground truth keypoints location is obtained exploiting VisionTool's assistance annotation feature. The presence of physical markers makes the annotation process precise and repeatable, since it is immediate to build the annotation masks on top of the existing markers. On the other hand, occlusions and peculiar motion patterns represent a challenge for detecting the semantic features in the dataset (see Figure 8.4 (a-b)).

FACIAL KEYPOINTS DETECTION DATASET. The Facial Keypoints Detection dataset [Bengio, 2016] was released for a kaggle competition focused on improving features detection accuracy in the context of face recognition. It contains 96 × 96 pixels images of different subjects faces, with a total of 7049 training images and 1783 testing images. Complete annotations are only provided for a subset of the training data. The detection task consists in identifying 15 facial keypoints, divided in 4 semantic groups : (i) eyebrow: left and right inner and outer limits; (ii) eye: left and right eye center, inner and outer corners; (iii) nose: nose tip, (iv) mouth: left and right corners, top and bottom centers. Here, the challenge is mostly related to the low image resolution and the ambiguity in the identification (and annotation) of the keypoints (*e.g.*, the top and bottom center of mouth, can be annotated and correctly predicted within a radius of several pixels, see Figure 8.4 (c-d) for an example).

PLANKTON DATASET. The plankton dataset [Pastore et al., 2020] contains static images of swimming plankton extracted from 1-minute videos of 10 species of plankton acquired using a digital detector. The system used for acquisition employs the principles of a lensless microscope. The dataset includes a total of 5000 images (500 per species) for training, and 1400 images for testing (140 per species). We evaluate VisionTool's accuracy in detecting the center of the plankton cell. No ground truth is available, so we need to annotate the data for actually evaluating VisionTool's detectors accuracy. To perform annotation, first, we exploit an image-processing algorithm to select the centroid of the cell body (*i.e.*, contour detection on available cell body masks, followed by selection of centroid for the contour with highest area). Then, we visually inspect the annotation with VisionTool's annotation GUI, correcting the body cell center detection, when needed. In the plankton dataset, the challenge is represented by the low-resolution images and the intrinsic semantic of the keypoint to detect. For circular shape cells, in fact, the annotation process is trivial and precise. However, a few of the classes included in the dataset (*e.g.*, the spirostomum ambiguum, the dileptus and the stentor coeruleous) can contract and relax (see Figure 8.4 (e-f-g) for an example), radically changing their shape, making hard and not unique the identification of the center cell for annotation and, consequently, for prediction.

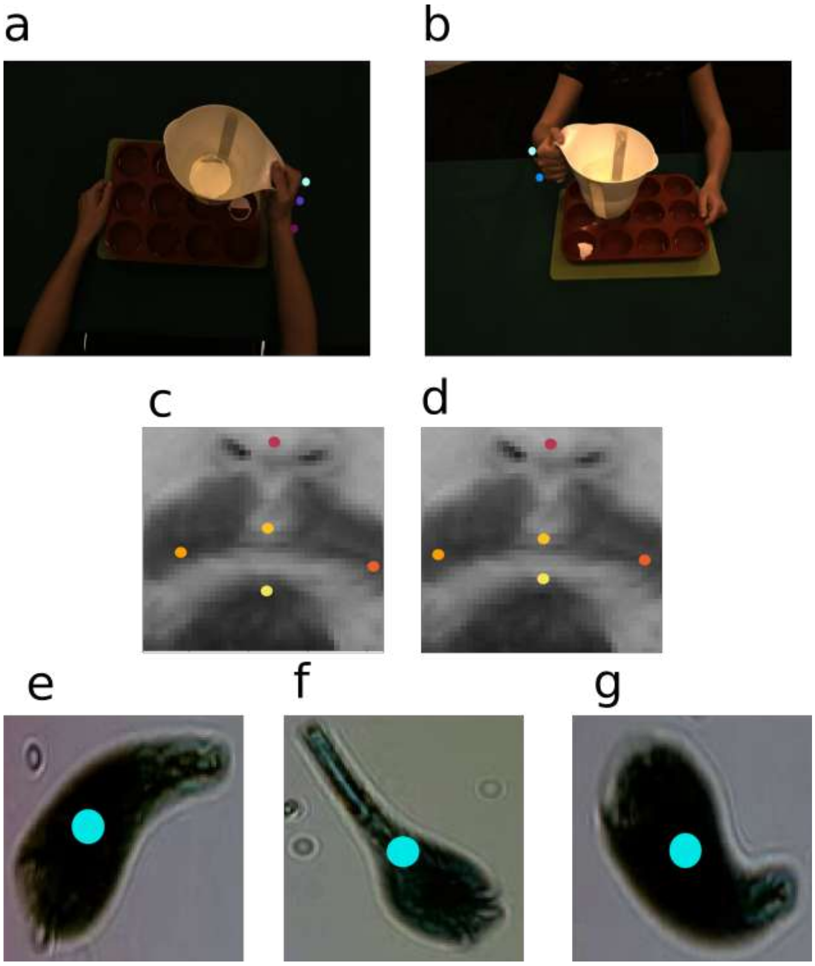


Figure 8.4: Example of challenges for the datasets included in the work. (a-b) Moca's keypoints occlusion in egocentric viewpoint (a) and frontal viewpoint (b). It is common for such views to have the little finger occluded by index, as well as wrist occluded by hand. (c-d) Mouth keypoints can be correctly annotated with a difference of several pixels. (c) ground-truth; (d) example of a different manual annotation. (e-g) stentor ceruleous contracting and relaxing during different stages of swimming with significant shape changing.

## 8.4 Results

### 8.4.1 *VisionTool's results on MOCA dataset*

AUTOMATIC ANNOTATION ACCURACY The toolbox can be adopted as an annotation assistant (*i.e.,* trained on frames belonging to a certain video and tested on its remaining frames), to speed-up the annotation process while reducing user efforts. We used the MOCA dataset testing videos for the three viewpoints (*i.e.,* lateral, egocentric and frontal) to validate VisionTool as automatic annotator. The set of semantic features to detect includes: index, little finger, hand, wrist, and elbow. The first step consisted in using the toolbox to perform manual annotation of the 5 keypoints on a set of randomly extracted training frames. We used a random subset of 10 action videos among the 20 available from the lateral viewpoint to perform an automatic annotation accuracy evaluation as a function of the number of frames manually annotated. For this experiment, we use a LinkNet [Chaurasia and Culurciello, 2017] neural network with EfficientNetb1 [Tan and Le, 2019] backbone pre-trained on ImageNet [Deng et al., 2009] (RMSprop optimizer, weighted categorical cross-entropy as loss function, batch size equal to 5). The number of annotated frames is 10, 25 and 50. As expected, mAP increase with the number of annotated frames, reaching a maximum value of 0.974 for 50 annotated frames (see Table 8.1). A higher number of annotated frames could lead to higher detection accuracy, however, we limit our analysis to 50 frames, since the aim of the experiment is to test the toolbox potential with minimal manual annotation efforts.

| Frames | $mAP^{0.5}$ | $mAP^{0.75}$ | $mAP_{index}$ | $mAP_{little\ finger}$ | $mAP_{hand}$ | $mAP_{wrist}$ | $mAP_{elbow}$ | mAP |
|---|---|---|---|---|---|---|---|---|
| 10 | 0.888 | 0.869 | 0.818 | 0.855 | 0.866 | 0.882 | 0.890 | 0.862 |
| 25 | 0.979 | 0.966 | 0.852 | 0.862 | 0.967 | 0.888 | 0.890 | 0.892 |
| 50 | 0.984 | 0.977 | 0.949 | 0.972 | 0.975 | 0.986 | 0.989 | **0.974** |

Table 8.1: VisionTool's detection accuracy with respect to number of annotated frames on MOCA dataset. A LinkNet with EfficientNetb1 backbone is trained on (i) 10; (ii) 25 and (iii) 50 frames, and used to predict the remaining ones, for each of the 10 lateral viewpoint videos included in the evaluation subset. The results reported in this table correspond to the average mAP computed across the whole subset of videos.

Transfer learning from ImageNet pre-trained models is a key-feature in VisionTool, allowing to obtain high detection accuracy, while providing better generalization than training from scratch, with randomly initialized weights. To show the importance of ImageNet fine-tuning, we train and test VisionTool's semantic features extraction algorithms with randomly initialized weights. With the same number of annotated frames per video and the same neural network and backbone (*i.e.,* LinkNet neural network with EfficientNetb1 backbone), in this case keypoints predictions confidence are below the adopted minimum level of significance (*i.e.,* 0.6 in our experiments), proving the importance of transfer learning to obtain high accuracy semantic features detectors with minimal training data.

As a next step, we evaluate the annotation accuracy with respect to the specific neural network applied. We use the same set of 50 annotated frames of previous step to compute prediction accuracy with the 4 different neural networks available (Unet, LinkNet, Pyramid Scene Parsing Network (PSPNet) and Feature Pyramid Network (FPN)) and two popular neural network backbones in the computer vision literature: EfficientNetb1 and ResNet50 [He et al., 2016]. As reported in Table 8.2, EfficientNetb1 outperformed ResNet50 for all the considered neural networks, with FPN and Unet leading to higher accuracy with respect to the other models. Table 8.3 provides information on the neural networks used in this experiment with respect to number of FLoating point Operations Per Second (FLOPs) and parameters. As we can see, even if Unet and FPN with EfficientNetb1 backbone accuracy are similar, the former works with a number of FLOPs significantly lower than the latter. Thus, having in mind the best compromise between efficiency and accuracy, we use Unet with EfficientNetb1 as backbone, and we train it with 50 annotated frames to evaluate VisionTool's annotation accuracy on the entire MOCA dataset. Table 8.4 summarizes the obtained results.

| Net/Backbone | mAP$^{0.5}$ | mAP$^{0.75}$ | mAP$_{\text{index}}$ | mAP$_{\text{little finger}}$ | mAP$_{\text{hand}}$ | mAP$_{\text{wrist}}$ | mAP$_{\text{elbow}}$ | mAP |
|---|---|---|---|---|---|---|---|---|
| FPN/Efficientb1 | 0.991 | 0.984 | 0.957 | 0.973 | 0.981 | 0.987 | 0.991 | **0.978** |
| FPN/ResNet50 | 0.975 | 0.944 | 0.874 | 0.949 | 0.923 | 0.978 | 0.985 | 0.942 |
| LinkNet/Efficientb1 | 0.992 | **0.987** | 0.974 | 0.945 | 0.985 | 0.971 | 0.976 | 0.969 |
| LinkNet/ResNet50 | 0.858 | 0.849 | 0.769 | 0.786 | 0.859 | 0.788 | 0.890 | 0.819 |
| PSPNet/Efficientb1 | 0.987 | 0.957 | 0.894 | 0.929 | 0.928 | 0.957 | 0.949 | 0.931 |
| PSPNet/ResNet50 | 0.983 | 0.914 | 0.803 | 0.867 | 0.850 | 0.927 | 0.935 | 0.876 |
| Unet/Efficientb1 | **0.993** | 0.981 | 0.962 | 0.976 | 0.978 | 0.984 | 0.976 | 0.970 |
| Unet/ResNet50 | 0.952 | 0.945 | 0.848 | 0.875 | 0.878 | 0.887 | 0.975 | 0.893 |

Table 8.2: VisionTool's detection accuracy on MOCA dataset, with respect to neural networks and backbones. The 4 neural networks (*i.e.*, FPN, LinkNet, PSPNet and Unet) are combined with EfficientNetb1 and ResNet50 backbone. Each model is trained on the 50 annotated frames, and used to predict the remaining ones, for each of the 10 lateral view-point videos included in the evaluation subset. The results reported in this table correspond to the average mAP computed across the whole subset of videos.

GENERALIZATION: PREDICTION OF UNSEEN VIDEOS We showed that VisionTool is able to provide high-accuracy semantic features detectors with minimal annotated data, when used as annotator (*i.e.*, trained on frames belonging to a certain video and tested on its remaining frames). However, when dealing with semantic features extraction tasks, generalization properties are crucial, since the same keypoints will have to be accurately detected in different testing videos with respect to the training ones. This is especially true in pose estimation tasks, where different subjects performs the same action in different environments. To investigate how the algorithms implemented in the toolbox generalize and perform on unseen videos, we use each set of 20

| Net/Backbone | FLOPS(bilions) | Number Params(milions) | Number Layers |
|---|---|---|---|
| FPN/Efficientb1 | 12.80 | 0.96 | 379 |
| FPN/ResNet50 | 6.43 | 2.69 | 237 |
| LinkNet/Efficientb1 | 8.04 | 0.86 | 388 |
| LinkNet/ResNet50 | 2.09 | 2.88 | 246 |
| PSPNet/Efficientb1 | 1.76 | 0.18 | 142 |
| PSPNet/ResNet50 | 0.90 | 0.39 | 116 |
| Unet/Efficientb1 | 8.72 | 1.26 | 373 |
| Unet/ResNet50 | 2.58 | 3.26 | 231 |

Table 8.3: Neural networks and backbones complexity in terms of FLoating point Operations Per Second (FLOPS), number of parameters and layers.

| View point | $mAP^{0.5}$ | $mAP^{0.75}$ | $mAP_{index}$ | $mAP_{little\ finger}$ | $mAP_{hand}$ | $mAP_{wrist}$ | $mAP_{elbow}$ | mAP |
|---|---|---|---|---|---|---|---|---|
| All together | 0.992 | 0.987 | 0.974 | 0.945 | 0.985 | 0.971 | 0.976 | 0.970 |

Table 8.4: VisionTool's detection accuracy on MOCA dataset, when used as annotator. A Unet with EfficientNetb1 backbone is trained on 50 frames, and used to predict the remaining ones, for each of the 60 videos included in the dataset. The results reported in this table correspond to the average mAP computed across the whole set of videos.

videos (one for each viewpoint) to perform a k-fold experiment, with k = 5, each time using one fold for testing and the remaining four to train the detection algorithms (16 training and 4 testing videos). As we can see in Table 8.5 the toolbox is able to provide features detectors that generalize well between different videos. In fact, the $mAP^{0.5}$ is higher than 0.95 for all of the considered sets of videos, while the mean mAP across the 5 different folds, is higher than 0.90. As expected, the elbow and the wrist are the easiest keypoints to detect, since they are the most stable with respect to different videos, while the index and the little-finger are the hardest ones, since they are the ones characterized by the highest level of motion. Finally, as expected, the frontal viewpoint is the hardest one to predict, since videos acquired with such viewpoint present the highest variability of keypoints detection and number of occlusion with respect to the 20 cooking actions. As a final step, we investigated how accurate are VisionTool's detections when trained on three different viewpoints videos at once. Hence, we trained a neural network on the entire dataset with a k-fold approach (k = 5). We split the dataset into the 5 folds imposing to have the same number of videos belonging to the three different viewpoints at each fold (*i.e.*, 16 videos per each view for training and 4 videos for testing, for a total of 48 training and 12 testing videos) obtaining a corresponding mAP equal to 0.908.

| View point | $mAP^{0.5}$ | $mAP^{0.75}$ | $mAP_{\text{index}}$ | $mAP_{\text{little finger}}$ | $mAP_{\text{hand}}$ | $mAP_{\text{wrist}}$ | $mAP_{\text{elbow}}$ | mAP |
|---|---|---|---|---|---|---|---|---|
| Lateral | 0.969 | 0.905 | 0.865 | 0.845 | 0.889 | 0.958 | 0.988 | 0.909 |
| Egocentric | 0.962 | 0.929 | 0.925 | 0.789 | 0.963 | 0.922 | 0.978 | 0.915 |
| Frontal | 0.957 | 0.858 | 0.861 | 0.907 | 0.836 | 0.930 | 0.992 | 0.905 |
| All together | 0.954 | 0.904 | 0.880 | 0.821 | 0.912 | 0.949 | 0.980 | 0.908 |

Table 8.5: VisionTool's detection accuracy on MOCA dataset. A k-fold (k=5) approach is used for each view point (*i.e.*, the detectors are trained on 4 folds and the remaining one was predicted). The results reported in the table correspond to the average mean AP (mAP) computed across the different folds.

### 8.4.2 *Face dataset results*

In this section, we evaluate if VisionTool is able to provide accurate features detection for the face dataset. We extract a set of 1500 images from the training set provided with full annotation. We split the dataset into training and testing with ratio 3:1, resulting in 1000 images for training and 500 for testing. We evaluate the 4 neural networks included in VisionTool (*i.e.*, Unet, LinkNet, PSPNet and FPN) with EfficientNetb1 backbone (considering that on the MOCA dataset this was the best performing backbone, batch size equal to 5, RMSprop optimizer). Table 8.6 summarizes the obtained results in terms of mAP. The detector based on FPN and EfficientNetb1 shows the highest detection accuracy, with a $mAP^{0.75}$ around 0.96 and a mAP of 0.86.

| Net/Backbone | $mAP^{0.5}$ | $mAP^{0.75}$ | $mAP_{\text{eyebrow}}$ | $mAP_{\text{eye}}$ | $mAP_{\text{nose}}$ | $mAP_{\text{mouth}}$ | mAP |
|---|---|---|---|---|---|---|---|
| FPN/Efficientb1 | 0.998 | 0.958 | 0.791 | 0.939 | 0.739 | 0.926 | **0.859** |
| LinkNet/Efficientb1 | 0.998 | 0.950 | 0.771 | 0.920 | 0.724 | 0.908 | 0.838 |
| PSPNet/Efficientb1 | 0.992 | 0.896 | 0.742 | 0.915 | 0.636 | 0.878 | 0.803 |
| Unet/Efficientb1 | 0.994 | 0.934 | 0.749 | 0.905 | 0.708 | 0.896 | 0.824 |

Table 8.6: Facial keypoints detection accuracy in terms of mAP. EfficientNetb1 is used as backbone for the 4 neural networks implemented in VisionTool. The 15 detected Keypoints are divided into 4 semantic groups, as explained in Subsection 8.3

.

### 8.4.3 *Plankton dataset results*

As a final quantitative application, we evaluate if VisionTool is able to provide an accurate detector for the center of the plankton cell body. We consider the testing set of 140 images for each of the 10 included classes of plankton in the dataset, for a total of 1400 images. We consider only the testing set because it contains a sufficient number of images to accomplish our task and because in this way we reduce labeling efforts. For each class, we annotate a random set of 50 images, as previously explained. After ground truth annotations have been created, we train the 4 neural networks included in VisionTool with the same configuration adopted for the Face dataset (previous subsection) on the

50 images for each class, and predicted the plankton cell center on the 90 remaining images. Table 8.7 shows the performances in terms of mAP. Despite the intrinsic morphology change and the arbitrarity in the keypoint annotation, VisionTool is able to achieve high detection accuracy. The most accurate detectors correspond to a FPN and Unet with EfficientNetb1 backbone, reaching a mAP$^{0.75}$ equal to 0.92 and a mAP around 0.91.

| Net/Backbone | mAP$^{0.5}$ | mAP$^{0.75}$ | mAP |
|---|---|---|---|
| FPN/Efficientb1 | 0.980 | 0.919 | **0.908** |
| LinkNet/Efficientb1 | 0.951 | 0.837 | 0.839 |
| PSPNet/Efficientb1 | 0.942 | 0.776 | 0.784 |
| Unet/Efficientb1 | 0.976 | 0.919 | 0.907 |

Table 8.7: Plankton cell center detection accuracy in terms of mAP. EfficientNetb1 is used as backbone for the 4 neural networks implemented in VisionTool.

## 8.5 Discussion

In this chapter, we introduced VisionTool, a toolbox for semantic features extraction. To facilitate broad usage and scientific community contribution, the toolbox is available at `https://github.com/Malga-Vision/VisionTool.git`. We showed that transfer learning from pre-trained deep neural network can be quickly applied to completely different contexts and applications (from cooking actions to swimming cells) with accurate results. We believe that VisionTool could supplement the list of available toolboxes for video analysis, allowing even inexperienced users to obtain high-accuracy features detectors for a wide range of applications.

DATASET ANNOTATION AND PERFORMANCES. VisionTool is based on transfer learning from ImageNet pre-trained deep neural networks, allowing to obtain high-accuracy detectors with minimal annotated training data. In our experiments, we showed that 50 frames were sufficient to obtain high accuracy detectors ($mAP > 0.9$) for the three investigated datasets. In general, the accuracy of fine-tuned features detectors may depend on the number and quality of annotations. A precise labeled training set may be not trivial to obtain, it is time-consuming and user-dependent. As a solution, our toolbox offers the possibility to obtain an additional set of data with an automatic procedure, where a deep neural network is trained to predict a subset of frames, with predictions that are later available in the annotation GUI for checking and potential correction. We used such a procedure to obtain a ground truth for the MOCA dataset, where annotations were not provided with data. However, in noisy videos where objects move with high frequency, frames where this particular behavior is present could be not part of the randomly selected minimal annotated set for training. The exclusion of such frames from training potentially brings to sub-optimal results. In such cases, a solution comes directly from VisionTool's output, with a post-processing training frame addition. In fact, VisionTool provides as output confidence maps (of the same size of the input image) for each keypoint, where pixel intensity corresponds to the confidence of that pixel belonging to the detected keypoint. These maps (also called probability maps) are thresholded with a minimum level of confidence to provide the final predicted keypoints locations. Hence, frames with particularly low levels of confidence could be added to the training set to test if the accuracy can be improved. Low values in the probability maps could also occur when keypoints are occluded. In this case, multiple viewpoints (as in the MOCA dataset) are ideal to improve precision in features extraction.

VISIONTOOL'S VERSATILITY. We showed that VisionTool is able to provide accurate detections for three different datasets: (i) MOCA; (ii) facial keypoints detection and (iii) swimming plankton cells. We chose such datasets because their different features supported the evaluation of specific aspects of the proposed toolbox. In the MOCA dataset, in fact, even if videos were acquired by three different viewpoints, it was still possible to obtain high-accuracy ($mAP > 0.9$) when detectors were trained with different viewpoints videos at once. In the face dataset, we showed that VisionTool provides accurate detections ($mAP > 0.9$) when input data are sequences of static low-resolution images and features are smaller and more user-dependent with respect to the previous dataset (where annotations coincide with physical marker positions). Finally, the plankton dataset has low-resolution images and the position of cell centroid is ambiguous and strongly dependent from the user. To prove this point, we asked three different annotators to provide annotations for 50 frames per each class. The standard deviation among the different set of annotations reached a maximum value of 7 pixels for the class *dileptus*, where strong intrinsic morphology change and the shape of the cell make harder to precisely identify its centroid. However, VisionTool's was still able to train an accurate detector ($mAP > 0.9$) for each of the ten species of plankton included in the dataset.

VISIONTOOL'S COMPUTATIONAL COST DETAILS. The deep neural networks embedded in the toolbox were trained and tested on resized version of the original video frames (in the current version, to size $288 \times 288$), that were later scaled to the original size with no effect on features detection accuracy. Thus, VisionTool's semantic features extraction can be quite fast on modern hardware. For instance, inference rate for the MOCA dataset spanned from 50 to 85 Hz on a Nvidia RTX2060 with 6GB of RAM (for Unet with EfficientNetb1 backbone). Such prediction time makes VisionTool compatible with real-time features detection applications. The inference time could be further decreased by increasing the resize rate, cropping the frames, or modifying the architectures (*e.g.*, with pruning algorithms) to speed up the prediction process.

# 9

# Conclusions

In this thesis we presented an approach to markerless human motion analysis relying only on RGB video acquisition and leveraging computer vision and deep learning algorithms. Our approach presents the following advantages with respect to the gold standard marker-based methods:

1. it requires less expertise and it has no bias introduced by any operators. In fact, while the operator during marker-based data acquisition needs to place markers carefully on the subjects skin in order to avoid biased results, our pipeline works fully automatically, and it is independent of any human performance;

2. it does not affect the naturalness of the motion in any way since it does not require cumbersome markers and sensors. Furthermore, it makes the data acquisition easier and faster because it is not necessary to place markers on the body skin;

3. it is less expensive and with a simpler setup, easier to be used outside laboratory environments, since it requires only RGB cameras.

We tested the reliability and the versatility of our markerless pipeline by comparing it with gold standard techniques and by adopting it in different application tasks.

Firstly, we focused on the study of infants spontaneous movements. In this case, we proceeded with two different techniques: (i) one based on the computation of quantitative parameters that could be adopted to distinguish between normal and abnormal motion patterns in videos acquired during the first weeks after birth and (ii) one based on graphs and NLP methods that could highlight abnormal aspects of the motion patterns usually extracted with a visual analysis. In both cases we obtained encouraging results suggesting the possibility of adopting computer-aided techniques for the quantitative characterization of preterm infants spontaneous movements.

Then, we focused on gait analysis. In this case, we were particularly interested in the comparison of our system with gold standard marker-based techniques. In fact, gait analysis has largely used and well defined protocols that allowed to highlight the drawbacks of our system. We performed both

2D and 3D analysis and we found differences between the results obtained with our markerless system and those obtained with the gold standard, especially during the swing phase, where the high motion of the feet led to motion blur in the acquired videos and, consequently, to small errors in the final results. The limitations highlighted with this analysis should be accounted when adopting our markerless pipeline to detect and monitor abnormal motion patterns in people with orthopaedic injury or neurological diseases and further investigated.

Lastly, to highlight the versatility of the pipeline, we reported the results of its application in two other tasks: (i) the analysis of the motion of violin players and (ii) the implementation of a video-based markerless body machine interface as an assistive tool to allow people with spinal cord injury to control the computer cursor with the motion of the head and shoulders. In these two tasks we demonstrated the wide applicability potential of our procedure.

The core step of our work was the detection of semantic features in the image plane. In fact, with the analysis of the trajectories of the coordinates of the keypoints detected in the image plane with a semantic features detector [Mathis et al., 2018] we were able to achieve all the results described in Part II. This consideration pushed us to think about the importance of having an efficient and easy-to-use semantic features detectors. For these reasons, we implemented VisionTool, our custom tool with an user-friendly graphical user interface for semantic features detection. With VisionTool we reached an higher control during training (*i.e.*, parameters and hyperparameters setting) and we introduced the possibility to select among different backbones CNN architectures depending on the complexity of the problem.

In the future, we are planning to extend the works presented in the previous chapters as follows: (i) adopt the two approaches implemented to characterize infants spontaneous movements to both acquisition sessions (Chapter 5); (ii) analyze the gait of people with Multiple Sclerosis (dataset already acquired) with the implemented markerless pipeline (Chapter 6); (iii) test the new version of our markerless Body Machine Interface (Chapter 7); (iv) focus on the general drawbacks of our pipeline highlighted within the thesis.

In conclusion, the results reported in this thesis suggest that the proposed markerless pipeline is a promising alternative with respect to marker-based systems to study and characterize human motion. We presented many advantages in terms of costs and usability. We highlighted also the main limits and we presented possible solutions to overcome them.

# Bibliography

Acer, Merve and Adnan Furkan Yıldız (2018). 'Force localization estimation using a designed soft tactile sensor.' In: *International Symposium on Wearable Robotics*. Springer, pp. 8–12.

Adde, Lars, Jorunn L Helbostad, Alexander R Jensenius, Gunnar Taraldsen, Kristine H Grunewaldt, and Ragnhild Støen (2010). 'Early prediction of cerebral palsy by computer-based video analysis of general movements: a feasibility study.' In: *Developmental Medicine & Child Neurology* 52.8, pp. 773–778.

Adde, Lars, Jorunn L Helbostad, Alexander Refsum Jensenius, Gunnar Taraldsen, and Ragnhild Støen (2009). 'Using computer-based video analysis in the study of fidgety movements.' In: *Early human development*.

Ahmad, Norhafizan, Raja Ariffin Raja Ghazilla, Nazirah M Khairi, and Vijayabaskar Kasi (2013). 'Reviews on various inertial measurement unit (IMU) sensor applications.' In: *International Journal of Signal Processing Systems* 1.2, pp. 256–262.

Ahmedt-Aristizabal, David, Simon Denman, Kien Nguyen, Sridha Sridharan, Sasha Dionisio, and Clinton Fookes (2019). 'Understanding patients' behavior: Vision-based analysis of seizure disorders.' In: *IEEE journal of biomedical and health informatics* 23.6, pp. 2583–2591.

Alghamdi, Rubayyi and Khalid Alfalqi (2015). 'A survey of topic modeling in text mining.' In: *Int. J. Adv. Comput. Sci. Appl.(IJACSA)* 6.1.

Allen, Marilee C (2008). 'Neurodevelopmental outcomes of preterm infants.' In: *Current opinion in neurology* 21.2, pp. 123–128.

Andrew, Alex M (2001). 'Multiple view geometry in computer vision.' In: *Kybernetes*.

Andriluka, Mykhaylo, Leonid Pishchulin, Peter Gehler, and Bernt Schiele (2014). '2d human pose estimation: New benchmark and state of the art analysis.' In: *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition*, pp. 3686–3693.

Andriluka, Mykhaylo, Stefan Roth, and Bernt Schiele (2009). 'Pictorial structures revisited: People detection and articulated pose estimation.' In: *2009 IEEE conference on computer vision and pattern recognition*. IEEE, pp. 1014–1021.

— (2012). 'Discriminative appearance models for pictorial structures.' In: *International journal of computer vision* 99.3, pp. 259–280.

Augasta, M Gethsiyal and Thangairulappan Kathirvalavakumar (2012). 'Reverse engineering the neural networks for rule extraction in classification problems.' In: *Neural processing letters* 35.2, pp. 131–150.

Baccinelli, Walter, Maria Bulgheroni, Valentina Simonetti, Francesca Fulceri, Angela Caruso, Letizia Gila, and Maria Luisa Scattoni (2020). 'Movidea: A

Software Package for Automatic Video Analysis of Movements in Infants at Risk for Neurodevelopmental Disorders.' In: *Brain Sciences* 10.4, p. 203.

Badler, Norman I and Joseph O'Rourke (1977). 'A human body modelling system for motion studies.' In:

Bartol, Kristijan, David Bojanić, Tomislav Petković, Nicola D'APUZZO, and Tomislav PRIBANIĆ (2020). 'A Review Of 3D Human Pose Estimation From 2D Images.' In: *Int. Conf. and Exhibition on 3D Body Scanning and Processing Technologies*.

Bateson, Melissa and Paul Martin (2021). *Measuring behaviour: an introductory guide*. Cambridge University Press.

Bax, Martin, Murray Goldstein, Peter Rosenbaum, Alan Leviton, Nigel Paneth, Bernard Dan, Bo Jacobsson, and Diane Damiano (2005). 'Proposed definition and classification of cerebral palsy, April 2005.' In: *Developmental medicine and child neurology* 47.8, pp. 571–576.

Bayley, Nancy (2006). *Bayley scales of infant and toddler development: administration manual*. Harcourt assessment.

Bazarevsky, Valentin, Ivan Grishchenko, Karthik Raveendran, Tyler Zhu, Fan Zhang, and Matthias Grundmann (2020). 'BlazePose: On-device Real-time Body Pose tracking.' In: *arXiv preprint arXiv:2006.10204*.

Beckung, Eva and Gudrun Hagberg (2002). 'Neuroimpairments, activity limitations, and participation restrictions in children with cerebral palsy.' In: *Developmental medicine and child neurology* 44.5, pp. 309–316.

Behrens, Janina R, Sebastian Mertens, Theresa Krüger, Anuschka Grobelny, Karen Otte, Sebastian Mansow-Model, Elona Gusho, Friedemann Paul, Alexander U Brandt, and Tanja Schmitz-Hübsch (2016). 'Validity of visual perceptive computing for static posturography in patients with multiple sclerosis.' In: *Multiple Sclerosis Journal* 22.12, pp. 1596–1606.

Bengio, Yoshua (2016). *Facial Keypoints Detection*. URL: `https://www.kaggle.com/c/facial-keypoints-detection/data`.

Betke, Margrit, James Gips, and Peter Fleming (2002). 'The camera mouse: visual tracking of body features to provide computer access for people with severe disabilities.' In: *IEEE Transactions on neural systems and Rehabilitation Engineering* 10.1, pp. 1–10.

Biase, Lazzaro di, Alessandro Di Santo, Maria Letizia Caminiti, Alfredo De Liso, Syed Ahmar Shah, Lorenzo Ricci, and Vincenzo Di Lazzaro (2020). 'Gait analysis in Parkinson's disease: An overview of the most accurate markers for diagnosis and symptoms monitoring.' In: *Sensors* 20.12, p. 3529.

Blei, David M, Andrew Y Ng, and Michael I Jordan (2003). 'Latent dirichlet allocation.' In: *the Journal of machine Learning research* 3, pp. 993–1022.

Borghese, N Alberto, L Bianchi, and F Lacquaniti (1996). 'Kinematic determinants of human locomotion.' In: *The Journal of physiology* 494.3, pp. 863–879.

Bos, Arend F, Aren J van Loon, Mijna Hadders-Algra, Albert Martijn, Albert Okken, and Heinz FR Prechtl (1997). 'Spontaneous motility in preterm, small-forgestational age infants II. Qualitative aspects.' In: *Early human development* 50.1, pp. 131–147.

Breiman, Leo (2001). 'Random forests.' In: *Machine learning* 45.1, pp. 5–32.

Burenius, Magnus, Josephine Sullivan, and Stefan Carlsson (2013). '3D pictorial structures for multiple view articulated pose estimation.' In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3618–3625.

Burger, Marlette and Quinette A Louw (2009). 'The predictive validity of general movements–a systematic review.' In: *European journal of paediatric neurology* 13.5, pp. 408–420.

*COCO 2020 Keypoint Detection Task*. `https://cocodataset.org/#keypoints-eval`. Accessed: 2021-10-01.

Cao, Zhe, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh (2019). 'OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields.' In: *IEEE transactions on pattern analysis and machine intelligence* 43.1, pp. 172–186.

Cao, Zhe, Tomas Simon, Shih-En Wei, and Yaser Sheikh (2017). 'Realtime multi-person 2d pose estimation using part affinity fields.' In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7291–7299.

Cappozzo, Aurelio, Ugo Della Croce, Alberto Leardini, and Lorenzo Chiari (2005). 'Human movement analysis using stereophotogrammetry: Part 1: theoretical background.' In: *Gait & posture* 21.2, pp. 186–196.

Carmo Vilas-Boas, Maria do, Hugo Miguel Pereira Choupina, Ana Patrícia Rocha, José Maria Fernandes, and João Paulo Silva Cunha (2019). 'Full-body motion assessment: Concurrent validation of two body tracking depth sensors versus a gold standard system during gait.' In: *Journal of biomechanics* 87, pp. 189–196.

Carse, Bruce, Barry Meadows, Roy Bowers, and Philip Rowe (2013). 'Affordable clinical gait analysis: An assessment of the marker tracking accuracy of a new low-cost optical 3D motion analysis system.' In: *Physiotherapy* 99.4, pp. 347–351.

Casadio, Maura, Assaf Pressman, Alon Fishbach, Zachary Danziger, Santiago Acosta, David Chen, Hsiang-Yi Tseng, and Ferdinando A Mussa-Ivaldi (2010). 'Functional reorganization of upper-body movement after spinal cord injury.' In: *Experimental brain research* 207.3-4, pp. 233–247.

Casadio, Maura, Rajiv Ranganathan, and Ferdinando A Mussa-Ivaldi (2012). 'The body-machine interface: a new perspective on an old theme.' In: *Journal of Motor behavior* 44.6, pp. 419–433.

Castelli, Andrea, Gabriele Paolini, Andrea Cereatti, and Ugo Della Croce (2015). 'A 2D markerless gait analysis methodology: Validation on healthy subjects.' In: *Computational and mathematical methods in medicine* 2015.

Chambers, Claire, Nidhi Seethapathi, Rachit Saluja, Helen Loeb, Samuel R Pierce, Daniel K Bogen, Laura Prosser, Michelle J Johnson, and Konrad P Kording (2020). 'Computer vision to automatically assess infant neuromotor risk.' In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 28.11, pp. 2431–2442.

Chaurasia, Abhishek and Eugenio Culurciello (2017). 'Linknet: Exploiting encoder representations for efficient semantic segmentation.' In: *2017 IEEE Visual Communications and Image Processing (VCIP)*. IEEE, pp. 1–4.

Chen, Yu-Luen, Fuk-Tan Tang, Walter H Chang, May-Keun Wong, Ying-Ying Shih, and Te-Son Kuo (1999). 'The new design of an infrared-controlled human-computer interface for the disabled.' In: *IEEE Transactions on Rehabilitation Engineering* 7.4, pp. 474–481.

Cheng, Yu, Bo Yang, Bo Wang, Wending Yan, and Robby T Tan (2019). 'Occlusion-aware networks for 3d human pose estimation in video.' In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 723–732.

Chiari, Lorenzo, Ugo Della Croce, Alberto Leardini, and Aurelio Cappozzo (2005a). 'Human movement analysis using stereophotogrammetry: Part 2: Instrumental errors.' In: *Gait & posture* 21.2, pp. 197–211.

Chiari, Lorenzo, Marco Dozza, Angelo Cappello, Fay B Horak, Velio Macellari, and Daniele Giansanti (2005b). 'Audio-biofeedback for balance improvement: an accelerometry-based system.' In: *IEEE transactions on biomedical engineering* 52.12, pp. 2108–2111.

Clark, Ross A, Kelly J Bower, Benjamin F Mentiplay, Kade Paterson, and Yong-Hao Pua (2013). 'Concurrent validity of the Microsoft Kinect for assessment of spatiotemporal gait variables.' In: *Journal of biomechanics* 46.15, pp. 2722–2725.

Colyer, Steffi L, Murray Evans, Darren P Cosker, and Aki IT Salo (2018). 'A review of the evolution of vision-based motion analysis and the integration of advanced computer vision methods towards developing a markerless system.' In: *Sports medicine-open* 4.1, p. 24.

Cook, Albert M and Janice Miller Polgar (2014). *Assistive Technologies-E-Book: Principles and Practice*. Elsevier Health Sciences.

Corazza, Stefano, Lars Muendermann, AM Chaudhari, T Demattio, Claudio Cobelli, and Thomas P Andriacchi (2006). 'A markerless motion capture system to study musculoskeletal biomechanics: visual hull and simulated annealing approach.' In: *Annals of biomedical engineering* 34.6, pp. 1019–1029.

Das, Devleena, Katelyn Fry, and Ayanna M Howard (2018). 'Vision-based detection of simultaneous kicking for identifying movement characteristics of infants at-risk for neuro-disorders.' In: *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, pp. 1413–1418.

Davis III, Roy B, Sylvia Ounpuu, Dennis Tyburski, and James R Gage (1991). 'A gait analysis data collection and reduction technique.' In: *Human movement science* 10.5, pp. 575–587.

Davis, Jesse and Mark Goadrich (2006). 'The relationship between Precision-Recall and ROC curves.' In: *Proceedings of the 23rd international conference on Machine learning*, pp. 233–240.

Davis, Roy B (1988). 'Clinical gait analysis.' In: *IEEE Engineering in Medicine and Biology Magazine* 7.3, pp. 35–40.

De Luca, Alice, Honoré Vernetti, Cristina Capra, Ivano Pisu, Cinzia Cassiano, Laura Barone, Federica Gaito, Federica Danese, Giovanni Antonio Checchia, Carmelo Lentino, et al. (2018). 'Recovery and compensation after robotic assisted gait training in chronic stroke survivors.' In: *Disability and Rehabilitation: Assistive Technology*, pp. 1–13.

*Deep Learning-Based Human Pose Estimation: A Survey*. `https://github.com/zczcwh/DL-HPE`. Accessed: 2021-10-01.

Della Croce, Ugo, Alberto Leardini, Lorenzo Chiari, and Aurelio Cappozzo (2005). 'Human movement analysis using stereophotogrammetry: Part 4: assessment of anatomical landmark misplacement and its effects on joint kinematics.' In: *Gait & posture* 21.2, pp. 226–237.

Delp, Scott L, Frank C Anderson, Allison S Arnold, Peter Loan, Ayman Habib, Chand T John, Eran Guendelman, and Darryl G Thelen (2007). 'OpenSim: open-source software to create and analyze dynamic simulations of movement.' In: *IEEE transactions on biomedical engineering* 54.11, pp. 1940–1950.

Deng, Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei (2009). 'Imagenet: A large-scale hierarchical image database.' In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee, pp. 248–255.

Di Mattia, Philip A, Francis Xavier Curran, and James Gips (2001). *An eye control teaching device for students without language expressive capacity: EagleEyes*. Vol. 53. Edwin Mellen Press.

Droeschel, David and Sven Behnke (2011). '3D body pose estimation using an adaptive person model for articulated ICP.' In: *International Conference on Intelligent Robotics and Applications*. Springer, pp. 157–167.

Eichner, Marcin, Manuel Marin-Jimenez, Andrew Zisserman, and Vittorio Ferrari (2012). '2d articulated human pose estimation and retrieval in (almost) unconstrained still images.' In: *International journal of computer vision* 99.2, pp. 190–214.

Elhayek, Ahmed, Edilson de Aguiar, Arjun Jain, Jonathan Thompson, Leonid Pishchulin, Mykhaylo Andriluka, Christoph Bregler, Bernt Schiele, and Christian Theobalt (2016). 'MARCOnI—ConvNet-Based MARker-less motion capture in outdoor and indoor scenes.' In: *IEEE transactions on pattern analysis and machine intelligence* 39.3, pp. 501–514.

Fan, Mingming, Dana Gravem, Dan M Cooper, and Donald J Patterson (2012). 'Augmenting gesture recognition with erlang-cox models to identify neurological disorders in premature babies.' In: *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pp. 411–420.

Felzenszwalb, Pedro F, Ross B Girshick, David McAllester, and Deva Ramanan (2009). 'Object detection with discriminatively trained part-based models.' In: *IEEE transactions on pattern analysis and machine intelligence* 32.9, pp. 1627–1645.

Felzenszwalb, Pedro F and Daniel P Huttenlocher (2005). 'Pictorial structures for object recognition.' In: *International journal of computer vision* 61.1, pp. 55–79.

Felzenszwalb, Pedro, David McAllester, and Deva Ramanan (2008). 'A discriminatively trained, multiscale, deformable part model.' In: *2008 IEEE conference on computer vision and pattern recognition*. Ieee, pp. 1–8.

Ferrari, Vittorio, Manuel Marin-Jimenez, and Andrew Zisserman (2008). 'Progressive search space reduction for human pose estimation.' In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 1–8.

Fischler, Martin A and Robert A Elschlager (1973). 'The representation and matching of pictorial structures.' In: *IEEE Transactions on computers* 100.1, pp. 67–92.

Friston, KJ, J Ashburner, S Kiebel, TE Nichols, and WD Penny (2007). 'Statistical Parametric Mapping: The Analysis of Functional Brain Images Academic Press.' In: *Statistical Parametric Mapping: The Analysis of Functional Brain Images Academic Press*.

Fritz, Nora E, Rhul Evans R Marasigan, Peter A Calabresi, Scott D Newsome, and Kathleen M Zackowski (2015). 'The impact of dynamic balance measures on walking performance in multiple sclerosis.' In: *Neurorehabilitation and neural repair* 29.1, pp. 62–69.

Fu, Yun and Thomas S Huang (2007). 'hMouse: Head tracking driven virtual computer mouse.' In: *2007 IEEE Workshop on Applications of Computer Vision (WACV'07)*. IEEE, pp. 30–30.

Gabel, Moshe, Ran Gilad-Bachrach, Erin Renshaw, and Assaf Schuster (2012). 'Full body gait analysis with Kinect.' In: *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, pp. 1964–1967.

Garbarino, Davide, Matteo Moro, Chiara Tacchino, Paolo Moretti, Maura Casadio, Francesca Odone, and Annalisa Barla (2021). 'Attributed Graphettes-Based Preterm Infants Motion Analysis.' In: *International Conference on Complex Networks and Their Applications*. Springer, pp. 82–93.

Garello, Luca, Matteo Moro, Chiara Tacchino, Francesca Campone, Paola Durand, Isabella Blanchi, Paolo Moretti, Maura Casadio, and Francesca Odone (2021). 'A Study of At-term and Preterm Infants' Motion Based on Markerless Video Analysis.' In: *2021 29th European Signal Processing Conference (EUSIPCO)*. IEEE, pp. 1196–1200.

Gers, Felix A, Jürgen Schmidhuber, and Fred Cummins (2000). 'Learning to forget: Continual prediction with LSTM.' In: *Neural computation* 12.10, pp. 2451–2471.

Girshick, Ross (2015). 'Fast r-cnn.' In: *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448.

Goldberg, Yoav (2017). 'Neural network methods for natural language processing.' In: *Synthesis lectures on human language technologies* 10.1, pp. 1–309.

Grossi, Giuliano, Raffaella Lanzarotti, Paolo Napoletano, Nicoletta Noceti, and Francesca Odone (2020). 'Positive technology for elderly well-being: A review.' In: *Pattern Recognition Letters* 137, pp. 61–70.

Hartley, R. and A. Zisserman (2004). *Multiple View Geometry in Computer Vision*. Second. Cambridge University Press, ISBN: 0521540518.

Hartley, Richard, Jochen Trumpf, Yuchao Dai, and Hongdong Li (2013). 'Rotation averaging.' In: *International journal of computer vision* 103.3, pp. 267–305.

Hasan, Adib, Po-Chien Chung, and Wayne Hayes (2017). 'Graphettes: Constant-time determination of graphlet and orbit identity including (possibly disconnected) graphlets up to size 8.' In: *PloS one* 12.8, e0181570.

Hasegawa, Naoya, Kas C Maas, Vrutangkumar V Shah, Patricia Carlson-Kuhta, John G Nutt, Fay B Horak, Tadayoshi Asaka, and Martina Mancini (2021). 'Functional limits of stability and standing balance in people with Parkinson's disease with and without freezing of gait using wearable sensors.' In: *Gait & posture* 87, pp. 123–129.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016). 'Deep residual learning for image recognition.' In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

He, Yihui, Rui Yan, Katerina Fragkiadaki, and Shoou-I Yu (2020). 'Epipolar transformers.' In: *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pp. 7779–7788.

Hesse, Nikolas, Christoph Bodensteiner, Michael Arens, Ulrich G Hofmann, Raphael Weinberger, and A Sebastian Schroeder (2018). 'Computer vision for medical infant motion analysis: State of the art and rgb-d data set.' In: *Proceedings of the ECCV*, pp. 0–0.

Higgins, Irina, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner (2016). 'beta-vae: Learning basic visual concepts with a constrained variational framework.' In:

Hinton, G (1976). 'Using relaxation to find a puppet.' In: *Proceedings of the 2nd Summer Conference on Artificial Intelligence and Simulation of Behaviour*, pp. 148–157.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). 'Long short-term memory.' In: *Neural computation* 9.8, pp. 1735–1780.

Hogg, David (1983). 'Model-based vision: a program to see a walking person.' In: *Image and Vision computing* 1.1, pp. 5–20.

Inertial, Performance While Using (2018). 'Wearable Sensory Apparatus Performance While Using Inertial Measurement Units.' In: *Wearable Robotics: Challenges and Trends: Proceedings of the 4th International Symposium on Wearable Robotics, WeRob2018, October 16-20, 2018, Pisa, Italy*. Vol. 22. Springer, p. 23.

Insafutdinov, Eldar, Leonid Pishchulin, Bjoern Andres, Mykhaylo Andriluka, and Bernt Schiele (2016). 'Deepercut: A deeper, stronger, and faster multi-person pose estimation model.' In: *European Conference on Computer Vision*. Springer, pp. 34–50.

Ioffe, Sergey and David A. Forsyth (2001). 'Probabilistic methods for finding people.' In: *International Journal of Computer Vision* 43.1, pp. 45–68.

Ionescu, Catalin, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu (2013). 'Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments.' In: *IEEE transactions on pattern analysis and machine intelligence* 36.7, pp. 1325–1339.

Iskakov, Karim, Egor Burkov, Victor Lempitsky, and Yury Malkov (2019). 'Learnable triangulation of human pose.' In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7718–7727.

Islam, Muhammad RU, Kun Xu, and Shaoping Bai (2018). 'Position sensing and control with FMG sensors for exoskeleton physical assistance.' In: *International Symposium on Wearable Robotics*. Springer, pp. 3–7.

Javanovic, Rados and Ian Scott MacKenzie (2010). 'MarkerMouse: mouse cursor control using a head-mounted marker.' In: *International Conference on Computers for Handicapped Persons*. Springer, pp. 49–56.

Jeong, Hyuk, Jong-Sung Kim, and Wook-Ho Son (2005). 'An emg-based mouse controller for a tetraplegic.' In: *2005 IEEE International Conference on Systems, Man and Cybernetics*. Vol. 2. IEEE, pp. 1229–1234.

Kane, Gary, Gonçalo Lopes, Jonny Sanders, Alexander Mathis, and Mackenzie Mathis (2020). 'Real-time, low-latency closed-loop feedback using markerless posture tracking.' In: *BioRxiv*.

Kang, Taeseok, Minsu Chae, Eunbin Seo, Mingyu Kim, and Jinmo Kim (2020). 'DeepHandsVR: Hand interface using deep learning in immersive virtual reality.' In: *Electronics* 9.11, p. 1863.

Khouri, N and E Desailly (2017). 'Contribution of clinical gait analysis to single-event multi-level surgery in children with cerebral palsy.' In: *Orthopaedics & Traumatology: Surgery & Research* 103.1, S105–S111.

Kidziński, Łukasz, Bryan Yang, Jennifer L Hicks, Apoorva Rajagopal, Scott L Delp, and Michael H Schwartz (2020). 'Deep neural networks enable quantitative movement analysis using single-camera videos.' In: *Nature communications* 11.1, pp. 1–10.

Kim, Hyungsook, David O'Sullivan, Ksenia Kolykhalova, Antonio Camurri, and Yonghyun Park (2021). 'Evaluation of a Computer Vision-Based System to Analyse Behavioral Changes in High School Classrooms.' In: *International Journal of Information and Communication Technology Education (IJICTE)* 17.4, pp. 1–12.

Kim, Soochan, Minje Park, Sasiporn Anumas, and Jaeha Yoo (2010). 'Head mouse system based on gyro-and opto-sensors.' In: *2010 3rd International Conference on Biomedical Engineering and Informatics*. Vol. 4. IEEE, pp. 1503–1506.

Kingma, Diederik P and Max Welling (2013). 'Auto-encoding variational bayes.' In: *arXiv preprint arXiv:1312.6114*.

Kocabas, Muhammed, Salih Karagoz, and Emre Akbas (2019). 'Self-supervised learning of 3d human pose using multi-view geometry.' In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1077–1086.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton (2012). 'Imagenet classification with deep convolutional neural networks.' In: *Advances in neural information processing systems* 25, pp. 1097–1105.

Kwolek, Bogdan, Agnieszka Michalczuk, Tomasz Krzeszowski, Adam Switonski, Henryk Josinski, and Konrad Wojciechowski (2019). 'Calibrated and synchronized multi-view video and motion capture dataset for evaluation of gait recognition.' In: *Multimedia Tools and Applications* 78.22, pp. 32437–32465.

Langhorne, Peter, Fiona Coupar, and Alex Pollock (2009). 'Motor recovery after stroke: a systematic review.' In: *The Lancet Neurology* 8.8, pp. 741–754.

Leardini, Alberto, Lorenzo Chiari, Ugo Della Croce, and Aurelio Cappozzo (2005). 'Human movement analysis using stereophotogrammetry: Part 3. Soft tissue artifact assessment and compensation.' In: *Gait & posture* 21.2, pp. 212–225.

Lee, Hsi-Jian and Zen Chen (1985). 'Determination of 3D human body postures from a single view.' In: *Computer Vision, Graphics, and Image Processing* 30.2, pp. 148–168.

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

Li, Wenhao, Hong Liu, Runwei Ding, Mengyuan Liu, Pichao Wang, and Wenming Yang (2021). 'Exploiting Temporal Contexts with Strided Transformer for 3D Human Pose Estimation.' In: *arXiv preprint arXiv:2103.14304*.

Lin, Tsung-Yi, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie (2017). 'Feature pyramid networks for object detection.' In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125.

Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick (2014a). 'Microsoft COCO: Common Objects in Context.' In: ed. by David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, pp. 740–755.

Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick (2014b). 'Microsoft coco: Common objects in context.' In: *European conference on computer vision*. Springer, pp. 740–755.

Liu, Jun, Amir Shahroudy, Dong Xu, Alex C Kot, and Gang Wang (2017). 'Skeleton-based action recognition using spatio-temporal lstm network with trust gates.' In: *IEEE transactions on pattern analysis and machine intelligence* 40.12, pp. 3007–3021.

Long, Qingqing, Yilun Jin, Guojie Song, Yi Li, and Wei Lin (2020). 'Graph structural-topic neural network.' In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1065–1073.

Lopez-Nava, Irvin Hussein and Angélica Muñoz-Meléndez (2016). 'Wearable inertial sensors for human motion analysis: A review.' In: *IEEE Sensors Journal* 16.22, pp. 7821–7834.

Marr, David and Herbert Keith Nishihara (1978). 'Representation and recognition of the spatial organization of three-dimensional shapes.' In: *Proceedings of the Royal Society of London. Series B. Biological Sciences* 200.1140, pp. 269–294.

Martinez, Julieta, Rayat Hossain, Javier Romero, and James J Little (2017). 'A simple yet effective baseline for 3d human pose estimation.' In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2640–2649.

Mathis, Alexander, Pranav Mamidanna, Kevin M Cury, Taiga Abe, Venkatesh N Murthy, Mackenzie Weygandt Mathis, and Matthias Bethge (2018). 'DeepLabCut: markerless pose estimation of user-defined body parts with deep learning.' In: *Nature neuroscience* 21.9, p. 1281.

*MediaPipe*. `https://google.github.io/mediapipe/`. Accessed: 2021-12-01.

Mehta, Dushyant, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt (2017). 'Vnect: Real-time 3d human pose estimation with a single rgb camera.' In: *ACM Transactions on Graphics (TOG)* 36.4, p. 44.

Meinecke, L, N Breitbach-Faller, C Bartz, R Damen, G Rau, and C Disselhorst-Klug (2006). 'Movement analysis in the early detection of newborns at risk for developing spasticity due to infantile cerebral palsy.' In: *Human movement science* 25.2, pp. 125–144.

Mentiplay, Benjamin F, Luke G Perraton, Kelly J Bower, Yong-Hao Pua, Rebekah McGaw, Sophie Heywood, and Ross A Clark (2015). 'Gait assessment using the Microsoft Xbox One Kinect: Concurrent validity and inter-day reliability of spatiotemporal and kinematic variables.' In: *Journal of biomechanics* 48.10, pp. 2166–2170.

Miehlbradt, Jenifer, Alexandre Cherpillod, Stefano Mintchev, Martina Coscia, Fiorenzo Artoni, Dario Floreano, and Silvestro Micera (2018). 'Data-driven body–machine interface for the accurate control of drones.' In: *Proceedings of the National Academy of Sciences* 115.31, pp. 7913–7918.

Milo, Ron, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon (2002). 'Network motifs: simple building blocks of complex networks.' In: *Science* 298.5594, pp. 824–827.

Mimno, David, Hanna Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum (2011). 'Optimizing semantic coherence in topic models.' In: *Proceedings of the 2011 conference on empirical methods in natural language processing*, pp. 262–272.

Moreno-Noguer, Francesc (2017). '3d human pose estimation from a single image via distance matrix regression.' In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2823–2832.

Moro, Matteo, Maura Casadio, Leigh Ann Mrotek, Rajiv Ranganathan, Robert Scheidt, and Francesca Odone (2021a). 'On The Precision Of Markerless 3d Semantic Features: An Experimental Study On Violin Playing.' In: *2021 IEEE International Conference on Image Processing (ICIP)*. IEEE, pp. 2733–2737.

Moro, Matteo, Giorgia Marchesi, Filip Hesse, Francesca Odone, and Maura Casadio (2022). 'Markerless vs. Marker-Based Gait Analysis: A Proof of Concept Study.' In: *Sensors* 22.5, p. 2011.

Moro, Matteo, Giorgia Marchesi, Francesca Odone, and Maura Casadio (2020). 'Markerless gait analysis in stroke survivors based on computer vision and deep learning: a pilot study.' In: *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, pp. 2097–2104.

Moro, Matteo, Fabio Rizzoglio, Francesca Odone, and Maura Casadio (2021b). 'A Video-Based MarkerLess Body Machine Interface: A Pilot Study.' In: *International Conference on Pattern Recognition*. Springer, pp. 233–240.

*Motive: optical motion capture software*. `https://optitrack.com/software/motive/`. Accessed: 2021-11-01.

Narayanan, Venkatraman, Bala Murali Manoghar, Vishnu Sashank Dorbala, Dinesh Manocha, and Aniket Bera (2020). 'Proxemo: Gait-based emotion learning and multi-view proxemic fusion for socially-aware robot navigation.' In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 8200–8207.

Needham, Laurie, Murray Evans, Darren P Cosker, Logan Wade, Polly M McGuigan, James L Bilzon, and Steffi L Colyer (2021). 'The accuracy of several pose estimation methods for 3D joint centre localisation.' In: *Scientific reports* 11.1, pp. 1–11.

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

Nicora, Elena, Gaurvi Goyal, Nicoletta Noceti, Alessia Vignolo, Alessandra Sciutti, and Francesca Odone (Dec. 2020). 'The MoCA dataset, kinematic and multi-view visual streams of fine-grained cooking actions.' In: *Scientific Data* 7. DOI: `10.1038/s41597-020-00776-9`.

*Optitrack*. `https://optitrack.com/`. Accessed: 2021-11-01.

*Optotrack*. `https://www.ndigital.com/`. Accessed: 2021-11-01.

O'rourke, Joseph and Norman I Badler (1980). 'Model-based image analysis of human motion using constraint propagation.' In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 6, pp. 522–536.

O'Connor, Ciara M, Susannah K Thorpe, Mark J O'Malley, and Christopher L Vaughan (2007). 'Automatic detection of gait events using kinematic data.' In: *Gait & posture* 25.3, pp. 469–474.

Palmer, Frederick B (2004). 'Strategies for the early diagnosis of cerebral palsy.' In: *The Journal of pediatrics* 145.2, S8–S11.

Pastore, Vito P., Thomas G. Zimmerman, Sujoy K. Biswas, and Simone Bianco (July 22, 2020). 'Annotation-free learning of plankton for classification and anomaly detection.' In: *Scientific Reports* 10.1, p. 12142. ISSN: 2045-2322. DOI: `10.1038/s41598-020-68662-3`. URL: `https://doi.org/10.1038/s41598-020-68662-3`.

Pataky, Todd C, Mark A Robinson, and Jos Vanrenterghem (2013). 'Vector field statistical analysis of kinematic and force trajectories.' In: *Journal of biomechanics* 46.14, pp. 2394–2401.

Pataky, Todd C, Jos Vanrenterghem, and Mark A Robinson (2015). 'Zero-vs. one-dimensional, parametric vs. non-parametric, and confidence interval vs. hypothesis testing procedures in one-dimensional biomechanical trajectory analysis.' In: *Journal of biomechanics* 48.7, pp. 1277–1285.

Pavlakos, Georgios, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis (2017). 'Coarse-to-fine volumetric prediction for single-image 3D human pose.' In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7025–7034.

Pavllo, Dario, Christoph Feichtenhofer, David Grangier, and Michael Auli (2019). '3d human pose estimation in video with temporal convolutions and semi-supervised training.' In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7753–7762.

Perez, Luis and Jason Wang (2017). 'The effectiveness of data augmentation in image classification using deep learning.' In: *arXiv preprint arXiv:1712.04621*.

Pierella, Camilla, Farnaz Abdollahi, Elias Thorp, Ali Farshchiansadegh, Jessica Pedersen, Ismael Seáñez-González, Ferdinando A Mussa-Ivaldi, and Maura Casadio (2017). 'Learning new movements after paralysis: Results from a home-based study.' In: *Scientific reports* 7.1, pp. 1–11.

Pishchulin, Leonid, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter V Gehler, and Bernt Schiele (2016). 'Deepcut: Joint subset partition and labeling for multi person pose estimation.' In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4929–4937.

Popa, Alin-Ionut, Mihai Zanfir, and Cristian Sminchisescu (2017). 'Deep multitask architecture for integrated 2d and 3d human sensing.' In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6289–6298.

*Pose Detection with MoveNet and TensorFlow js*. `https://blog.tensorflow.org/2021/05/next-generation-pose-detection-with-movenet-and-tensorflowjs.html`. Accessed: 2021-12-01.

Prechtl, Heinz F (1990). 'Qualitative changes of spontaneous movements in fetus and preterm infant are a marker of neurological dysfunction.' In: *Early human development*.

Prechtl, Heinz (1997). *State of the art of a new functional assessment of the young nervous system. An early predictor of cerebral palsy*.

Pržulj, Natasa, Derek G Corneil, and Igor Jurisica (2004). 'Modeling interactome: scale-free or geometric?' In: *Bioinformatics* 20.18, pp. 3508–3515.

Pueo, Basilio (2016). 'High speed cameras for motion analysis in sports science.' In: *Journal of Human Sport and Exercise* 11.1, pp. 53–73.

Qiu, Haibo, Chunyu Wang, Jingdong Wang, Naiyan Wang, and Wenjun Zeng (2019). 'Cross view fusion for 3d human pose estimation.' In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4342–4351.

Rahmati, Hodjat, Ole Morten Aamo, Øyvind Stavdahl, Ralf Dragon, and Lars Adde (2014). 'Video-based early cerebral palsy prediction using motion segmentation.' In: *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, pp. 3779–3783.

Rahmati, Hodjat, Ralf Dragon, Ole Morten Aamo, Lars Adde, Øyvind Stavdahl, and Luc Van Gool (2015). 'Weakly supervised motion segmentation with particle matching.' In: *Computer Vision and Image Understanding* 140, pp. 30–42.

Rajagopal, Apoorva, Christopher L Dembia, Matthew S DeMers, Denny D Delp, Jennifer L Hicks, and Scott L Delp (2016). 'Full-body musculoskeletal model for muscle-driven simulation of human gait.' In: *IEEE transactions on biomedical engineering* 63.10, pp. 2068–2079.

Ramanan, Deva (2006). 'Learning to parse images of articulated bodies.' In: *Nips*. Vol. 1. 6. Citeseer, p. 7.

Ramanan, Deva, David A Forsyth, and Andrew Zisserman (2005). 'Strike a pose: Tracking people by finding stylized poses.' In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Vol. 1. IEEE, pp. 271–278.

Reddy, N Dinesh, Laurent Guigues, Leonid Pishchulin, Jayan Eledath, and Srinivasa G Narasimhan (2021). 'TesseTrack: End-to-End Learnable Multi-Person Articulated 3D Pose Tracking.' In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15190–15200.

Reich, Simon, Dajie Zhang, Tomas Kulvicius, Sven Bölte, Karin Nielsen-Saines, Florian B Pokorny, Robert Peharz, Luise Poustka, Florentin Wörgötter, Christa Einspieler, et al. (2021). 'Novel AI driven approach to classify infant motor functions.' In: *Scientific Reports* 11.1, pp. 1–13.

Rezatofighi, Hamid, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese (2019). 'Generalized intersection over union: A metric and a loss for bounding box regression.' In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 658–666.

Rizzoglio, Fabio, Camilla Pierella, Dalia De Santis, Ferdinando A Mussa-Ivaldi, and Maura Casadio (2020). 'A hybrid body-machine interface integrating signals from muscles and motions.' In: *Journal of Neural Engineering*.

Röder, Michael, Andreas Both, and Alexander Hinneburg (2015). 'Exploring the space of topic coherence measures.' In: *Proceedings of the eighth ACM international conference on Web search and data mining*, pp. 399–408.

Rodrigues, Thiago Braga, Debora Pereira Salgado, Ciarán Ó Catháin, Noel O'Connor, and Niall Murray (2020). 'Human gait assessment using a 3D marker-less multimodal motion capture system.' In: *Multimedia Tools and Applications* 79.3, pp. 2629–2651.

Rogez, Grégory and Cordelia Schmid (2016). 'Mocap-guided data augmentation for 3d pose estimation in the wild.' In: *arXiv preprint arXiv:1607.02046*.

Rogez, Gregory, Philippe Weinzaepfel, and Cordelia Schmid (2017). 'Lcr-net: Localization-classification-regression for human pose.' In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3433–3441.

Ronfard, Remi, Cordelia Schmid, and Bill Triggs (2002). 'Learning to parse pictures of people.' In: *European Conference on Computer Vision*. Springer, pp. 700–714.

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (2015). 'U-net: Convolutional networks for biomedical image segmentation.' In: *International Conference on Medical image computing and computer-assisted intervention*. Springer, pp. 234–241.

Ruggero Ronchi, Matteo and Pietro Perona (2017). 'Benchmarking and error diagnosis in multi-instance pose estimation.' In: *Proceedings of the IEEE international conference on computer vision*, pp. 369–378.

Saboune, Jamal and François Charpillet (2007). 'Markerless human motion tracking from a single camera using interval particle filtering.' In: *International Journal on Artificial Intelligence Tools* 16.04, pp. 593–609.

Salton, Gerard and Donna Harman (2003). 'Information retrieval.' In: *Encyclopedia of computer science*, pp. 858–863.

Scarselli, Franco, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini (2008). 'The graph neural network model.' In: *IEEE transactions on neural networks* 20.1, pp. 61–80.

Shan, Wenkang, Haopeng Lu, Shanshe Wang, Xinfeng Zhang, and Wen Gao (2021). 'Improving Robustness and Accuracy via Relative Information Encoding in 3D Human Pose Estimation.' In: *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 3446–3454.

Sigal, Leonid, Alexandru O Balan, and Michael J Black (2010). 'Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion.' In: *International journal of computer vision* 87.1-2, p. 4.

Simonyan, Karen and Andrew Zisserman (2014). 'Very deep convolutional networks for large-scale image recognition.' In: *arXiv preprint arXiv:1409.1556*.

Sival, DA, GHA Visser, and HFR Prechtl (1992). 'The effect of intrauterine growth retardation on the quality of general movements in the human fetus.' In: *Early human development* 28.2, pp. 119–132.

Song, Yale, David Demirdjian, and Randall Davis (2012). 'Continuous body and hand gesture recognition for natural human-computer interaction.' In: *ACM Transactions on Interactive Intelligent Systems (TiiS)* 2.1, pp. 1–28.

Stahl, Annette, Christian Schellewald, Øyvind Stavdahl, Ole Morten Aamo, Lars Adde, and Harald Kirkerod (2012). 'An optical flow-based method to predict infantile cerebral palsy.' In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 20.4, pp. 605–614.

Sun, Ke, Bin Xiao, Dong Liu, and Jingdong Wang (2019). 'Deep high-resolution representation learning for human pose estimation.' In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5693–5703.

Tacchino, Chiara, Martina Impagliazzo, Erika Maggi, Marta Bertamino, Isa Blanchi, Francesca Campone, Paola Durand, Marco Fato, Psiche Giannoni, Riccardo Iandolo, et al. (2021). 'Spontaneous movements in the newborns: a tool of quantitative video analysis of preterm babies.' In: *Computer Methods and Programs in Biomedicine* 199, p. 105838.

Tan, Mingxing and Quoc Le (2019). 'Efficientnet: Rethinking model scaling for convolutional neural networks.' In: *International Conference on Machine Learning*. PMLR, pp. 6105–6114.

Tekin, Bugra, Artem Rozantsev, Vincent Lepetit, and Pascal Fua (2016). 'Direct prediction of 3d body poses from motion compensated sequences.' In: *Pro-*

*ceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 991–1000.

Thorp, Elias B, Farnaz Abdollahi, David Chen, Ali Farshchiansadegh, Mei-Hua Lee, Jessica P Pedersen, Camilla Pierella, Elliot J Roth, Ismael Seáñez Gonzáles, and Ferdinando A Mussa-Ivaldi (2015). 'Upper body-based power wheelchair control interface for individuals with tetraplegia.' In: *IEEE transactions on neural systems and rehabilitation engineering* 24.2, pp. 249–260.

Tome, Denis, Chris Russell, and Lourdes Agapito (2017). 'Lifting from the deep: Convolutional 3d pose estimation from a single image.' In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2500–2509.

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

Tsuji, Toshio, Shota Nakashima, Hideaki Hayashi, Zu Soh, Akira Furui, Taro Shibanoki, Keisuke Shima, and Koji Shimatani (2020). 'Markerless Measurement and evaluation of General Movements in infants.' In: *Scientific reports* 10.1, pp. 1–13.

Tu, Kun, Jian Li, Don Towsley, Dave Braines, and Liam D Turner (2019). 'gl2vec: Learning feature representation using graphlets for directed networks.' In: *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining*, pp. 216–221.

Vafadar, Saman, Wafa Skalli, Aurore Bonnet-Lebrun, Marc Khalifé, Mathis Renaudin, Amine Hamza, and Laurent Gajny (2021). 'A novel dataset and deep learning-based approach for marker-less motion capture during gait.' In: *Gait & Posture* 86, pp. 70–76.

Van Hamersveld, Koen T, Perla J Marang-van de Mheen, Lennard A Koster, Rob GHH Nelissen, Sören Toksvig-Larsen, and Bart L Kaptein (2019). 'Marker-based versus model-based radiostereometric analysis of total knee arthroplasty migration: a reanalysis with comparable mean outcomes despite distinct types of measurement error.' In: *Acta orthopaedica* 90.4, pp. 366–372.

Vapnik, Vladimir (2013). *The nature of statistical learning theory*. Springer science & business media.

Varol, Gül, Ivan Laptev, Cordelia Schmid, and Andrew Zisserman (2021). 'Synthetic Humans for Action Recognition from Unseen Viewpoints.' In: *IJCV*.

*Vicon*. `https://www.vicon.com/`. Accessed: 2021-11-01.

Voulodimos, Athanasios, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis (2018). 'Deep learning for computer vision: A brief review.' In: *Computational intelligence and neuroscience* 2018.

Wade, Logan, Laurie Needham, Polly McGuigan, and James Bilzon (2022). 'Applications and limitations of current markerless motion capture methods for clinical gait biomechanics.' In: *PeerJ* 10, e12995.

Wang, Zhecan, Jian Zhao, Cheng Lu, Fan Yang, Han Huang, Yandong Guo, et al. (2020). 'Learning to detect head movement in unconstrained remote gaze estimation in the wild.' In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3443–3452.

Wei, Shih-En, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh (2016). 'Convolutional pose machines. computer vision and pattern recognition (cvpr).' In: *2016 IEEE Conference on*. Vol. 2.

Whittle, Michael W (1996). 'Clinical gait analysis: A review.' In: *Human Movement Science* 15.3, pp. 369–387.

— (2014). *Gait analysis: an introduction*. Butterworth-Heinemann.

Wren, Tishya AL, Carole A Tucker, Susan A Rethlefsen, George E Gorton III, and Sylvia Õunpuu (2020). 'Clinical efficacy of instrumented gait analysis: Systematic review 2020 update.' In: *Gait & posture* 80, pp. 274–279.

Xiao, Bin, Haiping Wu, and Yichen Wei (2018). 'Simple baselines for human pose estimation and tracking.' In: *Proceedings of the European conference on computer vision (ECCV)*, pp. 466–481.

Xu, Xu and Raymond W McGorry (2015). 'The validity of the first and second generation Microsoft Kinect™ for identifying joint center locations during static postures.' In: *Applied ergonomics* 49, pp. 47–54.

Yakubovskiy, Pavel (2019). *Segmentation Models*. https://github.com/qubvel/segmentation_models.

Yang, Yi and Deva Ramanan (2012). 'Articulated human detection with flexible mixtures of parts.' In: *IEEE transactions on pattern analysis and machine intelligence* 35.12, pp. 2878–2890.

Zhang, Zhe, Chunyu Wang, Weichao Qiu, Wenhu Qin, and Wenjun Zeng (2021). 'AdaFuse: Adaptive Multiview Fusion for Accurate Human Pose Estimation in the Wild.' In: *International Journal of Computer Vision* 129.3, pp. 703–718.

Zhang, Zhengyou (2000). 'A flexible new technique for camera calibration.' In: *IEEE Transactions on pattern analysis and machine intelligence* 22.11, pp. 1330–1334.

Zhao, Hengshuang, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia (2017). 'Pyramid scene parsing network.' In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2881–2890.

Zheng, Ce, Wenhan Wu, Taojiannan Yang, Sijie Zhu, Chen Chen, Ruixu Liu, Ju Shen, Nasser Kehtarnavaz, and Mubarak Shah (2020). 'Deep learning-based human pose estimation: A survey.' In: *arXiv preprint arXiv:2012.13392*.

Zhou, Huiyu and Huosheng Hu (2008). 'Human motion tracking for rehabilitation—A survey.' In: *Biomedical signal processing and control* 3.1, pp. 1–18.

Zhou, Xingyi, Qixing Huang, Xiao Sun, Xiangyang Xue, and Yichen Wei (2017). 'Towards 3d human pose estimation in the wild: a weakly-supervised approach.' In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 398–407.

Zhou, Xingyi, Xiao Sun, Wei Zhang, Shuang Liang, and Yichen Wei (2016). 'Deep kinematic pose regression.' In: *European Conference on Computer Vision*. Springer, pp. 186–201.

Zhou, Xingyi, Dequan Wang, and Philipp Krähenbühl (2019). 'Objects as points.' In: *arXiv preprint arXiv:1904.07850*.

# Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements.

*Genova, Italy*
*April 2022*

Matteo Moro