Monocular 3D Human Pose Estimation From Thermographic Images Using Neural Networks

Monokulare 3D-Posenschätzung von Menschen aus Thermografie-Bildern mit Hilfe neuronaler Netze

Bachelor thesis by Julian Imhof
Date of submission: December 18, 2023

1. Review: Dr.-Ing. (habil.) Stefan Göbel
2. Review: Sebastian Dill
Darmstadt

TECHNISCHE UNIVERSITÄT DARMSTADT

Computer Science Department

Serious Games Group

Erklärung zur Abschlussarbeit gemäß § 22 Abs. 7 APB TU Darmstadt

Hiermit erkläre ich, Julian Imhof, dass ich die vorliegende Arbeit gemäß § 22 Abs. 7 APB der TU Darmstadt selbstständig, ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt habe. Ich habe mit Ausnahme der zitierten Literatur und anderer in der Arbeit genannter Quellen keine fremden Hilfsmittel benutzt. Die von mir bei der Anfertigung dieser wissenschaftlichen Arbeit wörtlich oder inhaltlich benutzte Literatur und alle anderen Quellen habe ich im Text deutlich gekennzeichnet und gesondert aufgeführt. Dies gilt auch für Quellen oder Hilfsmittel aus dem Internet.

Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Mir ist bekannt, dass im Falle eines Plagiats (§ 38 Abs. 2 APB) ein Täuschungsversuch vorliegt, der dazu führt, dass die Arbeit mit 5,0 bewertet und damit ein Prüfungsversuch verbraucht wird. Abschlussarbeiten dürfen nur einmal wiederholt werden.

Bei einer Thesis des Fachbereichs Architektur entspricht die eingereichte elektronische Fassung dem vorgestellten Modell und den vorgelegten Plänen.

Darmstadt, December 18, 2023

J. Imhof

Contents

Abstract

In recent years, the popularity of Human Pose Estimation (HPE) has grown due to research, practical applications, and technological advancements. HPE refers to the process of estimating the pose of the human body from an image, which is a difficult task due to the possible degrees of freedom, appearance variability, and poses. The task is currently addressed utilizing Convolutional Neural Networks (CNNs), which have proven capable of acquiring the intricate mapping between images and poses. The CNN methodology has contributed to numerous HPE techniques based on diverse image types. While most of these methods use RGB images, some techniques utilize depth images or a combination of both. In this study, we investigate the potential of thermographic images for HPE. We demonstrate the implementation of a linear transformation to bypass the need for proprietary calibration algorithms. Several preprocessing methods are utilized to tailor thermographic images to meet CNN input prerequisites. We address the absence of publicly accessible datasets containing thermal images by creating a new dataset. Our procedure is evaluated using this dataset, and its effectiveness is established. Despite the challenges, implementing thermographic images as a part of HPE shows to be a promising approach.

1 Introduction

This chapter gives an overview of the work. It starts with a general introduction to the topic of human pose estimation and its applications in the medical field. Then, it introduces the specific topic of monocular human pose estimation from thermographic images and its potential applications. Finally, it outlines the structure of this work.

1.1 Human Pose Estimation

HPE is a computer vision task that aims to predict a person's body configuration from images or videos by inferring critical points, such as limb and joint positions, and generating a skeletal model. This task is considered challenging due to the complex nature of the human body's range of motion, potential occlusions, and the variable factors of lighting and clothing.

HPE is of significant importance in the medical field because of its potential to serve various purposes, including physical therapy, education, and prosthetics. During surgical simulations, HPE can analyze the surgeon's posture and movements compared to expected behaviors [1]. Furthermore, it can be used in prosthetics to design more effective and adaptable devices for patients [2]. Early diagnosis of diseases such as Parkinson's can be critical, especially since changes in gait and posture are often the earliest indicators. Monitoring a person's movements can enable the early detection of abnormalities, leading to timely intervention [3]. Physiotherapy can utilize tracking technology to monitor patients' progress in recovery. Furthermore, it can provide valuable recommendations for enhancing exercise routines that can facilitate the recovery process [4]. Additionally, it can accurately evaluate the performance of patients during prescribed exercise regimens by offering feedback to ensure precise execution [5].

1.2 Monocular Human Pose Estimation

Monocular Three-Dimensional Human Pose Estimation (M3DHPE) is a type of HPE that uses a single RGB camera as an input source. This approach differs from other methods that employ multiple cameras or depth sensors. The M3DHPE approach is tasked with inferring a 3D pose from 2D images, which presents unique challenges such as depth ambiguities and occlusions.

One of the primary obstacles of M3DHPE in comparison to simple monocular 2D HPE is the shift in domains concerning both appearance and pose space, which typically impacts model performance detrimentally. This challenge is significantly prominent as three-dimensional human data is generally gathered in a controlled laboratory setup [6]. Another impediment is the inherent uncertainty between two-dimensional and three-dimensional perspectives, making it arduous to estimate posture accurately from a singular viewpoint [7].

Despite the challenges posed by complex multi-person scenarios, M3DHPE offers numerous benefits. One such benefit is its robustness and functionality in real-life applications. Moreover, Liu *et al.* showed in their 2022 work how it can be trained using zero real 3D human pose data, which becomes advantageous when such data is not readily available [6]. It also enhances the existing datasets with increased diversity in terms of poses, human appearances, clothing, occlusions, and viewpoints [8].

1.3 Thermographic Imaging

Thermographic imaging utilizes specialized infrared cameras to accurately capture heat signatures passively in the form of infrared radiation. Its extensive use in the medical field stems from its radiation-free and non-invasive diagnosis and temperature measurement capabilities [9].

One of the most prominent advantages of thermographic imaging is its non-invasive character. Thermographic imaging can be performed without physical contact, unlike other diagnostic procedures that may require incision or touch [10]. This aspect makes it an ideal tool for patients who are sensitive to touch or find other diagnostic procedures uncomfortable.

Thermographic imaging is an economical and advantageous diagnostic technique. It proves to be more affordable than other diagnostic imaging procedures [11, 12]. This promotes wider accessibility, enabling early detection of various health conditions.

Moreover, thermographic imaging facilitates the mapping of disease patterns within the body. Most diseases exhibit heat distribution patterns in the body, which can be detected for diagnosing various health issues by medical professionals [10]. This diagnosis approach benefits patient care and also promotes cost savings through its reliance on inexpensive equipment.

Additionally, the digitization of thermographic imaging enables easy sharing of data among medical experts or with patients themselves [10]. This encourages collaboration amongst healthcare professionals and promotes patient involvement in their own healthcare.

The most commonly utilized area for medical thermography is detecting breast cancer [13–16]. Early detection of such conditions can notably enhance treatment outcomes and elevate survival rates. Other applications include the detection of cardiac diseases [17], inflammatory diseases [18–20], diabetes and its symptoms [21, 22], skin diseases [23, 24], and many more [25, 26].

1.4 Monocular Human Pose Estimation From Thermographic Images

Combining thermography and M3DHPE may prove to be an effective tool in medical and health-related fields. Thermography allows for a detailed temperature distribution analysis of the human body, which is useful in identifying medical conditions such as inflammation and poor circulation. When paired with M3DHPE, it can provide a more comprehensive understanding of an individual's physical state. For instance, the technology can detect abnormal body postures and elevated body temperatures simultaneously, leading to early detection of certain diseases and optimization of treatment. This application may also prove useful in sports medicine for injury prevention and performance enhancement by identifying overheating areas and analyzing body postures during physical activities.

This works hence aims to provide a pipeline to perform M3DHPE on thermographic images and evaluate the results.

The remainder of this work is organized as follows. Chapter 3 discusses related research on HPE and Thermographic Human Recognition (THR). Chapter 4 introduces the methods for acquiring and preparing data, evaluating results, conducting M3DHPE, and fine-tuning. Chapter 5 analyzes the plausibility and accuracy of the results. Finally, chapter 6 outlines the significance of the results and potential avenues for advancing this research topic.

2 Fundamentals

This chapter outlines fundamental requirements for comprehending the upcoming work. First, it describes the technicalities and operations of the HPE frameworks considered in this work before doing the same for thermography.

2.1 Human Pose Estimation

There are numerous frameworks and approaches for HPE. Deep learning using CNNs has become the dominant approach in recent years, mainly due to its superior performance. As this work focuses on M3DHPE, only approaches that can infer 3D joint locations are considered. The following three approaches are considered: BlazePose, AlphaPose, and MotionBERT. These approaches were chosen because they are recent, State Of The Art (SOTA) approaches offering theoretical benefits and promising practical applications.

2.1.1 BlazePose

BlazePose [27], a HPE model developed by Bazarevsky *et al.*, has been incorporated into Mediapipe [28], Google's collection of computer vision models. The model can detect 33 keypoints on the human body, for a superset of the ones used by BlazeFace [29], BlazePalm [30], and Coco [31]. These additional keypoints provide essential information regarding the location, scale, and rotation of the face, hands, and feet.

BlazePose consists of two stages: a detector that identifies the location of individuals in the image and a CNN that infers 2D and 3D human poses based on the original image and a Region of interest (ROI). The x and y coordinates of both the 2D and 3D joint locations are given in pixel units, while the 3D depth values are on a similar scale as the x and y coordinates. However, this similarity may not be reliable, as the depth values are obtained by a different method than the x and y coordinates. Additionally, the 3D joint locations are centered around the hip center.

Detector

BlazePose uses BlazeFace as its detector. BlazeFace is a lightweight face detector based on the Single Shot MultiBox Detector (SSD) framework [32]. It was developed with mobile devices in mind, making it lightweight and fast. BlazeFace is a CNN composed of one 2D 5×5 convolution layer with a stride of 2, followed by five what the authors call *Single BlazeBlocks*, and six *Double BlazeBlocks*. The Single BlazeBlock comprises a 5×5 depthwise convolution layer with a stride of either 1 or 2, followed by a 1×1 convolution layer with a stride of 1. In parallel, the input is passed through a max pooling layer, and the channels are zero-padded to match the number of output channels of the 1×1 convolution layer. The output of both branches is then

added together. The Double BlazeBlock comprises two Single BlazeBlocks in sequence but with only one max pooling layer in the parallel step. The output of the Double BlazeBlock is the sum of the output of the second Single BlazeBlock and the output of the padded max pooling layer. The output of the last Double BlazeBlock can be interpreted as multiple overlapping bounding boxes. These are then combined into a single bounding box by taking a weighted average of the bounding boxes' coordinates, where the weights are the confidence scores of the bounding boxes.

BlazeFace does not only detect faces but also the center of the hip and a bounding box around the entire body. This bounding box is then used as the ROI for the second stage. Using a lightweight face detector as the first stage of a HPE model is a common approach. However, it can lead to problems when the face is occluded. In this case, the detector might not detect the face, and the HPE model cannot infer the pose.

Pose Estimator

The second stage of BlazePose is a CNN that infers 2D and 3D human poses given the original image and a ROI. The exact architecture of the CNN is not specified in the paper. However, the authors state that it is inspired by the Stacked Hourglass approach [33]. The Stacked Hourglass approach is a CNN that uses a sequence of repeated pooling and upsampling steps to produce a heatmap of the input image. The heatmap is a two-dimensional array of values representing the probability of a keypoint being at a specific location in the image. The Stacked Hourglass approach uses skip connections to preserve spatial information. The output of the first downsampling step is passed through a series of convolution layers and then added to the output of the first upsampling step. This process is repeated until the desired resolution is reached. The output of the last upsampling step is then passed through a convolution layer to produce the heatmap.

Dataset

The authors trained BlazePose on a custom dataset consisting of 85×10^3 images of which 25×10^3 show a single person in an fitness exercise pose. The remaining 60×10^3 images show one or few people in various common poses. In all images the people are not occluded and both head and shoulder keypoints are easily annotated. The images were annotated by hand with 2D and 3D keypoint locations.

Variations

BlazePose has three different variants: Lite, Heavy, and Full. Lite offers the quickest inference with the least memory usage. Heavy yields the highest precision with a longer time overhead. Full delivers an intermediate solution between the other two versions. The variants differ in the number of connections between the layers and the number of parameters. The Lite variant has the fewest connections and parameters with 2.7 MFLOP and 1.3 million parameters. The Full variant has 6.9 MFLOP and 3.5 million parameters. The Heavy is not mentioned in the paper but it can be inferred that it has more connections and parameters than the Full variant.

2.1.2 AlphaPose

AlphaPose [34] is another deep learning framework comprised of different models for HPE. These models excels in HPE due to their ability to accurately handle occlusions and track individuals across frames in videos, resulting in exceptional performance in crowded scenes. As this work focuses on M3DHPE, only the model for this task is described in detail. Internally HyberIK [35] is used for M3DHPE.

AlphaPose operates with a stepwise process similar to BlazePose. First, a ROI is acquired using one of three different approaches. The first approach is a human re-identification model that is used to identify individuals across frames. This approach is recommended by the authors, but no paper on the details of the model has been published yet. The second approach is a detector-based approach that uses a modular object detection model, such as YOLOX [36], to acquire a ROI. The third approach uses PoseFlow [37] to track individuals across frames. PoseFlow first detects all people across all frames in a video sequence. Then, it combines spatial conjoint ROIs into a single grouping, called a *pose flow*. Lastly, a non-maximum suppression algorithm is applied to join spatially disjoint *pose flows* and decide which ROI to keep per person per frame.

Secondly, given a ROI, a CNN is utilized to deduce a three-dimensional heatmap, from which essential three-dimensional joint locations are regressed. Simultaneously, the CNN determines shape parameters and joint rotations. Next, all of the previously generated data is processed by HyberIK to solve for the relative rotation of each joint. Finally, utilizing simple forward kinematics allows for the computation of new joint positions, resulting in superior quality outcomes compared to the joint positions estimated in the second phase.

Detector

AlphaPose can utilize multiple object detection models, such as YOLOv3 [38] and YOLOX. YOLOv3 is a CNN that uses a single forward pass to predict bounding boxes and class probabilities for those boxes simultaneously. To prevent duplicate detections, YOLOv3 uses non-maximum suppression. YOLOX is a lightweight object detection model based on YOLOv3. Instead of relying on predefined anchor boxes and computing offsets to predict bounding boxes, YOLOX detects bounding boxes directly. YOLOX uses the same backbone as YOLOv3, Darknet-53 [38], a CNN composed of 53 convolution layers with a kernel size of 3×3 and 1×1 and residual connections. The output of the last convolution layer is passed through a global average pooling layer and then through a fully connected layer to produce the final output. YOLOX's neck comprises a feature pyramid network and a path aggregation network to combine features from different scales. YOLOX uses a decoupled head that separately predicts the class probabilities and the bounding box coordinates. Both heads are composed of multiple convolution layers with a size of 3×3, followed by a 1×1 convolution layer. With these modifications, the authors show that YOLOX achieves a better trade-off between speed and accuracy than YOLOv3 [36].

Pose Estimator

Like BlazePose, AlphaPose with the HyberIK model uses a CNN to infer basic 3D joint locations from a ROI. Further, using fully connected layers, the CNN also predicts shape parameters and joint rotations. As a backbone, HyberIK uses ResNet-34 [39], a CNN composed of a single 7×7 convolution layer, followed by 34 3×3 convolution layers with residual connections. The output of the last convolution layer is passed through a global average pooling layer and then through a fully connected layer to produce the final output. HyberIK uses two heads to separately predict the 3D joint locations, shape parameters, and joint angles. The head for

the 3D joint locations comprises three deconvolution layers, followed by a 1×1 convolution layer. This head predicts a three-dimensional heatmap of the input image, from which the 3D joint locations are obtained using a soft-argmax operation. The head for the shape parameters and joint angles comprises an average pooling layer, followed by two fully connected layers. The shape parameters are used to generate a rest pose, which is then algorithmically transformed using the joint angles and positions to obtain a final posed mesh. The final 3D joint locations are obtained from this posed mesh through simple forward kinematics.

Dataset

To train and evaluate the HyberIK model, Li *et al.* used the 3DPW [40], MPI-INF-3DHP [8], and COCO [31] datasets in total and parts of the Human3.6M [41] dataset each for training and evaluation. The 3DPW dataset contains 60 outdoor video sequences of people performing various activities. The videos were automatically annotated using the inertial measurement units and the video data. The MPI-INF-3DHP dataset comprises 8 individuals in 8 indoor and outdoor scenes performing various activities recorded from 14 different camera viewpoints, resulting in 1.3×10^6 frames. The specific details of the annotation process are not provided. The COCO dataset contains 328×10^3 images, of which 200×10^3 were manually annotated with 2D keypoints. The Human3.6M dataset contains 3.6 million images of 11 individuals performing 17 different activities. The images were recorded from 4 different camera angles, from which the 3D joint locations were inferred.

2.1.3 MotionBERT

MotionBERT [42] is a deep learning model tailored for human motion analysis. It offers a unified pretraining framework that can address several sub-tasks, such as M3DHPE, action recognition based on skeletal structure, and mesh recovery.

MotionBERT differs from the two previous approaches. The Neural Network (NN) only takes a sequence of 2D joint locations as input and infers 3D joint locations. The 2D joint locations can be obtained from a 2D HPE model, such as AlphaPose with the ResNet-50 backbone. This makes the full pipeline similar to the one used by AlphaPose. At first, a ROI is acquired using a detector. Then, a CNN is used to infer 2D joint locations from the image and the ROI. Finally, the 2D joint locations are passed to the MotionBERT model to infer 3D joint locations.

Detector

MotionBERT is not concerned with the detector used for the 2D HPE model. However, when used with AlphaPose, the same detectors as described in section 2.1.2 can be used. In this work, the YOLOX detector is used.

Pose Estimator

In this work, AlphaPose with the FastPost model is used to infer 2D joint locations from the image and the ROI. The FastPost model is a CNN using a modified ResNet-50 backbone as introduced by Dai *et al.* ResNet-50 is a CNN composed of a single 7×7 convolution layer, followed by 50 3×3 convolution layers with residual connections. The output of the last convolution layer is passed through a global average pooling layer and

then through a fully connected layer to extract the required features. FastPost uses deformable convolution layers instead of a regular convolution layer in some of the stages of the ResNet-50 backbone. A deformable convolution layer uses a set of offsets to sample the input feature map at different locations. The offsets are learned during training. This allows the model to learn deformations in the input feature map. [43] The output of the fully connected layer is passed through three Dense Upsampling Convolution (DUC) blocks. A DUC block consists of a single convolution layer, followed by a pixel shuffle layer as introduced by Shi *et al.* [44]. A pixel shuffle layer is a type of upsampling layer that rearranges the elements of the input feature map to produce a feature map with a higher resolution. The output of the last DUC block is passed through a convolution layer to produce a heatmap of the input image. The 2D joint locations are obtained from the heatmap using a two step process. First, an element-wise sigmoid function is applied to the heatmap to obtain the confidence scores of the keypoints. Then, the 2D joint locations are obtained using a global normalization with the sum of the heatmap values as the normalization factor.

In contrast to other approaches, AlphaPose does not require the detector to produce a ROI with a high confidence score. Instead, AlphaPose works around the problem of redundant detections given a low required confidence score by using a non-maximum suppression algorithm to combine multiple overlapping predictions into a single prediction. This allows AlphaPose to detect people even when they are heavily occluded.

3D Joint Location Lifting

The 2D joint locations are passed to the MotionBERT model to infer 3D joint locations. MotionBERT is a NN based on the Transformer architecture proposed by Vaswani *et al.* [45]. The Transformer architecture is a NN that uses attention mechanisms to process data sequences. The Transformer architecture is composed of an encoder and a decoder. The encoder is a stack of identical layers. Each layer consists of a multi-head attention layer and a feed-forward network. The decoder is also a stack of identical layers. Each layer comprises a multi-head attention layer, a feed-forward network, and an encoder-decoder attention layer. The encoder-decoder attention layer uses the encoder output as the key and value and the output of the previous decoder layer as the query. The output of the decoder is passed through a linear layer to produce the final output. [45]

MotionBERT uses a Transformer encoder to infer 3D joint locations from the 2D joint locations. The input to the Transformer encoder is a sequence of 2D joint locations. The output of the Transformer encoder is a sequence of 3D joint locations. The Transformer encoder is composed of a stack of 12 identical layers. Each layer consists of a multi-head attention layer and a feed-forward network. The multi-head attention layer uses the previous layer's output as the key, value, and query. The output of the Transformer encoder is passed through a linear layer to produce the final output. The output of the linear layer is a sequence of 3D joint locations.

Dataset and Training

To train and evaluate the MotionBERT model, Zhu *et al.* used the Human3.6M dataset, as well as the AMASS [46] dataset. The AMASS dataset is a meta dataset that combines 15 different motion capture datasets into a single dataset with over 40 hours of motion capture data.

MotionBERT utilizes a novel pre-training stage to recover 3D motion from incomplete 2D observations with noise. This confers superior insight into spatial and temporal motion patterns. Consequently, the derived data proves highly consistent across these metrics.

2.1.4 Thermographic Human Pose Estimation

All existing approaches are designed to achieve HPE on RGB or grayscale images. Nonetheless, thermographic images are currently not incorporated into these methods, despite their numerous potential benefits.

2.2 Thermography

Thermography captures long wave infrared radiation, typically in the 9–14 micron range of the electromagnetic spectrum. In 1901, Planck published the law of black body radiation, which states that all objects emit infrared radiation, the amount of which is dependent on the object's temperature [47]. This allows to remotely measure the surface temperature of objects using specialized cameras.

Because warm-blooded animals, especially humans, are easily detected by their contrast with cooler environments, thermography holds a special place in military and surveillance applications [48, 49].

Thermography cameras are based on the same principles as regular cameras. They consist of a lens, a sensor, and a display. However, instead of capturing visible light, they capture infrared radiation. The lens focuses the infrared radiation onto the sensor, which converts the radiation into an electrical signal. However, the sensor does not split the incoming radiation into three distinct color channels, as is the case with regular cameras. Instead, it converts the radiation into a single grayscale image. This is achieved by employing a special thermographic image sensor. There are two types of thermographic image sensors: photon detectors and thermal detectors. Photon detectors are based on the photoelectric effect. They convert the incoming radiation into an electrical signal by absorbing photons. To work properly, photon detectors require an active cooling system. Thermographic sensors based on photon detectors can record images with a higher resolution, frame rate, and dynamic range than thermal detectors. However, they are typically more expensive and require more power due to their increased complexity and the active cooling system. Thermal detectors are rely on so called microbolometers. A microbolometer is a thermal sensor that measures the change in resistance of a thin film of typically vanadium oxide or amorphous silicon when heated by infrared radiation. The change in resistance in the microbolometer is then measured by a readout circuit and converted into a digital value. To ensure accuracy, each microbolometer is suspended above the readout circuit by a micro-scale bridge-like structure. [50]

3 Related Work

This chapter presents a comprehensive overview of THR frameworks, including their historical evolution and conceptual approaches. Firstly, it delves into the genesis and necessity of THR. Subsequently, SOTA approaches are explored in detail.

THR pertains to the computer vision field that involves the identification and localization of individuals in thermographic images. It can also cover pose classification and pose estimation.

3.1 Algorithmic Approaches

Early algorithmic methods for detecting and localizing humans in thermographic images primarily rely on grayscale values. In their work, "Real-time tracking of non-rigid objects using mean shift" Comaniciu *et al.* developed an infrared algorithm that detects human targets using the Mean Shift algorithm. The authors utilized the grayscale characteristics of the human body to identify targets, thereby simplifying the tracking problem. The Bhattacharyya coefficient was introduced as a measure of similarity between the current and candidate models [51]. Nanda *et al.* proposed probabilistic templates for identifying pedestrians, accounting for variations in human shape, especially in low contrast scenarios or when body parts are missing [52].

Subsequently, combining thermal features with other human characteristics became popular for human recognition. Fernández-Caballero *et al.* proposes a new method for extracting human ROIs that takes motion into account. The fusion of thermal and motion data reduces false positives and boosts system accuracy [53]. The "Mutual Guidance-Based Saliency Propagation for Infrared Pedestrian Images" paper presents pedestrian detection method that combines two saliency types: Thermal Analysis based Saliency (TAS) and Appearance Analysis weighted Saliency (AAS). The TAS measures pedestrian stability via maximally stable extremal regions, while the AAS extracts pedestrian intensity and shape features. The study also introduces a mutual guidance-based saliency propagation model to integrate saliency features and improve saliency performance [54].

These algorithmic methods were crucial in the advancement of thermal human recognition, yet they are less proficient when compared to their deep learning-based counterparts.

3.2 Deep Leaning Based Approaches

In 2017, Biswas *et al.* introduced a deep learning-based method to detect pedestrians using a linear support tensor machine that depends on a Local Steering Kernel and a Histogram of Oriented Gradients as the mid-level representation of a given input. To enhance speed and accuracy, Biswas *et al.* suggested utilizing the multichannel Discrete Fourier Transform as the detection methodology instead of a sliding window-based technique [55].

Another prevalent task in thermographic image processing is Human Action Recognition (HAR), often with a special focus on fall detection. Akula *et al.* introduced a CNN to predict one of six actions in a given 30-second infrared video [56]. Prior to being passed to the NN, the images are transformed using an mean normalization and scaled to a parameterized size. In 2023, Guo *et al.* presented a dataset on HAR by compiling data from a preexisting set and developing three new classifiers [57]. This was done in the hope of providing a general dataset and benchmark for future work. The newly developed classifiers are based on the popular YOLOF [58], YOLOX [36], and TOOD [59] detectors, respectively.

3.3 Influence on This Work

However, no research has focused on the HPE task in thermographic images, which motivates this work. Consequently, no readily available dataset for this task exists, prompting the creation of a new dataset for HPE in thermographic images. Moreover, a benchmark for future research in the field of HPE in thermographic images is also targeted, akin to the work carried out by Guo *et al.*

4 Methods

This chapter explains the work and how it was carried out. Section 4.1 covers the data collection process, while section 4.2 outlines varied techniques employed to optimally prepare data for different NNs. These NNs and their use are explained in detail in section 4.3. Section 4.4 offers detailed information on evaluating metrics for all methods.

4.1 Data Acquisition

4.1.1 Recording

All data utilized for this work was gathered at TU Darmstadt with an ambient temperature of approximately 25 degrees Celsius. The main tool used was the Optris PI640i [60] thermal camera with the 33° x 25° lens. Two Reolink RLC410 cameras [61] were also employed to record the participants simultaneously from two different angles. Furthermore, some additional recordings were collected using a VarioCAM HD head 900 [62] for reference. The participants were instructed to execute several physical exercises while facing the cameras at different angles. Figure 4.1 provides a schematic top-down view of the camera setup.

Due to the offset of the two frontal cameras, the angle deviation towards the subject is around $\gamma = \arcsin(0.05/2 \cdot \sin(90°)) = 1.433°$. Since this has a negligible effect on the actual data, it is ignored hereafter.

4.1.2 Collected Data

Six participants' performance was recorded while performing five diverse exercises. The exercises were chosen to capture a wide range of movements and to incorporate different body parts. The exercises are listed in table 4.1. As the thermographic camera regularly calibrates itself, the participants were instructed to perform each exercise for around 30 seconds to ensure that the camera was able to capture a complete repetition of each exercise. The participants were also instructed to perform some exercises at different angles to the cameras. The angles were chosen to be $\alpha = 0°, 45°, 90°$ to capture the exercises from the front, in between, and from the side.

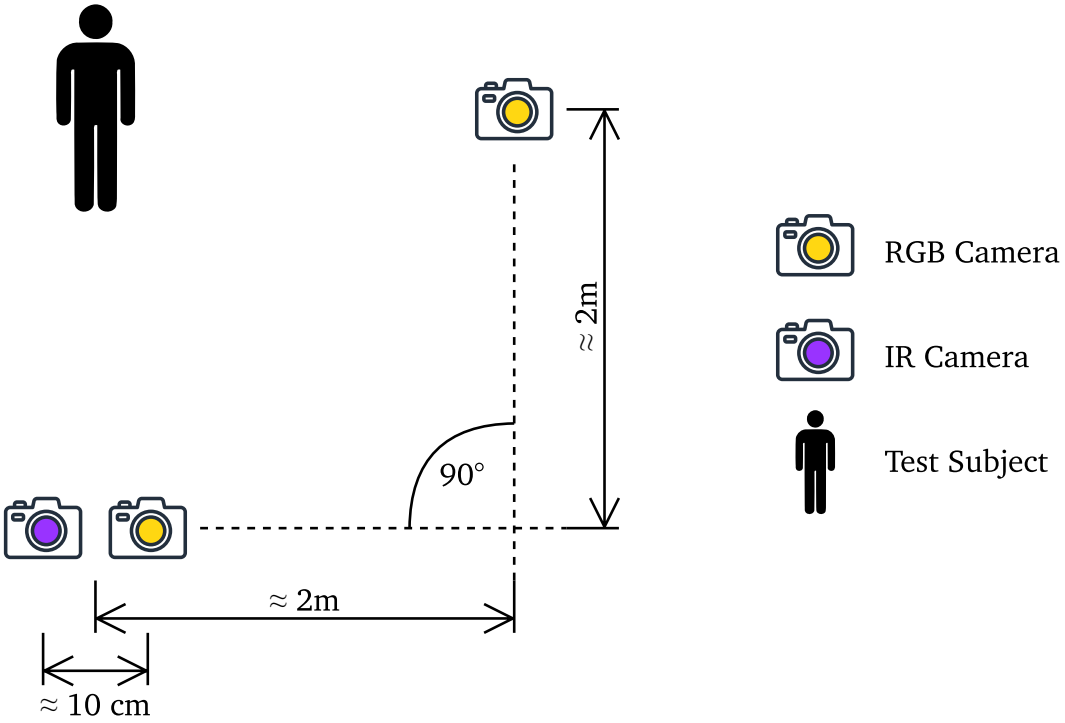
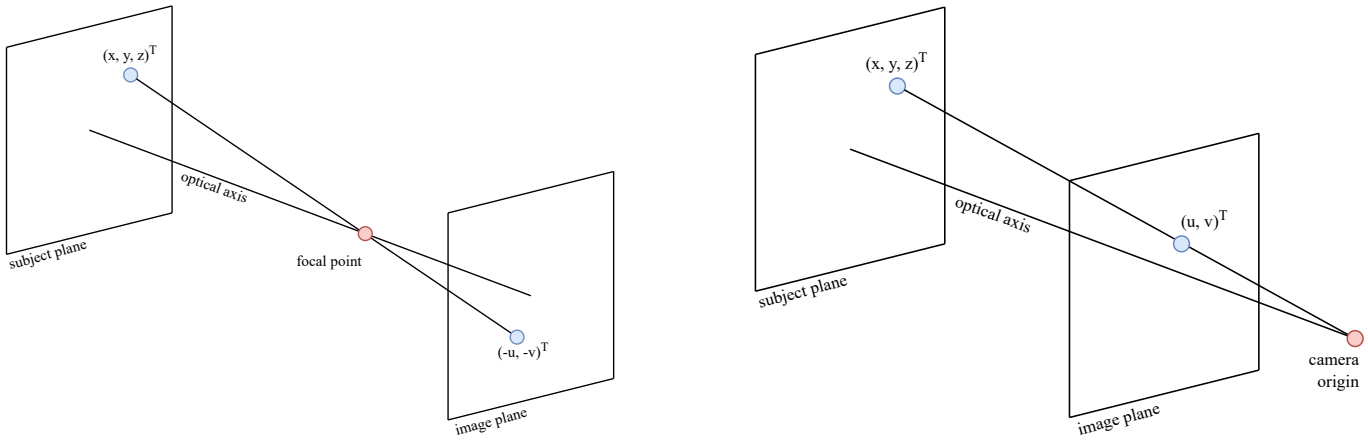


Figure 4.1: Top-down view of the camera setup used for data collection.

Exercise	Recorded angles in °	Description
Dips	$0, 45, 90$	The participant holds onto a chair with their arms behind their back and lifts their body up and down.
Jumping jacks	0	The participant stands upright and jumps while spreading their arms and legs.
Push-ups	45	The participant lies on their stomach on the ground and pushes their body up with their arms.
Sit-ups	45	The participant lies on their back on the ground and lifts their upper body until their elbows touch their knees.
Squats	$0, 45, 90$	The participant stands upright and bends their knees until their thighs are parallel to the ground.

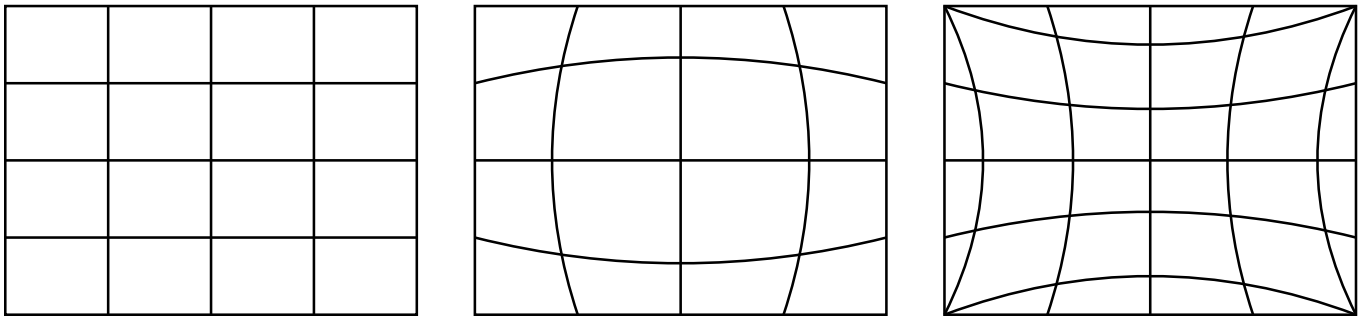
Table 4.1: Exercises performed by the participants.



(a) Abstract model of a pinhole camera

(b) Simplified model of a pinhole camera

Figure 4.2: Pinhole camera models project 3D world coordinates $(x, y, z)^T$ onto 2D image coordinates $(u, v)^T$.



(a) undistorted image

(b) barrel distortion

(c) pincushion distortion

Figure 4.3: Examples of radial distortions

4.1.3 Intrinsic Sensor Calibration

Every camera can be simplified as a pinhole camera, as shown in Figure 4.2a. All light rays pass through the focal point, which projects them onto the image plane. For the sake of mathematical simplicity, this can be further abstracted by moving the image plane between the camera origin and the object. This simplified model is shown in Figure 4.2b.

However, this linear camera model cannot model nonlinear distortions caused by imperfect camera lenses. The most prevalent type of distortion in optical imaging is radial distortion, which causes non-linear changes in magnification with respect to an object's distance from the optical axis. Examples for simple radial distortions can be seen in Figure 4.3. In practice, distortions often comprise a combination of barrel and pincushion distortions as a result of the manufacturer's effort to reduce the overall distortion [63].

A first model for these distortions along with an approach to correct them has been proposed by Heikkila *et al.* [64]. For this correction a multitude of images is required to infer the intrinsic camera parameters.

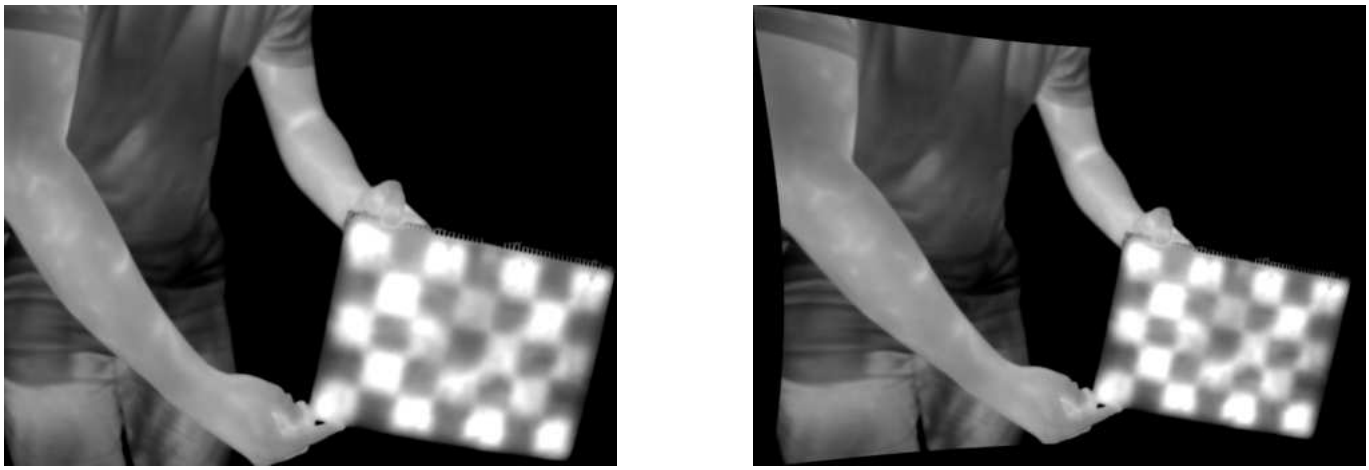
Intrinsic Calibration Using Checkerboard Patterns and OpenCV

OpenCV [65] is a popular open-source computer vision library that provides many algorithms for image processing and computer vision. One of these algorithms is a camera calibration algorithm that can be used to determine the camera's intrinsic parameters. This algorithm requires a set of images of a checkerboard pattern taken from different angles. The algorithm first automatically detects the corners of the checkerboard pattern similar to the Harris corner detector [66]. Specifically, the image first gets converted to grayscale. Then, a gamma correction is applied as a simple histogram equalization. Next, the image is converted to a binary image using an adaptive threshold as proposed by Otsu [67]. Then, the algorithm calculates the centroid and corner candidates for each blob in the image. Then, the algorithm identifies the prospected corners by matching the candidates to a set of criteria. The algorithm then refines the corner positions using Shi *et al.*'s corner refinement algorithm [68]. Next, the algorithm uses the detected corners to calculate the camera's intrinsic parameters. Finally, the algorithm uses the model for radial and tangential distortions first proposed by Brown in 1966 to find the distortion coefficients so that the undistorted calibration grid is minimally skewed across all images [69]. The Brown model is a simple model that assumes that the radial distortion is proportional to the distance from the optical axis. The undistorted pixel coordinates $(x_u, y_u)^T$ are calculated using the following equations, where $(x_d, y_d)^T$ are the distorted pixel coordinates, $(x_c, y_c)^T$ is the center of distortion, $r^2 = (x_d - x_c)^2 + (y_d - y_c)^2$, K_n is the n-th radial distortion coefficient, and P_n is the n-th tangential distortion coefficient.

$$
\begin{aligned}
x_{\mathrm{u}} &= x_{\mathrm{d}} + (x_{\mathrm{d}} - x_{\mathrm{c}})(K_1 r^2 + K_2 r^4 + \cdots) + \Big(P_1 \left(r^2 + 2(x_{\mathrm{d}} - x_{\mathrm{c}})^2\right) \\
&\quad + 2P_2 \left(x_{\mathrm{d}} - x_{\mathrm{c}}\right)\left(y_{\mathrm{d}} - y_{\mathrm{c}}\right)\Big)(1 + P_3 r^2 + P_4 r^4 \cdots) \\
y_{\mathrm{u}} &= y_{\mathrm{d}} + (y_{\mathrm{d}} - y_{\mathrm{c}})(K_1 r^2 + K_2 r^4 + \cdots) + \Big(2P_1 \left(x_{\mathrm{d}} - x_{\mathrm{c}}\right)\left(y_{\mathrm{d}} - y_{\mathrm{c}}\right) \\
&\quad + P_2 \left(r^2 + 2(y_{\mathrm{d}} - y_{\mathrm{c}})^2\right)\Big)(1 + P_3 r^2 + P_4 r^4 \cdots)
\end{aligned}
$$

Intrinsic Calibration Using Checkerboard Patterns and BoofCV

The approach outlined in the previous section is a simple and effective way to calibrate a camera. However, it is not robust to blurry images due to the corner detection algorithm used. BoofCV [70] is another open-source computer vision library that provides similar functionality to OpenCV. However, it uses a different approach to detect the corners of the checkerboard pattern. Instead of using a corner detection algorithm, it uses a pyramidal image processing approach to detect the corners. This approach is more robust to blurry images because it employs multiple novel techniques to detect the corners. First, it uses a pyramidal image processing approach to detect the corners, which implies that the image is scaled down to lower resolutions, and the corners are then detected in the lower-resolution images. This rescaling makes the algorithm more robust to blurry images because the corners are detected in the lower-resolution images, which are naturally less blurry. Then, the algorithm uses a novel technique to detect the corners in the lower-resolution images. This technique is based on the idea that the corners are the intersection of two edges. First, the algorithm finds all possible corners by finding the intersection of two edges. Then, all possible corners that do not occur in the lower-resolution images are filtered out. Finally, all possible corners that do not follow the expected grayscale pattern are filtered out. This filtering is performed to filter out false positives. Lastly, the corners are refined



(a) Distorted frame as output by the camera.

(b) Undistorted frame using camera intrinsics as calculated using BoofCV.

Figure 4.4: Comparison of distorted and undistorted frame using a 5×7 calibration target, the Brown model and solving for three radial coefficients as well as tangential distortions.

using the same algorithm as OpenCV. The algorithm then calculates the camera's intrinsic parameters using the detected corners and the same Brown model as OpenCV.

Actual Calibration

A black and white checkerboard pattern measuring 10×7 squares with a square size of 26.2 mm was utilized as the calibration target for the RGB cameras. The obtained data was subsequently processed using the OpenCV implementation described above to determine the camera intrinsics.

The calibration target for the thermographic camera was created by drawing a 5×7 checkerboard pattern using black ink on a sheet of A4 paper as seen in Figure 4.4. The ink's ability to retain heat longer than the blank paper resulted in the pattern being imprinted as a thermographic signature upon being heated. The recorded images were processed using a lossy linear grayscale transformation as outlined in section 4.2. As the heat penetrates the paper, the edges become slightly blurred. This is a problem for the traditional algorithms implemented in OpenCV. To circumvent these problems BoofCV was used which implements an algorithm proposed by Abeles to handle blurry images [71].

For both the RGB and thermographic cameras, the undistorted images were cropped to remove the black borders introduced by the calibration. This cropping was performed automatically by using the ROI calculated by the `getOptimalNewCameraMatrix` function provided by OpenCV.

4.1.4 Temporal Alignment

The RGB and thermographic cameras do not synchronize, resulting in different frame rates. The RGB cameras record 30 frames per second, while the thermographic camera records 32 frames per second. In order to synchronize the data, the thermographic video's frame rate is reduced to 30 frames per second by dropping

(a) Linear transformed image of the considered camera

(b) Grayscale conversion of reference thermography camera

(c) Difference of the same image section, both normalized

Figure 4.5: Comparison of known good temperature scale to presumed linear transformation

every 16th frame. This results in a theoretical loss of 6.25 % of the data. However, the lost data is negligible because there is no corresponding frame in the RGB data for each dropped frame.

Furthermore, the recordings were started non-simultaneously. This results in different starting points for the RGB and thermographic videos. All videos are trimmed to a common starting point to align the data temporally. This starting point is chosen so the RGB and thermographic videos show simultaneous actions before the participants move. This alignment is achieved by manually identifying recognizable movement changes, such as the participant reaching the lowest point of a squat. The data lost by trimming the videos is insignificant, as only one repetition per exercise per participant is used for the evaluation. This repetition is chosen arbitrarily, ensuring the thermographic camera was not calibrated during the exercise.

4.2 Data Preprocessing

This section mainly focuses on preparing the thermographic data for optimal results. As the RGB data is already in a suitable format, and used to generate the ground truth, it does not require any further processing apart from resizing as described in section 4.2.5. The thermographic data, however, requires a more complex approach.

Thermographic cameras often export data in a radiometric video format, where the raw sensor values, which are 16 bit, are encoded as a raw data stream in a suitable container. In the case of the Optris camera used in this context, the AVI container is used with the pixel format set to YUV2 yuyv422. However, this pixel format does not represent the data's encoding. It is only used because it holds 16 bit per pixel. Using a regular AVI decoder to decode such video will result in images that look correct in contrast to the human eye but have a color tint, as seen in Figure 4.10a. This incorrect representation occurs because the first byte per pixel is interpreted as luminance, while the second is interpreted as alternating blue or red chrominance for adjacent pixels. In order to get a more precise data representation, the stream can be decoded manually using 16 bit unsigned integers for each pixel instead of using regular AVI decoders.

A proprietary algorithm dependent on the camera, lens model, and some calibration data is required to extract the exact temperature values per pixel. Nonetheless, it is enough to estimate this calibration because pose extraction NNs only necessitate a visual depiction of the data, not its exact temperature. Applying a simple linear transformation to the raw data yields a sufficiently accurate result. Figure 4.5c shows the difference between a known calibration from the reference camera and the data linearly transformed from the Optris camera.

The average deviation of grayscale pixel values is approximately 3.1135 %, or approximately $7.9394/255$. This deviation appears to be primarily caused by slight subject movement and a minimal shift in camera angle.

Therefore, a linear transformation can be applied to the previously obtained 16-bit unsigned integers to convert them into luminosity values for a grayscale image.

$$\mathrm{T}_{\text{lossless}}\colon \begin{cases} \{x \in \mathbb{N}_0 \colon x < 65536\} \to [0,1] \\ x \mapsto \left\lfloor \dfrac{\text{pixel}_{i,j,\text{uint16}} - MIN}{MAX} \right\rfloor \end{cases} \tag{4.1}$$

Here, the variables MIN and MAX hold the minimum and maximum values per frame, respectively. Using global MIN and MAX is not possible as the calibration changes on a per frame basis. However, if it was possible, it could improve temporal consistency at a cost of processing the video in two passes, which results in performance overhead.

4.2.1 Clipping

The further temperatures differ from the average human skin temperature, the less helpful they are in detecting human poses. Hence, extreme temperatures might be disregarded. To implement this, the linear transformation can be modified by introducing lower and upper cutoff percentages $OFFSET_L$ and $OFFSET_U$, respectively, such that the resulting transformation is as follows:

$$\begin{aligned} MIN' &\coloneqq MIN + (MAX - MIN) \cdot OFFSET_L \\ MAX' &\coloneqq MAX - (MAX - MIN) \cdot OFFSET_U \\ \mathrm{T}_{\text{lossy}}\colon &\begin{cases} \{x \in \mathbb{N}_0 \colon x < 65536\} \to [0,1] \\ x \mapsto \left\lfloor \min\left(1, \dfrac{\max\left(0, \text{pixel}_{i,j,\text{uint16}} - MIN'\right)}{MAX'}\right) \right\rfloor \end{cases} \end{aligned} \tag{4.2}$$

The variables MIN and MAX are assigned the same as previously. The offsets are defined such that $OFFSET_L, OFFSET_U \in [0,1] \land OFFSET_L + OFFSET_U \leq 1$.

4.2.2 Contrast Limited Adaptive Histogram Equalization

As shown in Figure 4.10b, the linear transformation results in a detailed image with low contrast. To enhance the image further, the Contrast Limited Adaptive Histogram Equalization (CLAHE) algorithm can be applied to the linearly transformed data.

The CLAHE algorithm is a modification of the Adaptive Histogram Equalization (AHE) algorithm, which is itself a modification of the traditional histogram equalization method. These algorithms share the objective of enhancing the contrast of an image. Histogram equalization describes the process of shifting pixel values, typically grayscale or luminosity, to flatten the histogram of a given image. This is performed through a simple five step process. Firstly, a normalized histogram of the pixel intensity values is generated for the image by calculating the probability for each pixel intensity and dividing that by the total amount of pixels. Secondly, the histogram is turned into a Cumulative Distribution Function (CDF) cdf so that the value for each discrete

pixel intensity is the sum of probabilities for all lower or equal pixel intensities. Thirdly, the inverse CDF cdf^{-1} is computed through one of many numerical algorithms. Fourthly, the inverse CDF is used to create a lookup table k' for all pixel intensity values k, such that $k'(k) = \text{cdf}^{-1}(k/(I-1))$, where I is the total number of possible pixel intensity values. Lastly, the lookup table is applied to all pixels, resulting in a histogram with a more uniform distribution.

As the conventional histogram equalization method only enhances the overall contrast, Ketcham introduced what they originally called Local Area Histogram Equalization (LAHE), presently known as Adaptive Histogram Equalization (AHE), which focuses on improving contrast locally [72]. The fundamental principle remains the same, but the histogram is not generated for the entire image. Rather, one is generated for the area directly surrounding each pixel. This method enhances low contrast regions that exhibit less frequent pixel intensities.

However, the AHE algorithm has the tendency to over-amplify noise in homogeneous regions. To prevent this, Zuiderveld proposed the Contrast Limited Adaptive Histogram Equalization algorithm [73]. For efficiency reasons, the image is divided into non-overlapping segments instead of using a sliding window. These segments are subsequently reassembled with a bilinear interpolation. The histogram is clipped at a preset threshold, so that all pixel intensities above the threshold are evenly distributed over all pixel intensities, before the CDF is computed. This process effectively restricts the CDF slope, preventing previously problematic over-amplification.

See Figure 4.6 for a step-by-step demonstration of how the CLAHE algorithm works.

4.2.3 Visualization

The data can be visualized either as a grayscale image or as a color image. As the NNs used in section 4.3 are trained on color images, three color channels are required. The grayscale image is the simplest representation, as the data is already encoded in a single channel.

Alternatively to the grayscale representation, the data can be visualized as a color image. Such colorization is accomplished through the utilization of LUTs. This work utilizes two distinct methodologies to add color to any grayscale image, based on two different concepts for enhancing the outcomes.

HSV Spiral

This approach aims to optimize the use of the 3 Byte pixel values to generate a greater Absolute Basic Color Difference (ABCD) between comparable values while preserving proximity between close ones. The ABCD is defined as the Euclidean distance between two colors within a particular color model space. We do not prioritize absolute perceptual or visual color distance, such as those suggested by Lv *et al.* [74]. To accomplish an increased local ABCD, we utilize a spiral around the outside of the HSV color model to generate the LUT as seen in Figure 4.7. The three dimensional HSV color space $\mathscr{C}$ is defined as $[0,1]^3$.

All tuples $(\theta, r, h)^T$ in the HSV color space describe points in a cylindrical coordinate system, where the hue θ is the angle around the cylinder, the saturation r is the distance from the center of the cylinder, and the value h is the height within the cylinder.

To convert any HSV coordinate $(\theta, r, h)^T$ to the corresponding cartesian coordinates $(x, y, z)^T$, we use the following mapping:

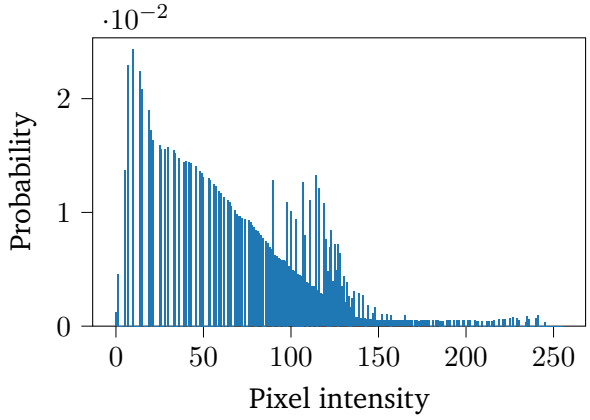
$$P: \begin{cases} \mathscr{C} \to \mathbb{R}^3 \\ \begin{pmatrix} \theta \\ r \\ h \end{pmatrix} \mapsto \begin{pmatrix} r \cdot \cos(2\pi \cdot \theta) \\ r \cdot \sin(2\pi \cdot \theta) \\ h \end{pmatrix} \end{cases} \tag{4.3}$$

To calculate the numerical absolute color distance from two colors c_1 and c_2 within the HSV color model, the following term results:

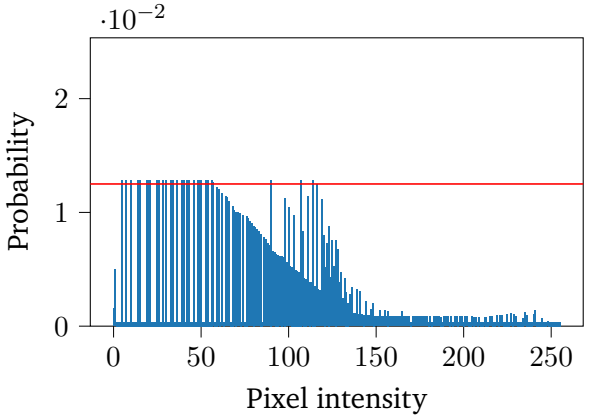
$$\Delta_{\mathscr{C}}: \begin{cases} \mathscr{C} \times \mathscr{C} \to \mathbb{R} \\ (c_1, c_2) \mapsto \|P(c_1) - P(c_2)\|_2 \end{cases}$$

(a) Original low contrast image

(b) Split image into smaller non-overlapping segments; typically the segments are only a few pixels in size, this figure only aims to illustrate the principles

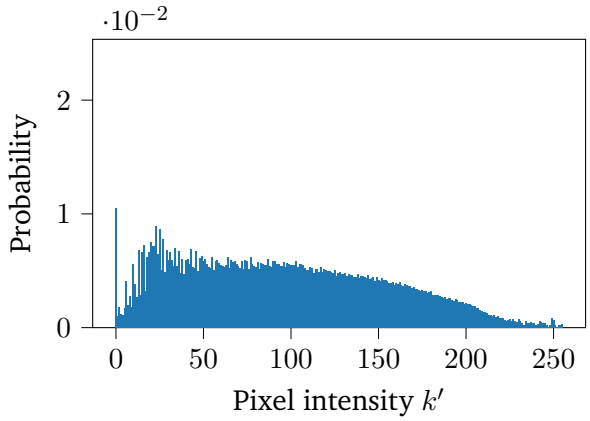


(c) Generate a histogram per segment; this shows the histogram for the bottom left segment



(d) Clip histogram per segment to prevent over-amplification; this shows the histogram for the bottom left segment

Figure 4.6: Step by step visualization of the principles behind the CLAHE algorithm



(e) Shift luminosity value per pixel to equalize the histogram per segment by applying the LookUp Table (LUT) k' based on inverse CDF cdf^{-1}; this shows the histogram for the bottom left segment

(f) AHE on all segments

(g) Join segments back together to one image

(h) Smooth out edges using bilinear interpolation

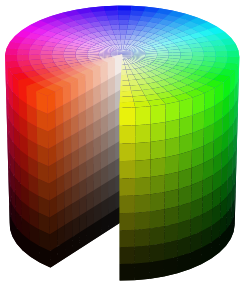
Figure 4.6: Step by step visualization of the principles behind the CLAHE algorithm

The standard grayscale representation of any normalized temperature value x can be expressed in the HSV color model as such:

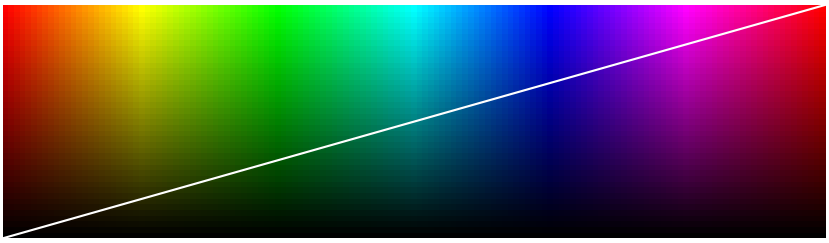
$$\mathrm{HSV}_{\mathrm{gray}}: \begin{cases} [0,1] \to \mathscr{C} \\ x \mapsto (0,0,x)^T \end{cases} \tag{4.4}$$

(4.3) and (4.4) yield the following function to determine ABCD per temperature value difference $x \in [0,1]$ such that $\exists v \in [0,1] \land v + x \in [0,1]$:

$$\Delta_{\mathrm{gray}}: \begin{cases} [0,1] \to \mathbb{R} \\ x \mapsto \Delta_{\mathscr{C}} \left(\mathrm{HSV}_{\mathrm{gray}}(v), \mathrm{HSV}_{\mathrm{gray}}(v+x) \right) \end{cases}$$



(a) Visualization of the HSV color space at discrete positions



(b) Unrolled outer surface of the HSV color space with a white line marking the spiral LUT

(c) Resulting LUT

Figure 4.7: Unrolled HSV color-space surface

$$
\begin{aligned}
&\Delta_{\text{gray}}(x) = \Delta_{\mathscr{C}}\left(\text{HSV}_{\text{gray}}(v), \text{HSV}_{\text{gray}}(v+x)\right) \\
&= \Delta_{\mathscr{C}}\left((0,0,v)^T, (0,0,v+x)^T\right) = \left\| P\left((0,0,v)^T\right) - P\left((0,0,v+x)^T\right) \right\|_2 \\
&= \left\| (0,0,v)^T - (0,0,v+x)^T \right\|_2 = \left\| (0,0,x)^T \right\|_2 = \sqrt{0+0+x^2} = x
\end{aligned}
$$

$$
\implies \Delta_{\text{gray}}(x) = x \tag{4.5}
$$

The mapping for the color on the spiral LUT given a normalized temperature value x is defined as follows:

$$
\text{HSV}_{\text{spiral}}: \begin{cases} [0,1] \to \mathscr{C} \\ x \mapsto (x,1,x)^T \end{cases} \tag{4.6}
$$

Thus, (4.3) and (4.6) imply the following mapping for the ABCD given a normalized temperature difference $x \in [0,1]$ such that $\exists v \in [0,1] \land v + x \in [0,1]$:

$$
\Delta_{\text{spiral}}: \begin{cases} [0,1] \to \mathbb{R} \\ x \mapsto \Delta_{\mathscr{C}}\left(\text{HSV}_{\text{spiral}}(v), \text{HSV}_{\text{spiral}}(v+x)\right) \end{cases}
$$

$$
\begin{aligned}
&\Delta_{\text{spiral}}(x) = \Delta_{\mathscr{C}}\left(\text{HSV}_{\text{spiral}}(v), \text{HSV}_{\text{spiral}}(v+x)\right) \\
&= \Delta_{\mathscr{C}}\left((v,1,v)^T, (v+x,0,v+x)^T\right) = \left\| P\left((v,1,v)^T\right) - P\left((v+x,1,v+x)^T\right) \right\|_2 \\
&= \left\| \begin{pmatrix} 1 \cdot \cos(2\pi \cdot v) \\ 1 \cdot \sin(2\pi \cdot v) \\ v \end{pmatrix} - \begin{pmatrix} 1 \cdot \cos(2\pi \cdot (v+x)) \\ 1 \cdot \sin(2\pi \cdot (v+x)) \\ v+x \end{pmatrix} \right\|_2 = \left\| \begin{pmatrix} \cos(2\pi \cdot v) - \cos(2\pi \cdot (v+x)) \\ \sin(2\pi \cdot v) - \sin(2\pi \cdot (v+x)) \\ v - (v+x) \end{pmatrix} \right\|_2
\end{aligned}
$$

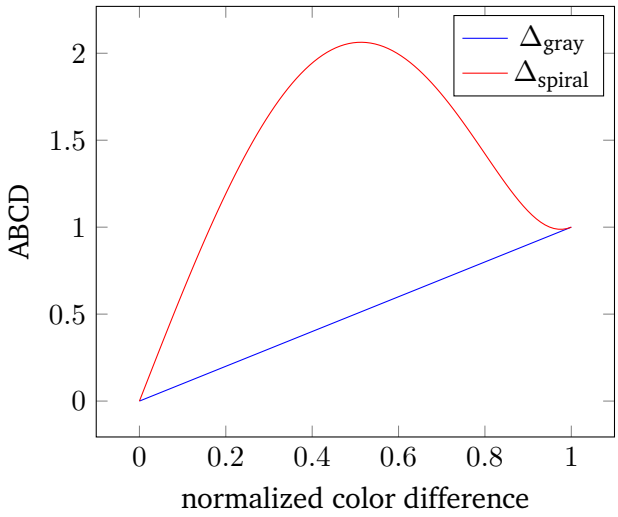


Figure 4.8: Absolute color difference per value difference for Δ_{gray} and Δ_{spiral}.

We substitute a for $2\pi \cdot v$ and b for $2\pi x$.

$$
\begin{aligned}
&\Delta_{\text{spiral}}(x) = \sqrt{(\cos(a) - \cos(a+b))^2 + (\sin(a) - \sin(a+b))^2 + x^2} \\
&= \sqrt{\cos(a)^2 - 2\cos(a)\cos(a+b) + \cos(a+b)^2 + \sin(a)^2 - 2\sin(a)\sin(a+b) + \sin(a+b)^2 + x^2} \\
&= \sqrt{1 + 1 - 2\cos(a)\cos(a+b) - 2\sin(a)\sin(a+b) + x^2} \\
&= \sqrt{2 - 2\cos(a)\left(\cos(a)\cos(b) - \sin(a)\sin(b)\right) - 2\sin(a)\left(\cos(b)\sin(a) + \cos(a)\sin(b)\right) + x^2} \\
&= \sqrt{2 - 2\cos(a)\cos(a)\cos(b) - 2\sin(a)\cos(b)\sin(a) + x^2} \\
&= \sqrt{2 - 2\cos(b)\left(\cos(a)\cos(a) - \sin(a)\sin(a)\right) + x^2} = \sqrt{2 - 2\cos(b) + x^2}
\end{aligned}
$$

Resubstituting $2\pi x$ for b yields $\sqrt{2 - 2\cos(2\pi x) + x^2}$.

$$\implies \Delta_{\text{spiral}}(x) = \sqrt{2 - 2\cos(2\pi x) + x^2} \tag{4.7}$$

Plotting the functions Δ_{gray} and Δ_{spiral} as given by (4.5) and (4.7) can be seen in Figure 4.8. It shows that especially for small value changes a greater absolute color change can be achieved by employing the spiral LUT. However, it should be noted that the absolute difference does not consistently increase. As a result, larger value differences may have a lower absolute color difference compared to smaller value differences. This should not be a problem though, as the main focus is to increase local differences.

Ironblack

The second approach is to employ a color map that more closely resembles common thermographic image visualizations. As not all pre-trained neural network models utilized in this study provide their training data, there is a possibility that this data may comprise thermographic images with a similar color map. This could have permitted the neural network to learn the color scheme as a feature, which may have the potential to enhance the accuracy of the neural network on images with comparable colors. Nevertheless, this is purely

Figure 4.9: Ironblack color map

speculative and cannot be substantiated without access to the training data. To achieve this, we utilize the ironblack color map provided by the open source project GetThermal [75]. The color map is depicted in Figure 4.9.

4.2.4 Full Thermographic Preprocessing Pipeline

The full preprocessing pipeline consists of three distinct steps. Firstly, the raw data is normalized either through the lossless linear transformation (4.1) or lossy linear transformation (4.2). Secondly, the CLAHE algorithm can be applied to the normalized data. Lastly, the data can be visualized either as a grayscale image or as a color image. The color image can be generated either through the HSV spiral LUT (4.6) or the ironblack LUT.

These three steps result in twelve unique representations of the same input data. All twelve representations are depicted in Figure 4.10.

4.2.5 RGB and Infrared Alignment

As the RGB images are captured with a higher field of view than the infrared images, the RGB images need to be cropped and aligned with the infrared images. This is done by using the infrared image as a reference and cropping the RGB image to the same size. The resolution is not altered, only the image section.

4.3 Pose Estimation

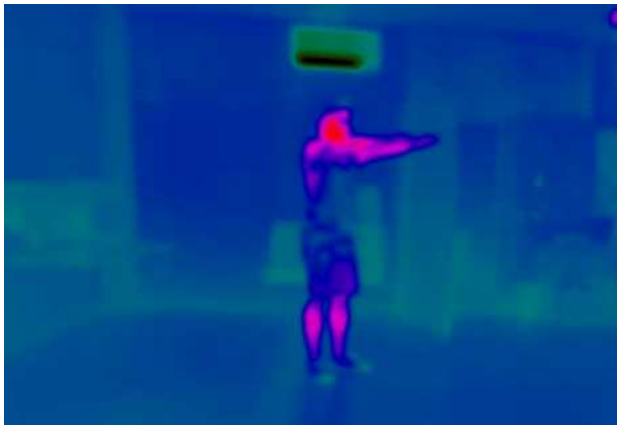
The pre-processed videos are fed into the respective pipelines of the considered HPE approaches, BlazePose, AlphaPose, and MotionBERT.

4.3.1 Comparative Evaluation

For the comparison of the three methods, we use a shared subset of the preprocessed dataset from section 4.2. The videos are processed in their entirety, leading to an obstacle in ground truth-based evaluation as synchronizing inference results with ground truth data is obligatory. Regrettably, not all reference implementations of the three methods offer feedback on missed estimates. Thus, assessment can solely rely on metrics that do not demand ground truth data, like the Average Relative Bone Length Over Time (ARBLOT) metric. While these metrics are insufficient to comprehensively evaluate the approaches, they can compare them and identify the most promising one. These metrics, while inadequate for complete evaluation, can compare the approaches and determine the most promising one. The results of this comparison are presented in Section 5.1.

(a) Incorrect representation as derived from a regular AVI decoder

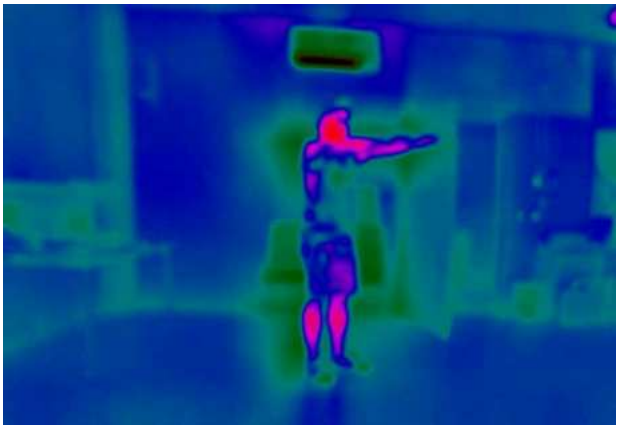
(b) $(\mathrm{T}_{\mathrm{lossless}}, \mathrm{grayscale})$ referred to as Lossless Linear Grayscale (LLGS)



(c) $(\mathrm{T}_{\mathrm{lossless}}, \mathrm{HSV}_{\mathrm{spiral}})$ referred to as Lossless Linear HSV (LLHSV)

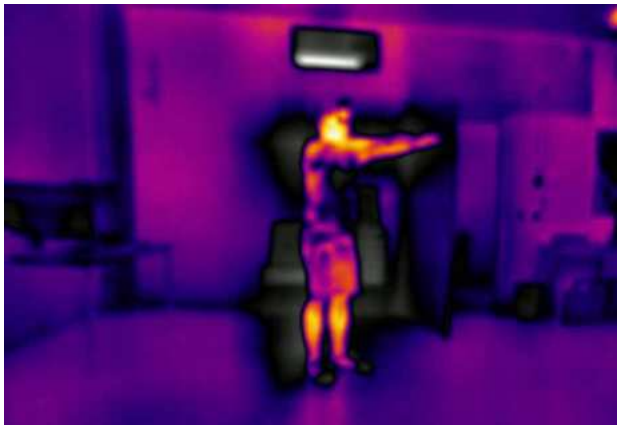
(d) $(\mathrm{T}_{\mathrm{lossless}}, \mathrm{ironblack})$ referred to as Lossless Linear Ironblack (LLIRBL)

(e) $(\mathrm{T}_{\mathrm{lossless}}, \mathrm{CLAHE}, \mathrm{grayscale})$ referred to as Lossless CLAHE Grayscale (LCGS)



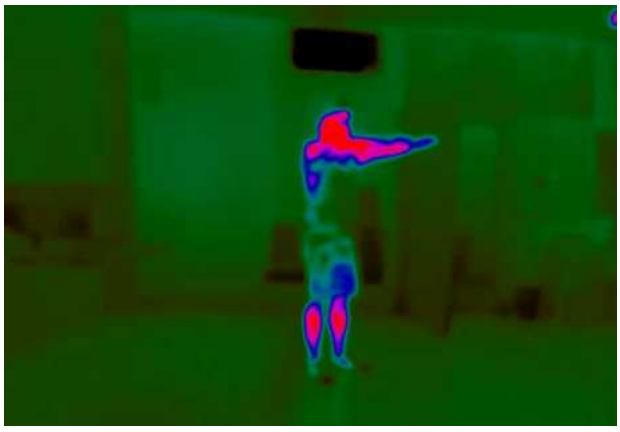
(f) $(\mathrm{T}_{\mathrm{lossless}}, \mathrm{CLAHE}, \mathrm{HSV}_{\mathrm{spiral}})$ referred to as Lossless CLAHE HSV (LCHSV)

Figure 4.10: Overview of all preprocessing approaches taken. The transformation variables were set as follows: $OFFSET_L = 120/255; OFFSET_U = 220/255; \mathrm{CLAHE}_{\mathrm{Limit}} = 2; \mathrm{CLAHE}_{\mathrm{GridSize}} = (8, 8)$



(g) $(\mathrm{T}_{\mathrm{lossless}}, \mathrm{CLAHE}, \mathrm{ironblack})$ referred to as Lossless CLAHE Ironblack (LCIRBL)

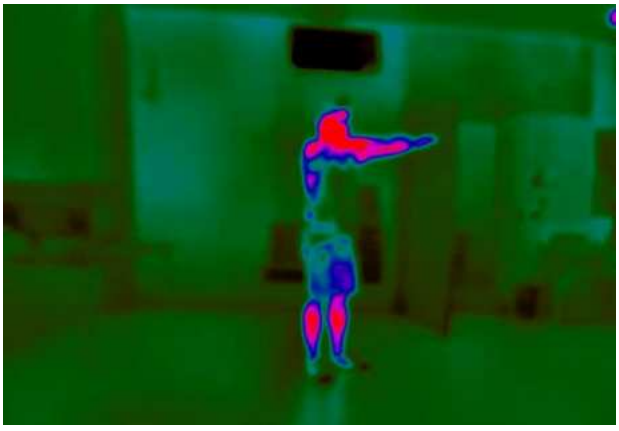
(h) $(\mathrm{T}_{\mathrm{lossy}}, \mathrm{grayscale})$ referred to as Lossy Linear Grayscale (LYLGS)



(i) $(\mathrm{T}_{\mathrm{lossy}}, \mathrm{HSV}_{\mathrm{spiral}})$ referred to as Lossy Linear HSV (LYLHSV)

(j) $(\mathrm{T}_{\mathrm{lossy}}, \mathrm{ironblack})$ referred to as Lossy Linear Ironblack (LYLIRBL)

(k) $(\mathrm{T}_{\mathrm{lossy}}, \mathrm{CLAHE}, \mathrm{grayscale})$ referred to as Lossy CLAHE Grayscale (LYCGS)



(l) $(\mathrm{T}_{\mathrm{lossy}}, \mathrm{CLAHE}, \mathrm{HSV}_{\mathrm{spiral}})$ referred to as Lossy CLAHE HSV (LYCHSV)

Figure 4.10: Overview of all preprocessing approaches taken. The transformation variables were set as follows: $OFFSET_L = 120/255; OFFSET_U = 220/255; \mathrm{CLAHE}_{\mathrm{Limit}} = 2; \mathrm{CLAHE}_{\mathrm{GridSize}} = (8, 8)$

(m) $(\mathrm{T}_{\mathrm{lossy}}, \mathrm{CLAHE}, \mathrm{ironblack})$ referred to as Lossy CLAHE Ironblack (LYCIRBL)

Figure 4.10: Overview of all preprocessing approaches taken. The transformation variables were set as follows: $OFFSET_L = 120/255; OFFSET_U = 220/255; \mathrm{CLAHE}_{\mathrm{Limit}} = 2; \mathrm{CLAHE}_{\mathrm{GridSize}} = (8, 8)$

MediaPipe BlazePose GHUM 3D

The first approach is the BlazePose implementation within the MediaPipe framework. Currently, the framework utilizes BlazePose, as outlined in section 2.1.1, to deduce 2D keypoints from RGB images. Then, a separate model based on GHUM [76] is employed to estimate the 3D pose from these 2D keypoints. The version of MediaPipe Pose used dates back to April 16, 2021. GHUM is a sophisticated process founded on deep learning variational autoencoders [77]. It is designed to evaluate facial expressions, as well as compute body shape and pose estimations. MediaPipe only uses the pose estimation part of GHUM. Xu *et al.* mainly focus on mesh reconstruction for the human body. The exact architecture for the pose estimation is not discussed. The model was trained on the Human3.6M dataset [41] and the CMU [78] dataset.

For the inference, the Python implementation of MediaPipe is used. OpenCV is used to decode the video, convert the frames into NumPY arrays [79], and transform these arrays from their native BGR form into the required RGB format. A single `PoseLandmarker` instance is used to process all frames of a video. For optimal results the heavy model of BlazePose, presented in section 2.1.1, is employed. The `detect_for_video` method is used to process the frames. This method returns all inference results including the 3D keypoints.

AlphaPose

The structure of AlphaPose is explained in detail in section 2.1.2. The inference is performed using the official Python Demo API of AlphaPose for 3D HPE. As detector YOLOX is employed, the pretrained model for HyberIK used in this work is provided by the authors of HyberIK.

MotionBERT

The basics of MotionBERT are detailed in section 2.1.3. Initially, all videos undergo processing via AlphaPose's official Python Demo API. For this task, the YOLOX detector is employed. The pretrained model used in this work for AlphaPose is based on the custom *Halpe26* dataset introduced by the authors of AlphaPose. The

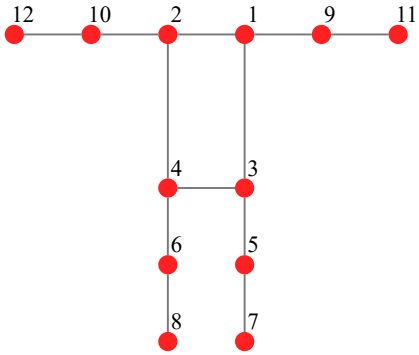


Figure 4.11: Limited keypoint set used for performance analysis.

resulting 2D keypoints are then fed into the reference implementation of the MotionBERT pipeline provided by the authors of MotionBERT.

4.3.2 Detailed Evaluation

Due to its superiority in the comparative evaluation, as presented in section 5.1, the MotionBERT approach is selected for further evaluation. The evaluation is performed on the entire dataset, as outlined in section 4.2.

To properly handle estimation misses the videos are split into single frames and each frame is processed individually. This potentially degrades estimation stability over time, but since the results are mainly evaluated on a per-frame basis, this is not a problem. Furthermore, section 5.2 shows that even with this approach, the results can still be fairly stable over time. The videos are split into individual frames using the default decoder implementation of the `VideoCapture` class from the OpenCV library.

Similar to the comparative evaluation described above, the same provided pretrained model and YOLOX are used in conjunction with the reference Python implementation of AlphaPose and MotionBERT. The results of the evaluation are presented in section 5.3.

4.4 Analysis

The main focuses of this analysis are plausibility and correctness. Considerations regarding memory usage and time consumption are of secondary importance.

Various methods of M3DHPE use different keypoint sets, but a common subset can be used to address this. The reduced set consists of keypoints for only the shoulders, hips, knees, ankles, elbows, wrists, and excludes other body parts. The keypoint data is sorted, as illustrated in Figure 4.11, and then encoded in a CSV file with one line per frame.

4.4.1 Ground Truth Generation

The dataset produced in sections sections 4.1 and 4.2 needs to have ground truth data for some of the evaluation metrics. Consequently, a methodology to generate said data is necessary. Our approach combines the pose estimation findings from the RGB camera's front and side views. Therefore, merging the keypoints from both views results in a simple process. In contrast to the depth direction, the pose estimation in the side-to-side and top-to-bottom direction is much more trivial. Therefore, the views can be merged by simply combining the two sets of keypoints, such that the new keypoint $(x, y, z)^{T\prime}$ is defined as $(x_1, y_1, y_2)^T$, where $(x_1, y_1, z_1)^T$ is the keypoint from the front view and $(x_2, y_2, z_2)^T$ is the keypoint from the side view. For all keypoints, the x coordinate describes the left-to-right direction as seen from the camera, the y coordinate describes the top-to-bottom direction as seen from the camera, and the z coordinate describes the depth direction as seen from the camera. The validity of this approach is verified in section 5.2.

Several metrics quantifying the results of pose estimation are utilized.

4.4.2 Inference Results

The first metric is the Percentage of Inference Results (PIR), which is the proportion of frames for which the pose estimation was successful. This metric is not that useful on its own but can be used to compare the performance of different approaches. It is also helpful to determine whether the pose estimation is stable or whether it fails frequently. Furthermore, it is used by the following quantitative metrics to scale their results.

4.4.3 Normalization

Before calculating any metrics, both the estimated and actual values are normalized linearly using the hip width, ensuring consistent outcomes irrespective of the camera's distance from the subject or other factors. Before normalization, the hip width is first filtered using a one dimensional gaussian kernel with a size of 5 and a standard deviation of 1. The resulting values are then used to normalize the estimated and actual values on a per-frame basis. This work considers two specific metrics.

4.4.4 Keypoint Displacement

Keypoint displacement refers to various metrics based on the Euclidean norm between estimated joint positions and their corresponding ground truth.

The first measure is the Percentage of Correct Keypoints (PCK), which calculates the proportion of keypoints for which the Euclidean distance falls below a certain threshold. Evaluating with a sufficiently large threshold supplies feedback on significantly missed keypoints, such as from obstructed body parts. The threshold is set as a percentage of the normalized hip width, such that a threshold of 0.1 corresponds to 10% of the hip width.

The second metric utilized is the Mean Per Joint Position Error (MPJPE), which calculates the average of the Euclidean distances across all frames and joints. This metric offers a comprehensive means to assess overall performance across various approaches. Due to the normalization, the MPJPE is measured in units of the hip width.

Both of these metrics are frequently employed not just in HPE but also in analogous computer vision tasks [80–84]. PCK, for example, can also be employed in knee surgery as proposed by Marmol *et al.* [85].

4.4.5 Relative Bone Length

In the context of HPE, bone lengths can be assumed to be constant. This can then be utilized to assess the plausibility of the results of a particular model and how stable it is.

On a per-frame basis, the Relative Bone Lengths (RBL) can be compared to anatomical assumptions. For example, the length of the forearm is approximately equal to that of the upper arm, and similarly, the length of the thigh is roughly equal to that of the shin. It is important to note that these proportions may vary between individuals.

Another approach is to evaluate the consistency of the estimated bone lengths over time. However, bone length estimations may vary due to the subject's distance from the camera. One way to solve this issue is to use the bone length ratios instead of the actual bone lengths, which requires ensuring a stable per-frame RBL.

This work utilizes the average ratio of left upper arm length (L_1) to shoulder width (L_2) over time, and the average ratio of right thigh length (L_3) to hip width (L_4) over time to calculate the ARBLOT between two consecutive frames (i and $i+1$), as shown below:

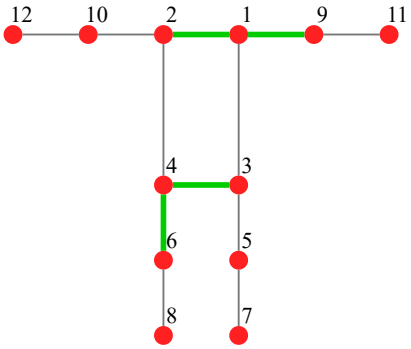


Figure 4.12: Illustration of the bone lengths used for the ARBLOT.

$$A_{1,i} := \frac{L_{1,i}}{L_{2,i}} * \frac{L_{2,i+1}}{L_{1,i+1}}$$
$$A_{2,i} := \frac{L_{3,i}}{L_{4,i}} * \frac{L_{4,i+1}}{L_{3,i+1}}$$
$$ARBLOT_i := \frac{A_{1,i} + A_{2,i}}{2}$$

These bones are chosen because they are expected to be relatively stable over time. Apart from the shoulder width, the other bone lengths represent actual bones in the human body and are thus expected to be constant. The shoulder width is chosen because it is expected to be relatively stable and easy to detect from only a silhouette.

4.4.6 Joint Angle Changes

The final metric examined is changes in joint angles. A graph can be created by calculating the angle between two potentially constructed bones and tracking changes in this angle between frames. This graph can then be used to assess the plausibility of a given approach. For instance, a video of a participant performing squats is expected to display a sinusoidal knee angle, while a static knee angle, is to be expected when analyzing sit ups. However, since this method does not yield measurable numerical outcomes, it can only be applied to provide a general assessment of different approaches.

4.4.7 Mirror Error

As thermographic images do not show significant three dimensional features in bodies with a uniform temperature, it is difficult to make out the orientation of the body. This is especially true for movements that are performed in the sagittal plane. To assess the this shortcoming, the mirror error is introduced. It is defined as the percentage of frames in which the estimation flipped across the frontal plane demonstrates a lower MPJPE than the original estimation. All ground truth based metrics are also evaluated on the corrected estimation. This is done to assess the impact of the mirror error on the estimation quality.

5 Results

This chapter presents the results of the experiments conducted in this work. First, in section 5.1 a single M3DHPE pipeline is selected from the three candidates presented in section 2.1. Then, in section 5.2 the ground truth generation methodology is verified. Finally, in section 5.3 the results of the different preprocessing methods are compared.

5.1 Model Selection

Limited tests were conducted on the LLGS data of the *Squat* 0° scene to select the most promising M3DHPE pipeline among the three candidates presented in section 2.1: BlazePose, AlphaPose, and MotionBERT. The objective of this test was to identify the most promising pipeline given the same non-optimized data. The LLGS data was chosen for this test because it is the closest to the raw thermal data.

For each M3DHPE pipeline, the the knee angle from was calculated for each frame of the video of one participant for the *Squat* 0° scene. The video was processed as a whole and not frame by frame as the concerns raised in section 4.3.2 do not apply to the evaluation of the metric employed in this test. Furthermore, the video contained six repetitions of the exercise. The knee angle is defined as the angle between the line from the hip joint to the knee joint and the line from the knee joint to the ankle joint for the left and right knee respectively. Figure 5.1 shows the bones used for the calculation of the knee angle. A calibration sequence can be seen in all estimations from around frame 280 to 300. The calibration sequence is a sequence of identical frames. It shows as an unchanged knee angle in the graphs with a sudden change at the end of the calibration sequence.

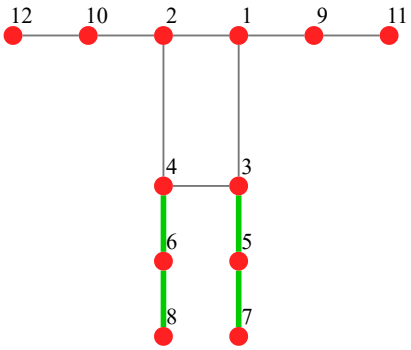


Figure 5.1: Illustration of the bones used for the change in the knee angle.

Figure 5.2 shows the knee angle from per frame as inferred by MediaPipe using BlazePose. It shows the general trend of the knee angle decreasing during the downward movement and increasing during the upward movement. However, the change in the knee angle is not smooth but rather jittery.

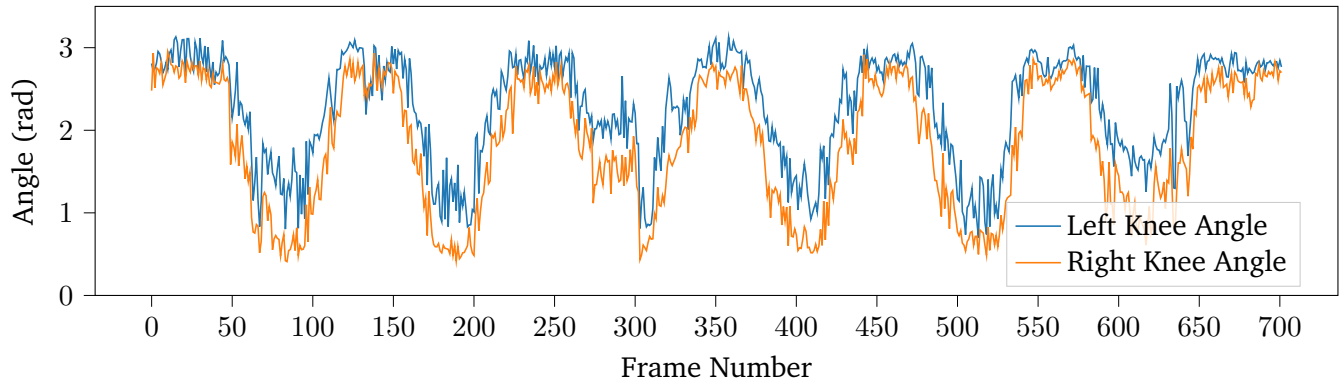


Figure 5.2: Knee angle over time for the *Squat* 0° scene of one participant as inferred by MediaPipe using BlazePose in radians.

Figure 5.3 shows the knee angle from per frame as inferred by AlphaPose. In contrast to MediaPipe, the knee angle shows no recognizable pattern. The angle is mostly wholly random and has no visible correlation with the actual movement of the participant. Furthermore, the changes in the angle between frames regularly exceed 1 rad, which is physically possible but does not fit the actual movement of the participant as it exceeds the expected speed of the movement.

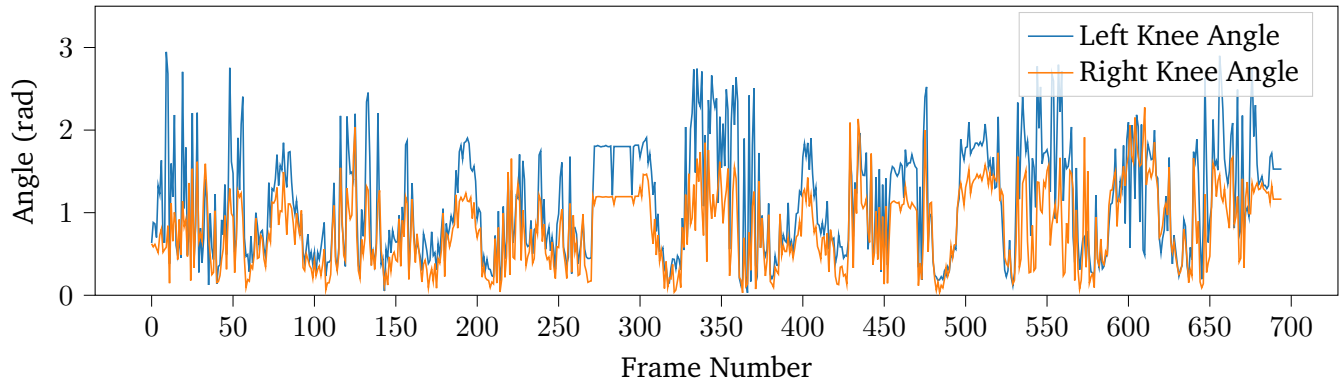


Figure 5.3: Knee angle over time for the *Squat* 0° scene of one participant as inferred by AlphaPose using HyberIK in radians.

Figure 5.4 shows the knee angle per frame as inferred by MotionBERT. The knee angle exhibits the most substantial relationship to the subject's actual movement. The change in knee angle is consistently smooth and closely matches the subject's actual movement. Furthermore, there are mostly no unexpected sudden changes in the knee angle, which fits the actual movement of the participant.

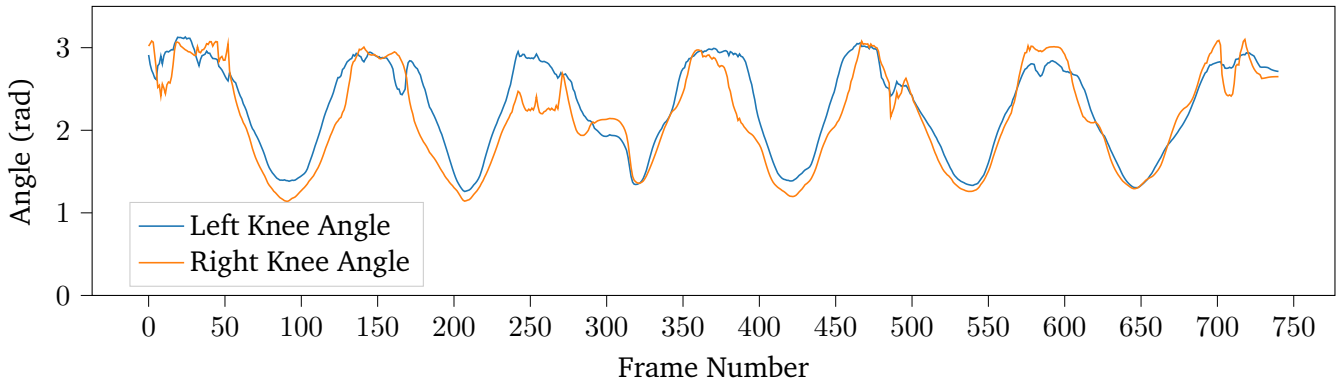


Figure 5.4: Knee angle over time for the *Squat* 0° scene of one participant as inferred by MotionBERT in radians.

Figure 5.6 shows the ratio of the hip width to the thigh length for the *Squat* 0° scene of one participant as inferred by MediaPipe using BlazePose, AlphaPose, and MotionBERT. The ratio of the hip width to the thigh length is calculated as the ratio of the distance between the left and right hip joint to the average distance between the left knee joint and the left hip joint and the right knee joint and the right hip joint. Figure 5.5 shows the bones used for the calculation of this ratio. The ratio is calculated for each frame of the video. Due to the different types of annotations used by the different frameworks, the ratio is not directly comparable. However, the stability of the ratio is comparable. The ratio of the hip width to the thigh length as inferred by MediaPipe is the least stable. AlphaPose is the most stable, and MotionBERT performs better than MediaPipe but worse than AlphaPose. The remarkable stability of AlphaPose is likely due to the architecture of the HyberIK model it employs. As explained in section 2.1.2, HyberIK initially transforms a base skeleton to generate a rest pose skeleton based on estimated shape parameters. Joint angles estimated are then estimated. This approach results in an extremely stable skeleton structure that is utilized for all video frames. The ratio of hip width to thigh length as inferred by AlphaPose cannot be directly compared to other frameworks due to this reason.

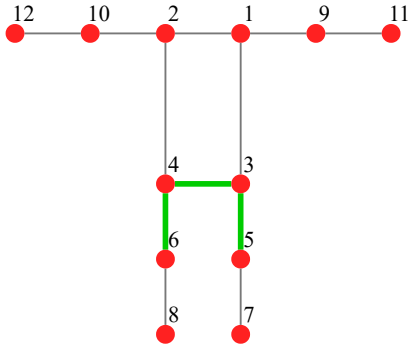


Figure 5.5: Illustration of the bones used for stability analysis

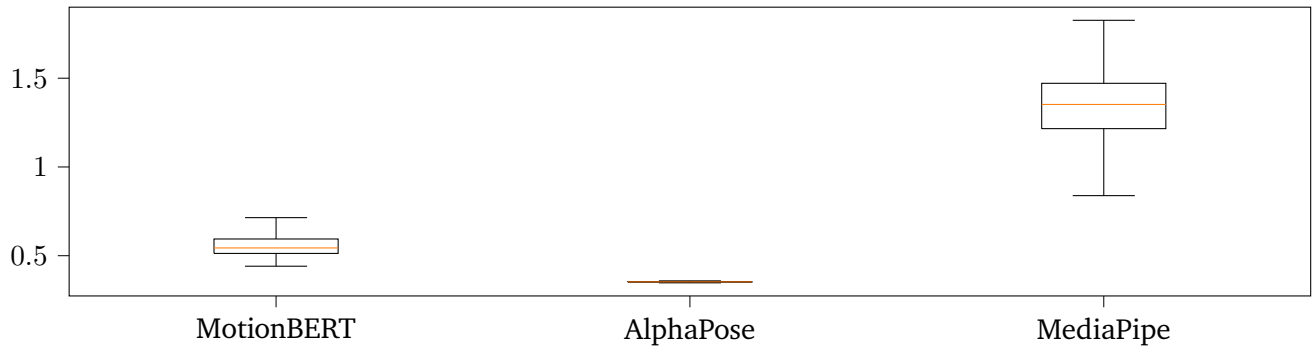


Figure 5.6: Boxplot of the ratio of hip distance to thigh length for the *Squat* 0° scene of one participant as inferred by MediaPipe using BlazePose, AlphaPose using HyberIK, and MotionBERT.

Due to its superior performance in M3DHPE, MotionBERT is the only HPE framework considered in the remainder of this work.

5.2 Ground Truth Generation

To verify that the ground truth generated in section 4.4.1 is valid, the ARBLOT is calculated for each participant and each setting from the front, side, and merged views from the RGB recordings. The quartiles shown in Figure 5.7 directly correlate to the mean and the variance of the three considered estimators of the skeletal stability. The front view has a mean of 1.0045 , the side view has a median of 1.0093 , and the merged view has a median of 1.0026 Since all three means are close enough to the expected value of 1 that they can be assumed to be 1, the ARBLOT metric is an mean-unbiased estimator. Thus, the quality of an estimator can be quantified solely by the variance of the ARBLOT. The front view has a variance of 5.4×10^{-4}, the side view has a variance of 1.61×10^{-3}, and the merged view has a variance of 3.4×10^{-4}. As the merged view has the lowest variance, it is the best estimator of skeletal stability, suggesting that it estimates the joint locations most accurately. This superior performance makes the merged view the best choice to represent the ground truth generation.

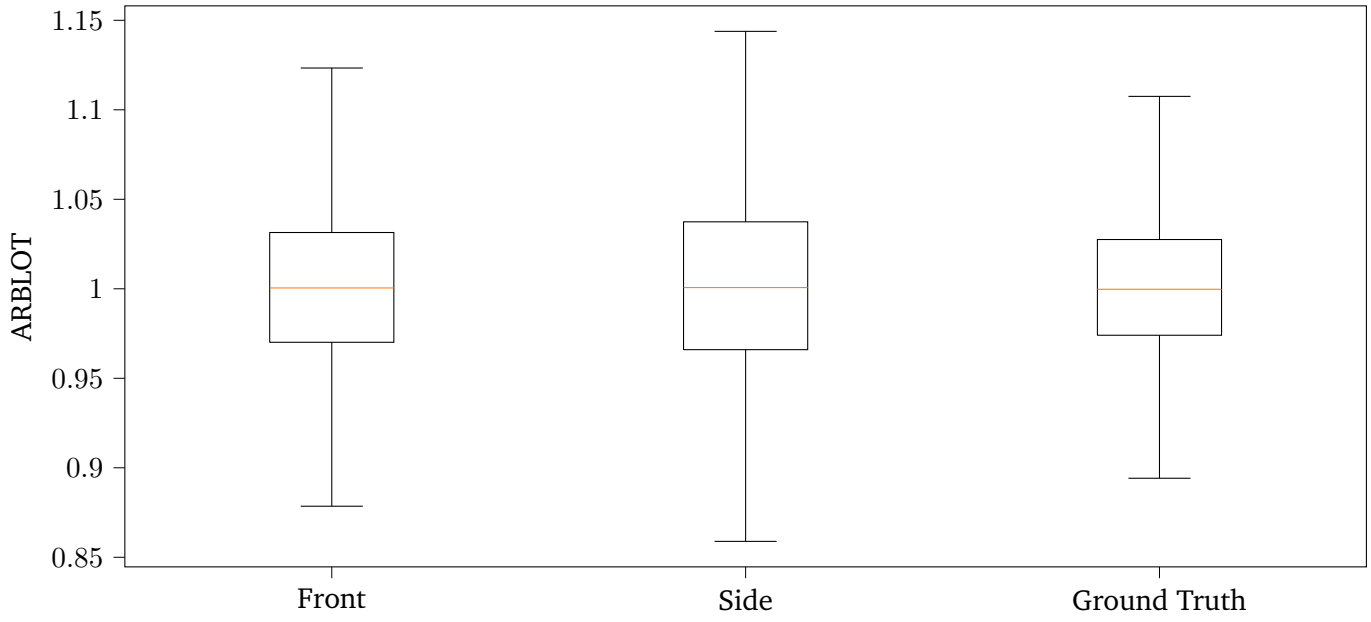


Figure 5.7: Boxplot for the ARBLOT of the front, side, and combined view across all settings and participants. The expected ratio is 1.

5.3 Preprocessing Comparison

For the comparison of the different preprocessing methods, all videos are processed on a per-frame basis. This is in contrast to the proceedings in section 5.1 to address the concerns raised in section 4.3.2.

The results are compared using the PIR, MPJPE, PCK, and ARBLOT metrics. The PIR is calculated as the proportion of frames for which the pose estimation was successful. The MPJPE is calculated as the mean of the Euclidean distance between the estimated and actual key points for each frame. The PCK is calculated as the proportion of key points for which the Euclidean distance between the estimated and actual key point is below a certain threshold. The ARBLOT is calculated as the ratio of the average of two bone length ratios in two consecutive frames. The exact definition of all metrics is given in section 4.4.

Table 5.1 shows the Percentage of Inference Results (PIR), as introduced in section 4.4.2, for each scene and preprocessing method. The PIR is calculated for each scene and participant individually, and then averaged over all scenes and participants. Upon examination of the data, it is clear that the grayscale representation consistently yields the highest average score, and largely the lowest uncertainty. Both colorization methods introduced in section 4.2.3, based on the HSV color spiral and the IronBlack color map, yield worse results. There is no explicit pattern in the degradation of the PIR across the two colorization methods. In most scenes, both approaches yield similar results, in others, one approach is superior to the other. The scenes involving *Push Ups* and *Sit Ups* have a significantly lower PIR compared to all other scenes. This is likely because these are the only scenes where the participant is not standing upright, which is a pose that is not commonly present in most datasets, including the one used to train HyberIK. This also results in a high degree of self-occlusion, which poses a challenge for most M3DHPE pipelines.

	Dips 0°	Dips 45°	Dips 90°	Jumping jack	Push ups	Sit ups	Squat 0°	Squat 45°	Squat 90°
LLGS	**100**	98 ± 4	**100**	100 ± 1	28 ± 22	78 ± 22	99 ± 2	100 ± 1	100 ± 1
LLHSV	**100**	51 ± 47	87 ± 4	83 ± 12	8 ± 8	8 ± 10	90 ± 13	88 ± 12	93 ± 11
LLIRBL	96 ± 6	75 ± 25	99 ± 1	88 ± 9	24 ± 14	34 ± 27	86 ± 24	100 ± 1	97 ± 8
LCGS	**100**	100 ± 1	**100**	**100**	39 ± 24	82 ± 17	**100**	100 ± 1	**100**
LCHSV	**100**	29 ± 33	43 ± 3	75 ± 11	0 ± 1	0	74 ± 23	59 ± 13	65 ± 22
LCIRBL	98 ± 3	36 ± 43	53 ± 26	73 ± 14	0	0	71 ± 26	82 ± 13	57 ± 19
LYLGS	**100**	100 ± 1	**100**	99 ± 2	53 ± 29	**88 ± 17**	**100**	100 ± 1	**100**
LYLHSV	79 ± 29	30 ± 33	31 ± 18	72 ± 25	3 ± 8	2 ± 4	79 ± 29	72 ± 26	70 ± 29
LYLIRBL	99 ± 2	49 ± 40	45 ± 2	63 ± 15	1 ± 3	11 ± 13	82 ± 21	82 ± 28	71 ± 29
LYCGS	**100**	**100 ± 1**	**100**	100 ± 1	**60 ± 23**	83 ± 20	**100**	**100 ± 1**	**100**
LYCHSV	94 ± 9	28 ± 33	21 ± 4	70 ± 19	0 ± 1	0	85 ± 15	71 ± 22	64 ± 33
LYCIRBL	92 ± 12	52 ± 48	41 ± 15	54 ± 7	1 ± 2	8 ± 9	79 ± 24	83 ± 23	61 ± 27

Table 5.1: Mean Percentage of Inference Results (PIR) and standard deviation between participants per scene and preprocessing; higher is better

The Mean Per Joint Position Error (MPJPE) per scene and preprocessing method is shown in table 5.2. The same averaging procedure as for the PIR is applied. The results mostly affirm the findings of the PIR metric in that the two colorization methods decrease the accuracy of the pose estimation. In some cases, notably the from the LLGS to the LLHSV representation in the *Squat* 90° scene, the MPJPE is significantly increased with a tighter uncertainty. Although the pose estimation for the *Push Ups* and *Sit Ups* scenes is successful for only a small proportion of frames, a good MPJPE does not necessarily follow. On the contrary, the MPJPE is significantly higher than in other scenes, which confirms the findings of the PIR metric. It is worth noting that the MPJPE is consistently lowest for the 0° view of the *Squats* scenes, where the participant is facing the camera, among the three different angles. Similarly, less pronounced results are observed for the different angles of the *Dips* scenes.

	Dips 0°	Dips 45°	Dips 90°	Jumping jack	Push ups	Sit ups	Squat 0°	Squat 45°	Squat 90°
LLGS	0.7	0.5 ± 0.2	1.5 ± 0.5	**0.4 ± 0.1**	4.0 ± 1.5	1.7 ± 0.5	**0.5 ± 0.2**	**1.1 ± 0.5**	1.9 ± 2.3
LLHSV	0.7	1.3 ± 1.4	2.1	0.8 ± 0.3	4.2 ± 1.7	1.9 ± 0.2	0.6 ± 0.3	1.3 ± 0.7	**0.9 ± 0.3**
LLIRBL	0.7	0.7 ± 0.5	1.6 ± 0.6	0.6 ± 0.2	4.6 ± 2.7	2.1 ± 0.8	0.9 ± 1.2	1.6 ± 1.0	2.3 ± 3.5
LCGS	**0.5 ± 0.1**	**0.4 ± 0.1**	1.6 ± 0.5	0.4 ± 0.1	4.0 ± 1.4	1.6 ± 0.3	0.5 ± 0.4	1.1 ± 0.8	1.4 ± 1.0
LCHSV	0.8 ± 0.2	2.1 ± 0.2	2.1 ± 0.6	0.7 ± 0.2	4.3		1.1 ± 0.7	1.3 ± 0.2	1.4 ± 0.6
LCIRBL	0.8 ± 0.1	1.1 ± 0.7	1.6 ± 0.2	1.0 ± 0.5			1.2 ± 1.2	1.6 ± 0.9	1.4 ± 0.5
LYLGS	0.6 ± 0.1	0.7 ± 0.5	1.4 ± 0.6	0.7 ± 0.3	3.2 ± 1.4	1.7 ± 0.4	0.7 ± 0.3	1.5 ± 1.2	1.9 ± 2.5
LYLHSV	0.8	1.1 ± 0.4	2.1 ± 1.0	1.3 ± 0.8	4.4 ± 2.7	2.6	1.3 ± 0.6	1.4 ± 1.1	1.5 ± 0.9
LYLIRBL	1.3 ± 0.8	1.8 ± 0.6	1.5 ± 0.7	1.6 ± 0.8	**2.9**	2.3 ± 0.4	0.9 ± 0.4	1.7 ± 1.0	1.5 ± 1.1
LYCGS	0.6 ± 0.1	0.6 ± 0.4	1.5 ± 0.3	0.6 ± 0.2	3.4 ± 1.7	1.7 ± 0.4	0.7 ± 0.4	1.5 ± 1.2	1.8 ± 2.2
LYCHSV	1.0 ± 0.5	1.2 ± 0.8	2.0 ± 0.9	1.3 ± 0.8	4.8		1.3 ± 0.7	1.4 ± 0.9	1.7 ± 1.6
LYCIRBL	1.2 ± 0.8	1.9 ± 0.4	**1.3 ± 0.5**	1.6 ± 0.6	5.7	**2.2 ± 0.2**	0.8 ± 0.3	1.9 ± 0.9	1.4 ± 0.6

Table 5.2: Mean Per Joint Position Error (MPJPE) and standard deviation between participants per scene and preprocessing; in units of hip width; lower is better

As the Percentage of Correct Keypoints (PCK) metric is derived from the same data as the MPJPE, the results are comparable. The metric is calculated per scene and preprocessing method, with an average computed over all participants. Table 5.3, displays the PCK values given a threshold of 0.5. The outcomes demonstrate a resemblance to the findings of the PIR and MPJPE metrics. Again, the accuracy of pose estimation reduces with colorization methods, though minor improvements are evident in some scenes. However, for all cases where the PCK is improved by colorization, the improvements lie within the uncertainty of the PCK of the grayscale data.

	Dips 0°	Dips 45°	Dips 90°	Jumping jack	Push ups	Sit ups	Squat 0°	Squat 45°	Squat 90°
LLGS	49 ± 2	73 ± 8	**34 ± 9**	72 ± 9	8 ± 4	19 ± 6	71 ± 17	43 ± 26	49 ± 20
LLHSV	43	35 ± 39	27 ± 13	61 ± 11	5 ± 6	6 ± 7	60 ± 25	28 ± 24	47 ± 17
LLIRBL	44 ± 3	67 ± 11	31 ± 12	66 ± 11	**11 ± 4**	10 ± 10	69 ± 21	26 ± 26	46 ± 23
LCGS	**68 ± 4**	**77 ± 11**	28 ± 6	**72 ± 8**	10 ± 3	**20 ± 6**	**73 ± 17**	**50 ± 19**	**49 ± 20**
LCHSV	38 ± 9	10 ± 16	20 ± 9	65 ± 10	0	0	56 ± 20	16 ± 18	34 ± 17
LCIRBL	40 ± 8	21 ± 29	26 ± 14	56 ± 11	0	0	58 ± 27	21 ± 19	38 ± 14
LYLGS	52 ± 5	65 ± 17	33 ± 15	62 ± 14	7 ± 4	17 ± 6	59 ± 16	30 ± 25	46 ± 21
LYLHSV	36 ± 5	30 ± 28	22 ± 9	38 ± 27	2 ± 4	0	35 ± 24	26 ± 30	38 ± 21
LYLIRBL	24 ± 31	14 ± 23	27 ± 17	32 ± 17	0	5 ± 6	45 ± 21	22 ± 27	47 ± 20
LYCGS	55 ± 7	66 ± 15	34 ± 8	66 ± 11	8 ± 4	17 ± 7	63 ± 17	31 ± 24	49 ± 23
LYCHSV	30 ± 20	19 ± 30	20 ± 17	40 ± 28	1 ± 4	0	37 ± 26	23 ± 25	39 ± 20
LYCIRBL	25 ± 29	11 ± 13	32 ± 14	37 ± 10	0	4 ± 5	47 ± 21	21 ± 25	44 ± 16

Table 5.3: Mean Percentage of Correct Keypoints (PCK) and standard deviation between participants per scene and preprocessing; threshold of 0.5, as described in section 4.4.4; higher is better

Furthermore, the PCK given a variable threshold is demonstrated in Figure 5.8. It is evident from this graph that, on average, the performance of the two colorization methods is significantly worse than that of the grayscale methods when disregarding the uncertainties. Additionally, the LCGS technique yields the most satisfactory results regardless of what is deemed acceptable.

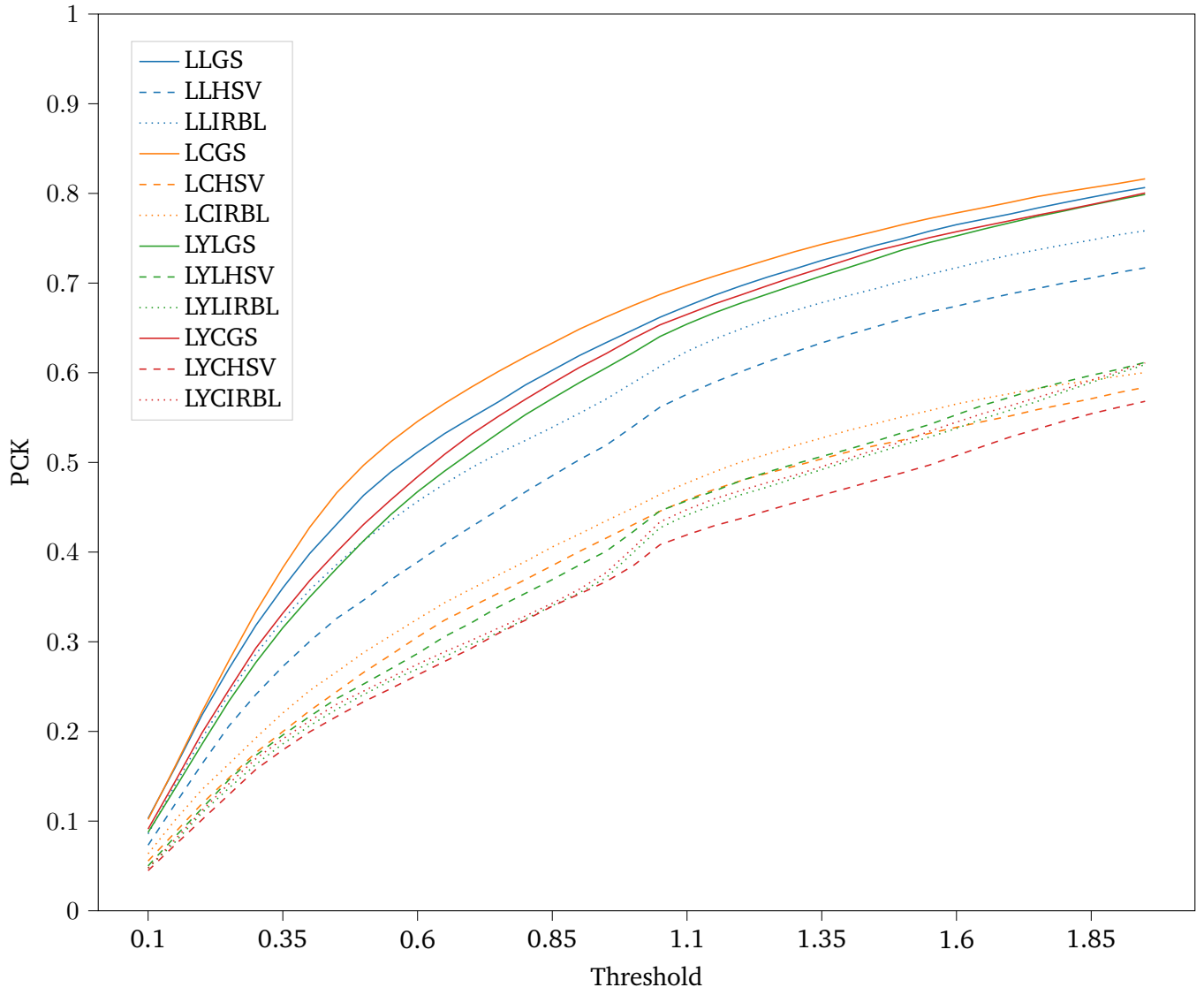


Figure 5.8: Mean PCK across all settings and participants, using a variable threshold for accepting keypoints as correct. All preprocessings with a grayscale representation are shown in a solid line, all those with the HSV color spiral in a dashed line, and all those with the IronBlack color map in a dotted line. The colors correspond to the combination of the other two preprocessing steps. The lossless linear representation is shown in blue, the lossless representation with the CLAHE algorithm is shown in orange, and the corresponding lossy representations are shown in green and red respectively. Frames without detection are not considered in this calculation.

The Average Relative Bone Length Over Time (ARBLOT) results for the different preprocessing methods are shown in Figure 5.9. The ARBLOT is calculated for each scene and each participant individually, and then averaged over all scenes and participants. Similar to the ARBLOT calculated in section 5.2, the results show a mean close to the expected value of 1, making these estimators mean-unbiased. Thus the preprocessing techniques are comparable by solely their variance. However, the variances only differ slightly, suggesting that the preprocessing techniques do not introduce significant errors in the skeleton structure.

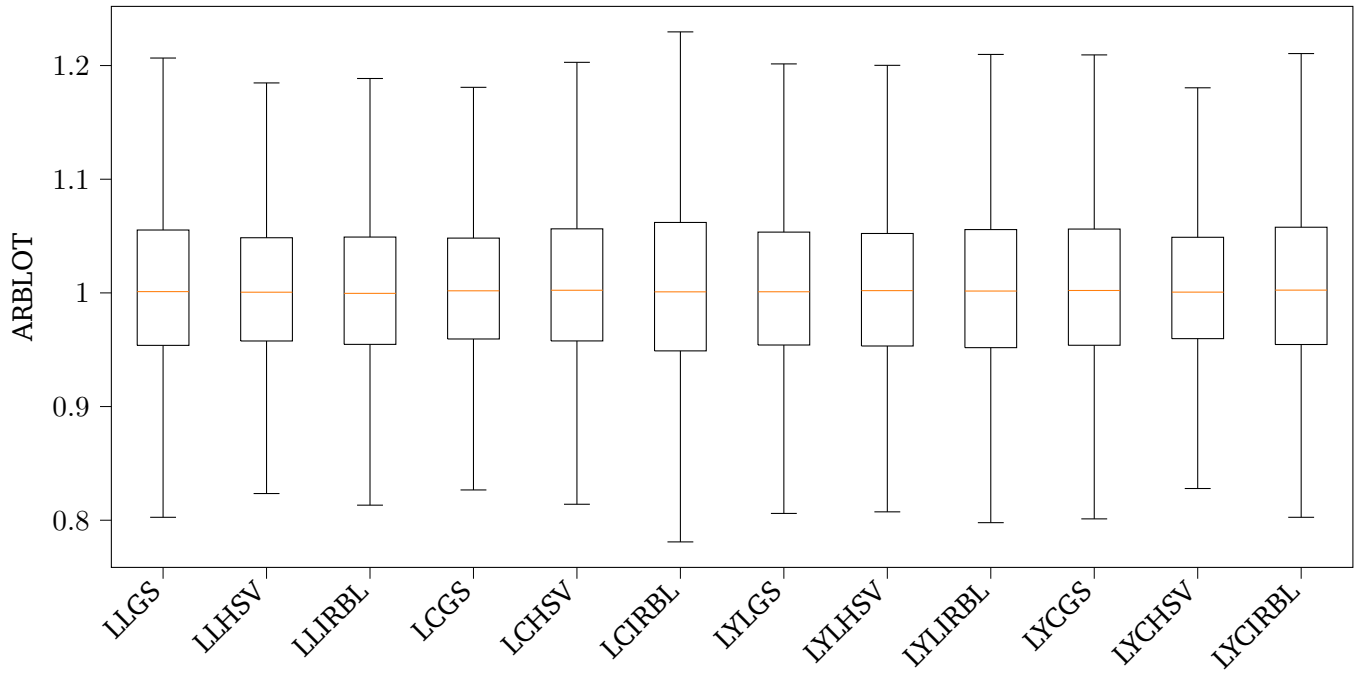


Figure 5.9: Boxplot for the ARBLOT of the different preprocessing approaches. The expected ratio is 1.

5.3.1 Mirror Error Corrected

The mirror error is calculated the same way as all other metrics, as the average of the mirror error for each scene and participant. As defined in section 4.4.7 a frame is considered mirrored if the MPJPE of the mirrored pose estimation is lower than the MPJPE of the original pose estimation. Table 5.4 shows the percentage of frames where a mirror error is detected. The mirror error does not necessarily indicate that the pose estimation, other than being mirrored, is correct. The highest mirror error over all preprocessings is detected in the *Dips* 90° and the *Push Ups* scenes. Both scenes show motion where the silhouette is not ambiguous. This suggests that these scenes may infer joint angles that are impossible for a human to achieve.

	Dips 0°	Dips 45°	Dips 90°	Jumping jack	Push ups	Sit ups	Squat 0°	Squat 45°	Squat 90°
LLGS	3 ± 4	3 ± 5	43 ± 29	4 ± 5	43 ± 30	15 ± 12	1 ± 1	11 ± 11	8 ± 12
LLHSV	3 ± 4	16 ± 27	35 ± 16	10 ± 9	28 ± 37	40 ± 25	15 ± 21	30 ± 26	6 ± 4
LLIRBL	0	0	56 ± 9	7 ± 7	26 ± 26	36 ± 27	7 ± 9	22 ± 23	11 ± 12
LCGS	0	1 ± 3	**68 ± 10**	2 ± 2	37 ± 30	9 ± 13	4 ± 8	17 ± 16	8 ± 19
LCHSV	14 ± 20	31 ± 42	62 ± 30	6 ± 4	65		17 ± 16	**57 ± 33**	27 ± 18
LCIRBL	4 ± 5	9 ± 12	59 ± 8	11 ± 13			9 ± 12	22 ± 20	18 ± 14
LYLGS	6 ± 8	12 ± 24	53 ± 3	13 ± 13	32 ± 18	19 ± 21	17 ± 11	21 ± 24	8 ± 11
LYLHSV	11 ± 5	11 ± 19	4 ± 6	42 ± 37	2 ± 1	73 ± 12	50 ± 25	47 ± 36	**37 ± 18**
LYLIRBL	**49 ± 70**	66 ± 39	44 ± 9	**44 ± 22**	72	53 ± 31	41 ± 17	39 ± 35	13 ± 13
LYCGS	3 ± 4	12 ± 23	60 ± 4	8 ± 7	33 ± 16	16 ± 19	16 ± 11	18 ± 20	8 ± 14
LYCHSV	32 ± 45	36 ± 51	13 ± 6	39 ± 37	1		**50 ± 24**	45 ± 32	32 ± 11
LYCIRBL	44 ± 62	**73 ± 23**	56 ± 25	36 ± 7	**84**	**58 ± 19**	34 ± 19	37 ± 34	15 ± 14

Table 5.4: Mirror error in percentage of frames where the flipped pose estimation has a lower MPJPE compared to the original estimation.

Since the mirror error can only be accounted for in ground truth-based metrics, only MPJPE and PCK metrics can be adjusted and compared. The corrected MPJPE is displayed in table 5.5. It suggests that correcting the mirror error improves the accuracy of pose estimation in all scenarios, albeit only marginally in most cases. However, this is expected, as the mirror error correction aims to simply lower the MPJPE per frame, thus lowering the average MPJPE.

	Dips +0°	Dips +45°	Dips +90°	Jumping jack	Push ups	Sit ups	Squat +0°	Squat +45°	Squat +90°
LLGS	+0.0 ± 0.0	+0.0 ± 0.3	−0.1 ± 0.6	+0.0 ± 0.2	−0.1 ± 2.1	+0.0 ± 0.6	+0.0 ± 0.3	+0.0 ± 0.7	+0.0 ± 3.2
LLHSV	+0.0 ± 0.0	+0.0 ± 1.9	−0.1 ± 0.1	−0.1 ± 0.3	−0.1 ± 2.5	−0.1 ± 0.3	−0.1 ± 0.4	+0.0 ± 1.0	+0.0 ± 0.4
LLIRBL	+0.0 ± 0.1	+0.0 ± 0.7	−0.1 ± 0.8	+0.0 ± 0.3	−0.1 ± 3.9	−0.1 ± 1.1	+0.0 ± 1.6	+0.0 ± 1.4	+0.0 ± 5.0
LCGS	+0.0 ± 0.1	+0.0 ± 0.2	−0.2 ± 0.7	+0.0 ± 0.1	−0.1 ± 1.8	+0.0 ± 0.4	+0.0 ± 0.6	+0.0 ± 1.1	−0.1 ± 1.3
LCHSV	−0.1 ± 0.2	−0.1 ± 0.2	**−0.2 ± 0.9**	+0.0 ± 0.3	+0.0 ± 0.0		−0.1 ± 1.0	**−0.1 ± 0.3**	−0.1 ± 0.8
LCIRBL	+0.0 ± 0.2	+0.0 ± 0.9	−0.1 ± 0.4	−0.1 ± 0.7			+0.0 ± 1.8	+0.0 ± 1.3	+0.0 ± 0.7
LYLGS	+0.0 ± 0.1	+0.0 ± 0.7	−0.1 ± 0.9	−0.1 ± 0.4	**−0.1 ± 1.9**	−0.1 ± 0.5	−0.1 ± 0.5	+0.0 ± 1.6	+0.0 ± 3.4
LYLHSV	−0.1 ± 0.1	−0.1 ± 0.6	+0.0 ± 1.4	−0.3 ± 0.9	+0.0 ± 3.8	−0.2 ± 0.0	**−0.2 ± 0.9**	−0.1 ± 1.5	**−0.1 ± 1.2**
LYLIRBL	**−0.2 ± 1.0**	**−0.2 ± 0.8**	−0.1 ± 1.1	**−0.3 ± 1.0**	+0.0 ± 0.0	−0.1 ± 0.5	−0.2 ± 0.6	−0.1 ± 1.4	+0.0 ± 1.6
LYCGS	+0.0 ± 0.2	+0.0 ± 0.6	−0.1 ± 0.5	+0.0 ± 0.3	−0.1 ± 2.4	+0.0 ± 0.5	−0.1 ± 0.6	+0.0 ± 1.7	+0.0 ± 3.1
LYCHSV	−0.2 ± 0.5	−0.2 ± 1.0	+0.0 ± 1.3	−0.3 ± 1.0	+0.0 ± 0.0		−0.2 ± 0.9	−0.1 ± 1.3	−0.1 ± 2.3
LYCIRBL	−0.2 ± 0.9	−0.2 ± 0.5	−0.1 ± 0.7	−0.2 ± 0.9	−0.1 ± 0.0	**−0.2 ± 0.4**	−0.2 ± 0.4	−0.1 ± 1.3	+0.0 ± 0.9

Table 5.5: Difference in the Mean Per Joint Position Error (MPJPE) and standard deviation corrected for the mirror error introduced in section 4.4.7; in units of hip width; lower is better

Table 5.6 depicts the corrected PCK given a threshold of 0.5. In contrast to the corrected MPJPE results, the corrected PCK is not always rectified by correcting the mirror error. Especially in the *Dips* 90° and *Push Ups* scenes, where the rate of mirror errors is exceptionally high, the PCK decreases after correcting the mirror error. This is likely caused by keypoint estimations that are relatively far from the actual keypoint location, but still closer than the mirrored keypoint estimation that they skew the MPJPE in favor of the mirrored keypoint estimation. However, the PCK only considers whether the keypoint is within a certain threshold of the actual keypoint location, which does not take outliers into account. Thus, the PCK is not skewed by outliers and is not rectified by correcting the mirror error.

	Dips +0°	Dips +45°	Dips +90°	Jumping jack	Push ups	Sit ups	Squat +0°	Squat +45°	Squat +90°
LLGS	+1 ± 3	+0 ± 12	−4 ± 14	+1 ± 13	+0 ± 5	+0 ± 9	+0 ± 24	−1 ± 37	+0 ± 28
LLHSV	+1 ± 1	+1 ± 54	−5 ± 18	+2 ± 15	+0 ± 8	+3 ± 9	+5 ± 33	+1 ± 33	−1 ± 24
LLIRBL	+0 ± 4	+0 ± 15	−6 ± 17	+1 ± 16	−1 ± 6	+2 ± 13	+0 ± 29	+1 ± 37	+0 ± 32
LCGS	+0 ± 5	+0 ± 16	−8 ± 7	+0 ± 11	−1 ± 4	+0 ± 8	+0 ± 24	+0 ± 27	**+1 ± 27**
LCHSV	+3 ± 11	+1 ± 22	−7 ± 10	+1 ± 13	+0 ± 0	+0 ± 0	+4 ± 28	**+3 ± 24**	−1 ± 25
LCIRBL	+1 ± 10	+0 ± 42	−6 ± 18	+3 ± 14	+0 ± 0	+0 ± 0	+2 ± 37	+1 ± 27	−1 ± 20
LYLGS	+0 ± 7	+1 ± 23	−7 ± 20	+3 ± 19	+0 ± 5	+1 ± 7	+4 ± 22	+1 ± 35	+0 ± 29
LYLHSV	+1 ± 7	+2 ± 39	**+0 ± 13**	+11 ± 32	+0 ± 6	+2 ± 3	+11 ± 31	+1 ± 41	−2 ± 29
LYLIRBL	+7 ± 37	**+5 ± 31**	−6 ± 22	**+11 ± 22**	+0 ± 1	**+5 ± 8**	+11 ± 27	+2 ± 37	−1 ± 27
LYCGS	+0 ± 10	+1 ± 20	−7 ± 11	+1 ± 16	+0 ± 6	+1 ± 9	+2 ± 24	+0 ± 34	+0 ± 32
LYCHSV	+5 ± 23	+3 ± 43	−2 ± 22	+11 ± 33	+0 ± 5	+0 ± 0	**+11 ± 34**	+2 ± 34	−2 ± 28
LYCIRBL	**+7 ± 35**	+4 ± 19	−6 ± 21	+9 ± 15	**+0 ± 1**	+3 ± 8	+9 ± 28	+2 ± 34	−1 ± 22

Table 5.6: Difference in the mean Percentage of Correct Keypoints (PCK) and standard deviation corrected for the mirror error introduced in section 4.4.7; threshold of 0.5, as described in section 4.4.4; higher is better

Figure 5.10 presents the average corrected PCK per preprocessing method with a variable threshold. The results suggest that correcting the mirror error mainly affects preprocessings that perform poorly regardless.

The top-performing preprocessings, which are based on grayscale representation, do not significantly benefit from correcting the mirror error but, in some cases, even perform worse.

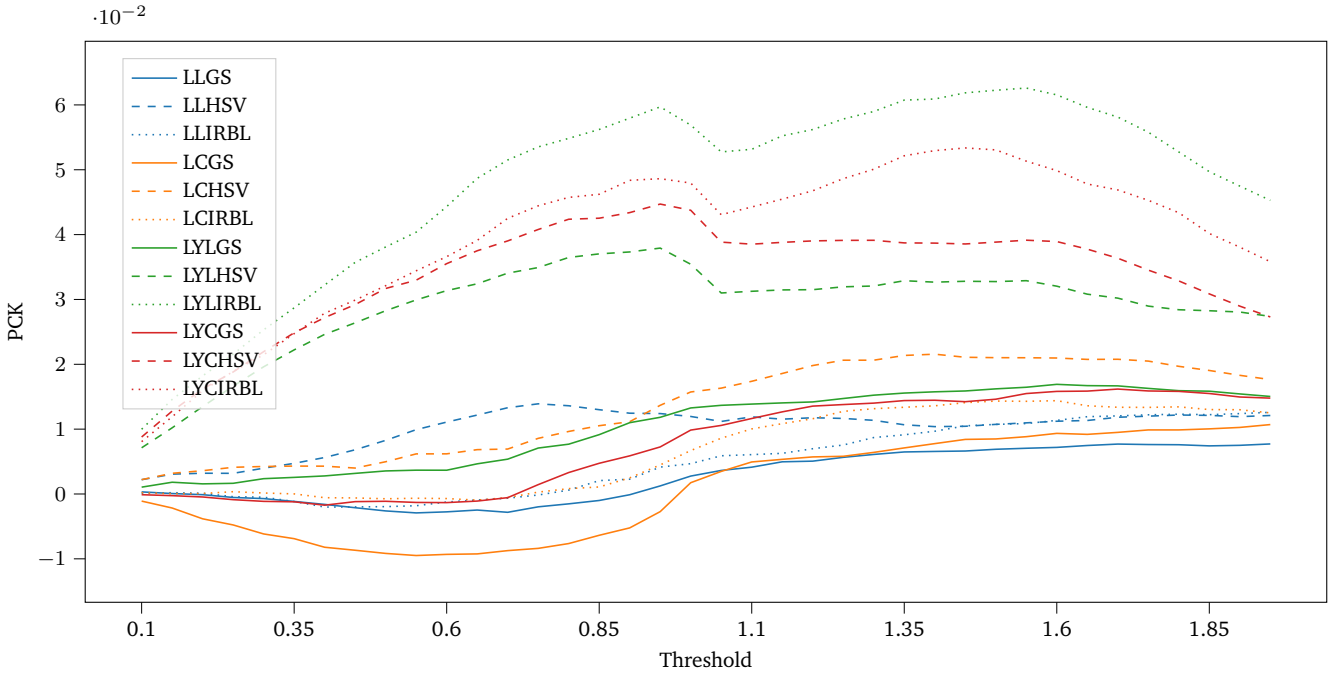


Figure 5.10: Difference of the mean PCK across all settings and participants corrected for the mirror error, using a variable threshold for accepting keypoints as correct. The colors and line styles correspond to the specific preprocessing in the same way as in Figure 5.8. Frames without detection are not considered in this calculation.

6 Discussion

This chapter discusses the results presented in chapter 5 and provides an outlook on interesting avenues for future work.

The results of the ARBLOT metric show that no matter the preprocessing technique, MotionBERT always has a relatively high temporal stability in the skeleton. The PIR suggests that given a suitable preprocessing, the M3DHPE pipeline can at least find a person in close to 100 % of all frames. However, the PIR does not provide any information on the quality of the pose estimation. The MPJPE results suggest that the M3DHPE pipeline is able to estimate the pose of a person to a reasonable degree of accuracy, especially in scenes with minimal self-occlusion. The results of the PCK metric confirm the high dependence of the quality of the pose estimation on the complexity of the problem. High self-occlusion and inadequate preprocessing techniques can lead to a significant decrease in the quality of the pose estimation. The corrections applied to account for the suspected mirror error only marginally improve the results. This suggests that the mirror error is only a minor issue for the accuracy of the pose estimation.

Overall, the results of the entire M3DHPE pipeline demonstrated in chapter 5 for thermographic images show less success than their RGB counterparts, but they still exhibit promise. The findings demonstrate that current M3DHPE workflows have the capability to estimate a person's pose from thermographic images. However, their current state does not meet the necessary standards for medical use. Comparative preprocessing results suggest that implementation of preprocessing techniques on input data can enhance the performance of the M3DHPE pipeline if done correctly. As such, there is significant potential for further enhancing the performance of M3DHPE on thermographic images in future work.

6.1 Future Work

To ensure accurate HPE performance and generalizability of results, the ground truth generation methodology was limited to methods that do not require the human to wear any additional equipment. This was done due to concerns regarding the impact of active electronics or infrared reflecting materials in the image on HPE performance. As such, the ground truth generation methodology presented in section 4.4.1 is not ideal. Future work should investigate the impact of active electronics and infrared reflecting materials on HPE performance and the generalizability of the results. If the impact is negligible, the ground truth generation methodology can be improved by utilizing for example a motion capture suit to generate the ground truth. This would allow for a more accurate ground truth and thus a more accurate evaluation of the M3DHPE pipeline.

Some assumptions were made on the negligibility of the impact of perspective distortion on the M3DHPE pipeline in section 4.1.1. Further research is necessary to investigate the validity of these assumptions. If the impact is significant, future work could investigate the usage of a beam splitter rig to simultaneously capture the RGB and thermographic images from the same perspective to reduce the impact of perspective distortion, similar to the work proposed by Zhang *et al.* [86].

Another aspect not yet investigated is the performance of thermographic M3DHPE on multiple people in the same image. Future research should investigate whether SOTA approaches to multi-person HPE perform similarly on thermographic images as they do on RGB images. If not, further investigation should be conducted to determine the reasons for this discrepancy and propose a solution.

The results presented here are limited to less challenging situations. The entire dataset consists of only one person in the image; the person is not occluded, and any movement is at a moderate pace. The recordings were also done in a controlled environment with consistent lighting conditions, a static background and camera, and a strong separation between the person and the background. As such, the results presented here do not represent the performance of M3DHPE on thermographic images in real-world scenarios. Further research is needed to evaluate the performance of M3DHPE on thermographic images in more challenging situations. These include but are not limited to occlusions, low light, and fast movements. The impact of low light on M3DHPE performance is exciting, as thermographic images are not affected by low light in the same way as RGB images.

With MotionBERT working by predicting movement patterns it is interesting to investigate the impact of small unexpected movements, like those exhibited by patients with Parkinson's disease, on the performance of the M3DHPE pipeline. Furthermore, another interesting avenue for future work is to compare this impact of unexpected movements to the performance of different M3DHPE pipelines that infer positions without movement patterns.

As MotionBERT uses AlphaPose for the initial 2D pose estimation in their reference implementation, it is interesting to investigate the impact of using a different 2D pose estimation model. Section 5.1 shows that the AlphaPose framework does not perform well on thermographic images.

Lastly, there are two main avenues for major improvements to M3DHPE pipeline on thermographic images. The first is the improvement of the preprocessing pipeline. The results presented in section 5.3 show that preprocessing can have a significant impact on the performance of the M3DHPE pipeline. As such, future work should investigate the impact of further preprocessing techniques, as well as the possibility of reconstructing RGB images from thermographic images to allow for the usage of pre-trained HPE models. The second avenue for major improvements is to transfer the M3DHPE pipeline to the target domain. This should not only be done to models that perform relatively well on thermographic images, but also to ones using a different backbone architecture. This would allow for the usage of M3DHPE pipelines on thermographic images without the need for preprocessing. Furthermore, it would allow for the usage of M3DHPE pipelines on thermographic images in real-time applications, as the inference time of the M3DHPE pipeline would be significantly reduced.

A Bibliography

[1] T. Casy, A. Tronchot, H. Thomazeau, X. Morandi, P. Jannin, and A. Huaulmé, ""stand-up straight!": Human pose estimation to evaluate postural skills during orthopedic surgery simulations", *International Journal of Computer Assisted Radiology and Surgery*, vol. 18, no. 2, pp. 279–288, Feb. 1, 2023, ISSN: 1861-6429. DOI: 10.1007/s11548-022-02762-5. [Online]. Available: https://doi.org/10.1007/s11548-022-02762-5 (visited on 11/01/2023).

[2] A. K. Patil, A. Balasubramanyam, J. Y. Ryu, B. Chakravarthi, and Y. H. Chai, "An open-source platform for human pose estimation and tracking using a heterogeneous multi-sensor system", *Sensors*, vol. 21, no. 7, p. 2340, Jan. 2021, ISSN: 1424-8220. DOI: 10.3390/s21072340. [Online]. Available: https://www.mdpi.com/1424-8220/21/7/2340 (visited on 09/19/2023).

[3] L. Schmidtke, A. Vlontzos, S. Ellershaw, A. Lukens, T. Arichi, and B. Kainz, *Unsupervised human pose estimation through transforming shape templates*, May 10, 2021. arXiv: 2105.04154[cs]. [Online]. Available: http://arxiv.org/abs/2105.04154 (visited on 09/19/2023).

[4] R. Rabbito, "Using deep learning-based pose estimation algorithms for markerless gait analysis in rehabilitation medicine", laurea, Politecnico di Torino, Mar. 30, 2021, 83 pp. [Online]. Available: https://webthesis.biblio.polito.it/17536/ (visited on 09/19/2023).

[5] D. Groos, H. Ramampiaro, and E. A. Ihlen, "EfficientPose: Scalable single-person pose estimation", *Applied Intelligence*, vol. 51, no. 4, pp. 2518–2533, Apr. 1, 2021, ISSN: 1573-7497. DOI: 10.1007/s10489-020-01918-7. [Online]. Available: https://doi.org/10.1007/s10489-020-01918-7 (visited on 09/19/2023).

[6] S. Liu, N. Sehgal, and S. Ostadabbas, "Adapted human pose: Monocular 3d human pose estimation with zero real 3d pose data", *Applied Intelligence*, vol. 52, no. 12, pp. 14 491–14 506, Sep. 1, 2022, ISSN: 1573-7497. DOI: 10.1007/s10489-022-03341-6. [Online]. Available: https://doi.org/10.1007/s10489-022-03341-6 (visited on 09/20/2023).

[7] W. Liu, Q. Bao, Y. Sun, and T. Mei, *Recent advances in monocular 2d and 3d human pose estimation: A deep learning perspective*, Apr. 23, 2021. DOI: 10.48550/arXiv.2104.11536. arXiv: 2104.11536[cs]. [Online]. Available: http://arxiv.org/abs/2104.11536 (visited on 09/20/2023).

[8] D. Mehta *et al.*, *Monocular 3d human pose estimation in the wild using improved CNN supervision*, Oct. 4, 2017. DOI: 10.48550/arXiv.1611.09813. arXiv: 1611.09813[cs]. [Online]. Available: http://arxiv.org/abs/1611.09813 (visited on 09/20/2023).

[9] D. Kesztyüs, S. Brucher, and T. Kesztyüs, "Use of infrared thermography in medical diagnostics: A scoping review protocol", *BMJ Open*, vol. 12, no. 4, e059833, Apr. 2022, ISSN: 2044-6055, 2044-6055. DOI: 10.1136/bmjopen-2021-059833. [Online]. Available: https://bmjopen.bmj.com/lookup/doi/10.1136/bmjopen-2021-059833 (visited on 09/18/2023).

[10] D. Staff. "Breakthrough benefits of thermography in the medical industry", KnowHow. (Jun. 2, 2021), [Online]. Available: https://knowhow.distrelec.com/medical-healthcare/breakthrough-benefits-of-thermography-in-the-medical-industry/ (visited on 09/18/2023).

[11] A. A. Khan and A. S. Arora, "Thermography as an economical alternative modality to mammography for early detection of breast cancer", *Journal of Healthcare Engineering*, vol. 2021, e5543101, Jul. 31, 2021, ISSN: 2040-2295. DOI: 10.1155/2021/5543101. [Online]. Available: https://www.hindawi.com/journals/jhe/2021/5543101/ (visited on 11/07/2023).

[12] B. Braga, G. Queirós, C. Abreu, and S. I. Lopes, "Assessment of low-cost infrared thermography systems for medical screening in nursing homes", in *2021 17th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Oct. 2021, pp. 157–162. DOI: 10.1109/WiMob52687.2021.9606256. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9606256 (visited on 11/07/2023).

[13] B. Polat *et al.*, "Sensitivity and specificity of infrared thermography in detection of subclinical mastitis in dairy cows", *Journal of Dairy Science*, vol. 93, no. 8, pp. 3525–3532, Aug. 2010, ISSN: 00220302. DOI: 10.3168/jds.2009-2807. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S002203021000370X (visited on 09/18/2023).

[14] E.-K. Ng, S. Fok, Y. Peh, F. Ng, and L. Sim, "Computerized detection of breast cancer with artificial intelligence and thermograms", *Journal of Medical Engineering & Technology*, vol. 26, no. 4, pp. 152–157, Jan. 2002, ISSN: 0309-1902, 1464-522X. DOI: 10.1080/03091900210146941. [Online]. Available: http://www.tandfonline.com/doi/full/10.1080/03091900210146941 (visited on 09/18/2023).

[15] E. Y. K. Ng and E. C. Kee, "Advanced integrated technique in breast cancer thermography", *Journal of Medical Engineering & Technology*, vol. 32, no. 2, pp. 103–114, Jan. 2008, ISSN: 0309-1902, 1464-522X. DOI: 10.1080/03091900600562040. [Online]. Available: http://www.tandfonline.com/doi/full/10.1080/03091900600562040 (visited on 09/18/2023).

[16] E. Y. K. Ng, L. N. Ung, F. C. Ng, and L. S. J. Sim, "Statistical analysis of healthy and malignant breast thermography", *Journal of Medical Engineering & Technology*, vol. 25, no. 6, pp. 253–263, Jan. 2001, ISSN: 0309-1902, 1464-522X. DOI: 10.1080/03091900110086642. [Online]. Available: http://www.tandfonline.com/doi/full/10.1080/03091900110086642 (visited on 09/18/2023).

[17] M. Prasał, K. M. Sawicka, and A. Wysokiński, "[thermography in cardiology]", *Kardiologia Polska*, vol. 68, no. 9, pp. 1052–1056, Sep. 2010, ISSN: 0022-9032.

[18] M. Brenner, "Thermal signature analysis as a novel method for evaluating inflammatory arthritis activity", *Annals of the Rheumatic Diseases*, vol. 65, no. 3, pp. 306–311, Mar. 1, 2006, ISSN: 0003-4967. DOI: 10.1136/ard.2004.035246. [Online]. Available: https://ard.bmj.com/lookup/doi/10.1136/ard.2004.035246 (visited on 09/18/2023).

[19] A. E. Denoble, N. Hall, C. F. Pieper, and V. B. Kraus, "Patellar skin surface temperature by thermography reflects knee osteoarthritis severity", *Clinical Medicine Insights: Arthritis and Musculoskeletal Disorders*, vol. 3, CMAMD.S5916, Jan. 2010, ISSN: 1179-5441, 1179-5441. DOI: 10.4137/CMAMD.S5916. [Online]. Available: http://journals.sagepub.com/doi/10.4137/CMAMD.S5916 (visited on 09/18/2023).

[20] D. Rusch, M. Follmann, B. Boss, and G. Neeck, "Dynamic thermography of the knee joints in rheumatoid arthritis (RA) in the course of the first therapy of the patient with methylprednisolone", *Zeitschrift für Rheumatologie*, vol. 59, pp. 131–135, S2 Oct. 2000, ISSN: 0340-1855. DOI: 10.1007/s003930070009. [Online]. Available: http://link.springer.com/10.1007/s003930070009 (visited on 09/18/2023).

[21] F. Ring, "Thermal imaging today and its relevance to diabetes", *Journal of Diabetes Science and Technology*, vol. 4, no. 4, pp. 857–862, Jul. 2010, ISSN: 1932-2968, 1932-2968. DOI: 10.1177/193229681000400414. [Online]. Available: http://journals.sagepub.com/doi/10.1177/193229681000400414 (visited on 09/18/2023).

[22] N. Papanas, K. Papatheodorou, D. Papazoglou, S. Kotsiou, and E. Maltezos, "Association between foot temperature and sudomotor dysfunction in type 2 diabetes", *Journal of Diabetes Science and Technology*, vol. 4, no. 4, pp. 803–807, Jul. 2010, ISSN: 1932-2968, 1932-2968. DOI: 10.1177/193229681000400406. [Online]. Available: http://journals.sagepub.com/doi/10.1177/193229681000400406 (visited on 09/18/2023).

[23] G. Martini, "Juvenile-onset localized scleroderma activity detection by infrared thermography", *Rheumatology*, vol. 41, no. 10, pp. 1178–1182, Oct. 1, 2002, ISSN: 14602172. DOI: 10.1093/rheumatology/41.10.1178. [Online]. Available: https://academic.oup.com/rheumatology/article-lookup/doi/10.1093/rheumatology/41.10.1178 (visited on 09/18/2023).

[24] T. Moore, S. Vij, A. Murray, M. Bhushan, C. Griffiths, and A. Herrick, "Pilot study of dual-wavelength (532 and 633 nm) laser doppler imaging and infrared thermography of morphoea", *British Journal of Dermatology*, vol. 160, no. 4, pp. 864–867, Apr. 2009, ISSN: 00070963, 13652133. DOI: 10.1111/j.1365-2133.2008.08933.x. [Online]. Available: https://academic.oup.com/bjd/article/160/4/864/6641917 (visited on 09/18/2023).

[25] M. Kocic, M. Lazovic, I. Dimitrijevic, D. Mancic, and A. Stankovic, "Evaluation of low level laser and interferential current in the therapy of complex regional pain syndrome by infrared thermographic camera", *Vojnosanitetski pregled*, vol. 67, no. 9, pp. 755–760, 2010, ISSN: 0042-8450, 2406-0720. DOI: 10.2298/VSP1009755K. [Online]. Available: https://doiserbia.nb.rs/Article.aspx?ID=0042-84501009755K (visited on 09/18/2023).

[26] O. Schlager *et al.*, "Correlation of infrared thermography and skin perfusion in raynaud patients and in healthy controls", *Microvascular Research*, vol. 80, no. 1, pp. 54–57, Jul. 2010, ISSN: 00262862. DOI: 10.1016/j.mvr.2010.01.010. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0026286210000117 (visited on 09/18/2023).

[27] V. Bazarevsky, I. Grishchenko, K. Raveendran, T. Zhu, F. Zhang, and M. Grundmann, *BlazePose: On-device real-time body pose tracking*, Jun. 17, 2020. DOI: 10.48550/arXiv.2006.10204. arXiv: 2006.10204[cs]. [Online]. Available: http://arxiv.org/abs/2006.10204 (visited on 09/21/2023).

[28] C. Lugaresi *et al.*, *MediaPipe: A framework for building perception pipelines*, Jun. 14, 2019. DOI: 10.48550/arXiv.1906.08172. arXiv: 1906.08172[cs]. [Online]. Available: http://arxiv.org/abs/1906.08172 (visited on 09/21/2023).

[29] V. Bazarevsky, Y. Kartynnik, A. Vakunov, K. Raveendran, and M. Grundmann, *BlazeFace: Sub-millisecond neural face detection on mobile GPUs*, Jul. 14, 2019. DOI: 10.48550/arXiv.1907.05047. arXiv: 1907.05047[cs]. [Online]. Available: http://arxiv.org/abs/1907.05047 (visited on 12/05/2023).

[30] F. Zhang *et al.*, *MediaPipe hands: On-device real-time hand tracking*, Jun. 17, 2020. arXiv: 2006.10214[cs]. [Online]. Available: http://arxiv.org/abs/2006.10214 (visited on 12/05/2023).

[31] T.-Y. Lin *et al.*, *Microsoft COCO: Common objects in context*, Feb. 20, 2015. DOI: 10.48550/arXiv.1405.0312. arXiv: 1405.0312[cs]. [Online]. Available: http://arxiv.org/abs/1405.0312 (visited on 12/05/2023).

[32] W. Liu *et al.*, "SSD: Single shot MultiBox detector", in vol. 9905, 2016, pp. 21–37. DOI: 10.1007/978-3-319-46448-0_2. arXiv: 1512.02325[cs]. [Online]. Available: http://arxiv.org/abs/1512.02325 (visited on 12/05/2023).

[33] A. Newell, K. Yang, and J. Deng, *Stacked hourglass networks for human pose estimation*, Jul. 26, 2016. DOI: 10.48550/arXiv.1603.06937. arXiv: 1603.06937[cs]. [Online]. Available: http://arxiv.org/abs/1603.06937 (visited on 12/05/2023).

[34] H.-S. Fang *et al.*, *AlphaPose: Whole-body regional multi-person pose estimation and tracking in real-time*, Nov. 7, 2022. DOI: 10.48550/arXiv.2211.03375. arXiv: 2211.03375[cs]. [Online]. Available: http://arxiv.org/abs/2211.03375 (visited on 09/21/2023).

[35] J. Li, C. Xu, Z. Chen, S. Bian, L. Yang, and C. Lu, *HybrIK: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation*, Apr. 27, 2022. DOI: 10.48550/arXiv.2011.14672. arXiv: 2011.14672[cs]. [Online]. Available: http://arxiv.org/abs/2011.14672 (visited on 10/17/2023).

[36] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, *YOLOX: Exceeding YOLO series in 2021*, Aug. 5, 2021. DOI: 10.48550/arXiv.2107.08430. arXiv: 2107.08430[cs]. [Online]. Available: http://arxiv.org/abs/2107.08430 (visited on 09/21/2023).

[37] Y. Xiu, J. Li, H. Wang, Y. Fang, and C. Lu, *Pose flow: Efficient online pose tracking*, Jul. 2, 2018. DOI: 10.48550/arXiv.1802.00977. arXiv: 1802.00977[cs]. [Online]. Available: http://arxiv.org/abs/1802.00977 (visited on 12/05/2023).

[38] J. Redmon and A. Farhadi, *YOLOv3: An incremental improvement*, Apr. 8, 2018. arXiv: 1804.02767[cs]. [Online]. Available: http://arxiv.org/abs/1804.02767 (visited on 12/06/2023).

[39] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, Dec. 10, 2015. DOI: 10.48550/arXiv.1512.03385. arXiv: 1512.03385[cs]. [Online]. Available: http://arxiv.org/abs/1512.03385 (visited on 12/06/2023).

[40] T. Von Marcard, R. Henschel, M. J. Black, B. Rosenhahn, and G. Pons-Moll, "Recovering accurate 3d human pose in the wild using IMUs and a moving camera", in *Computer Vision – ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., vol. 11214, Cham: Springer International Publishing, 2018, pp. 614–631, ISBN: 978-3-030-01248-9 978-3-030-01249-6. DOI: 10.1007/978-3-030-01249-6_37. [Online]. Available: https://link.springer.com/10.1007/978-3-030-01249-6_37 (visited on 12/06/2023).

[41] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1325–1339, Jul. 2014, ISSN: 1939-3539. DOI: 10.1109/TPAMI.2013.248. [Online]. Available: https://ieeexplore.ieee.org/document/6682899 (visited on 12/06/2023).

[42] W. Zhu, X. Ma, Z. Liu, L. Liu, W. Wu, and Y. Wang, *MotionBERT: A unified perspective on learning human motion representations*, Aug. 14, 2023. arXiv: 2210.06551[cs]. [Online]. Available: http://arxiv.org/abs/2210.06551 (visited on 09/21/2023).

[43] J. Dai *et al.*, *Deformable convolutional networks*, Jun. 5, 2017. arXiv: 1703.06211[cs]. [Online]. Available: http://arxiv.org/abs/1703.06211 (visited on 12/07/2023).

[44] W. Shi *et al.*, *Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network*, Sep. 23, 2016. arXiv: 1609.05158[cs,stat]. [Online]. Available: http://arxiv.org/abs/1609.05158 (visited on 12/07/2023).

[45] A. Vaswani *et al.*, *Attention is all you need*, Aug. 1, 2023. DOI: 10.48550/arXiv.1706.03762. arXiv: 1706.03762[cs]. [Online]. Available: http://arxiv.org/abs/1706.03762 (visited on 12/07/2023).

[46] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black. "AMASS: Archive of motion capture as surface shapes", arXiv.org. (Apr. 5, 2019), [Online]. Available: https://arxiv.org/abs/1904.03278v1 (visited on 12/07/2023).

[47] M. Planck, "Ueber das gesetz der energieverteilung im normalspectrum", *Annalen der Physik*, vol. 309, no. 3, pp. 553–563, 1901, ISSN: 1521-3889. DOI: 10.1002/andp.19013090310. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/andp.19013090310 (visited on 11/15/2023).

[48] A. Akula, R. Ghosh, and H. K. Sardana, "Thermal imaging and its application in defence systems", *AIP Conference Proceedings*, vol. 1391, no. 1, pp. 333–335, Oct. 20, 2011, ISSN: 0094-243X. DOI: 10.1063/1.3643540. [Online]. Available: https://doi.org/10.1063/1.3643540 (visited on 12/06/2023).

[49] S. M *et al.*, "An effective drone surveillance system using thermal imaging", in *2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)*, Oct. 2020, pp. 477–482. DOI: 10.1109/ICSTCEE49637.2020.9277292. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9277292 (visited on 12/06/2023).

[50] R. Jimenez *et al.*, "Fabrication of microbolometer arrays based on polymorphous silicon–germanium", *Sensors*, vol. 20, no. 9, p. 2716, Jan. 2020, ISSN: 1424-8220. DOI: 10.3390/s20092716. [Online]. Available: https://www.mdpi.com/1424-8220/20/9/2716 (visited on 12/06/2023).

[51] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift", in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No.PR00662)*, vol. 2, Jun. 2000, 142–149 vol.2. DOI: 10.1109/CVPR.2000.854761. [Online]. Available: https://ieeexplore.ieee.org/document/854761 (visited on 10/04/2023).

[52] H. Nanda and L. Davis, "Probabilistic template based pedestrian detection in infrared videos", in *Intelligent Vehicle Symposium, 2002. IEEE*, vol. 1, Versailles, France: IEEE, 2003, pp. 15–20, ISBN: 978-0-7803-7346-4. DOI: 10.1109/IVS.2002.1187921. [Online]. Available: http://ieeexplore.ieee.org/document/1187921/ (visited on 10/04/2023).

[53] A. Fernández-Caballero, M. T. López, and J. Serrano-Cuerda, "Thermal-infrared pedestrian ROI extraction through thermal and motion information fusion", *Sensors*, vol. 14, no. 4, pp. 6666–6676, Apr. 2014, ISSN: 1424-8220. DOI: 10.3390/s140406666. [Online]. Available: https://www.mdpi.com/1424-8220/14/4/6666 (visited on 10/04/2023).

[54] Y. Zheng, F. Zhou, L. Li, X. Bai, and C. Sun, "Mutual guidance-based saliency propagation for infrared pedestrian images", *IEEE Access*, vol. 7, pp. 113 355–113 371, 2019, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2019.2933310. [Online]. Available: https://ieeexplore.ieee.org/document/8788576 (visited on 10/04/2023).

[55] S. K. Biswas and P. Milanfar, "Linear support tensor machine with LSK channels: Pedestrian detection in thermal infrared images", *IEEE transactions on image processing: a publication of the IEEE Signal Processing Society*, vol. 26, no. 9, pp. 4229–4242, Sep. 2017, ISSN: 1941-0042. DOI: 10.1109/TIP.2017.2705426.

[56] A. Akula, A. K. Shah, and R. Ghosh, "Deep learning approach for human action recognition in infrared images", *Cognitive Systems Research*, vol. 50, pp. 146–154, Aug. 1, 2018, ISSN: 1389-0417. DOI: 10.1016/j.cogsys.2018.04.002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389041717302206 (visited on 10/17/2023).

[57] Y. Guo, Y. Chen, J. Deng, S. Li, and H. Zhou, "Identity-preserved human posture detection in infrared thermal images: A benchmark", *Sensors*, vol. 23, no. 1, p. 92, Jan. 2023, ISSN: 1424-8220. DOI: 10.3390/s23010092. [Online]. Available: https://www.mdpi.com/1424-8220/23/1/92 (visited on 10/01/2023).

[58] Q. Chen, Y. Wang, T. Yang, X. Zhang, J. Cheng, and J. Sun, *You only look one-level feature*, Mar. 17, 2021. DOI: 10.48550/arXiv.2103.09460. arXiv: 2103.09460[cs]. [Online]. Available: http://arxiv.org/abs/2103.09460 (visited on 11/01/2023).

[59] C. Feng, Y. Zhong, Y. Gao, M. R. Scott, and W. Huang, *TOOD: Task-aligned one-stage object detection*, Aug. 28, 2021. DOI: 10.48550/arXiv.2108.07755. arXiv: 2108.07755[cs]. [Online]. Available: http://arxiv.org/abs/2108.07755 (visited on 11/01/2023).

[60] "PI 640i: World's smallest VGA infrared camera", Optris. (), [Online]. Available: https://www.optris.com/en-us/product/infrared-cameras/pi-series/pi-640i/ (visited on 09/19/2023).

[61] "Reolink RLC-410: 5MP Super HD Bullet PoE IP-Kamera für Außen & Innen | Offiziell". (), [Online]. Available: https://reolink.com/product/rlc-410/ (visited on 10/11/2023).

[62] "InfraTec Wärmebildkamera Serie VarioCAM HD head 900". (), [Online]. Available: https://www.infratec.de/thermografie/waermebildkameras/variocam-hd-head-900/ (visited on 10/11/2023).

[63] B. Hönlinger and H. Nasse, *Distortion*, Oct. 2009. [Online]. Available: https://lenspire.zeiss.com/photo/app/uploads/2022/02/technical-article-distortion.pdf (visited on 11/15/2023).

[64] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction", in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Juan, Puerto Rico: IEEE Comput. Soc, 1997, pp. 1106–1112, ISBN: 978-0-8186-7822-6. DOI: 10.1109/CVPR.1997.609468. [Online]. Available: http://ieeexplore.ieee.org/document/609468/ (visited on 10/01/2023).

[65] "OpenCV", OpenCV. (), [Online]. Available: https://opencv.org/ (visited on 10/16/2023).

[66] C. Harris and M. Stephens, "A combined corner and edge detector", in *Procedings of the Alvey Vision Conference 1988*, Manchester: Alvey Vision Club, 1988, pp. 23.1–23.6. DOI: 10.5244/C.2.23. [Online]. Available: http://www.bmva.org/bmvc/1988/avc-88-023.html (visited on 12/04/2023).

[67] N. Otsu, "A threshold selection method from gray-level histograms", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, Jan. 1979, ISSN: 2168-2909. DOI: 10.1109/TSMC.1979.4310076. [Online]. Available: https://ieeexplore.ieee.org/document/4310076 (visited on 12/04/2023).

[68] J. Shi and Tomasi, "Good features to track", in *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 1994, pp. 593–600. DOI: 10.1109/CVPR.1994.323794. [Online]. Available: https://ieeexplore.ieee.org/document/323794 (visited on 12/04/2023).

[69] D. Brown, "Decentering distortion of lenses", 1966. [Online]. Available: https://www.semanticscholar.org/paper/Decentering-distortion-of-lenses-Brown/2ef001c656378a1c5cf80488b35684742220d3f9 (visited on 12/05/2023).

[70] "BoofCV". (), [Online]. Available: http://boofcv.org/index.php?title=Main_Page (visited on 10/16/2023).

[71] P. Abeles, *Pyramidal blur aware x-corner chessboard detector*, Oct. 26, 2021. arXiv: 2110.13793[cs]. [Online]. Available: http://arxiv.org/abs/2110.13793 (visited on 10/16/2023).

[72] D. J. Ketcham, "Real-time image enhancement techniques", in *Image Processing*, vol. 0074, SPIE, Jul. 9, 1976, pp. 120–125. DOI: 10.1117/12.954708. [Online]. Available: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/0074/0000/Real-Time-Image-Enhancement-Techniques/10.1117/12.954708.full (visited on 11/16/2023).

[73] K. Zuiderveld, "Contrast limited adaptive histogram equalization", in *Graphics Gems*, Elsevier, 1994, pp. 474–485, ISBN: 978-0-12-336156-1. DOI: 10.1016/B978-0-12-336156-1.50061-6. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/B9780123361561500616 (visited on 09/19/2023).

[74] J. Lv and J. Fang, "A color distance model based on visual recognition", *Mathematical Problems in Engineering*, vol. 2018, e4652526, May 20, 2018, ISSN: 1024-123X. DOI: 10.1155/2018/4652526. [Online]. Available: https://www.hindawi.com/journals/mpe/2018/4652526/ (visited on 11/21/2023).

[75] "Groupgets/GetThermal: Cross-platform USB thermal camera viewer". (), [Online]. Available: https://github.com/groupgets/GetThermal (visited on 09/19/2023).

[76] H. Xu, E. G. Bazavan, A. Zanfir, W. T. Freeman, R. Sukthankar, and C. Sminchisescu, "GHUM & GHUML: Generative 3d human shape and articulated pose models", in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA: IEEE, Jun. 2020, pp. 6183–6192, ISBN: 978-1-72817-168-5. DOI: 10.1109/CVPR42600.2020.00622. [Online]. Available: https://ieeexplore.ieee.org/document/9157563/ (visited on 10/17/2023).

[77] D. P. Kingma and M. Welling, *Auto-encoding variational bayes*, Dec. 10, 2022. DOI: 10.48550/arXiv.1312.6114. arXiv: 1312.6114[cs,stat]. [Online]. Available: http://arxiv.org/abs/1312.6114 (visited on 12/11/2023).

[78] H. Joo *et al.*, "Panoptic studio: A massively multiview system for social motion capture", in *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile: IEEE, Dec. 2015, pp. 3334–3342, ISBN: 978-1-4673-8391-2. DOI: 10.1109/ICCV.2015.381. [Online]. Available: http://ieeexplore.ieee.org/document/7410738/ (visited on 12/13/2023).

[79] C. R. Harris *et al.*, "Array programming with NumPy", *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020, ISSN: 1476-4687. DOI: 10.1038/s41586-020-2649-2. [Online]. Available: https://www.nature.com/articles/s41586-020-2649-2 (visited on 12/11/2023).

[80] I. Lifshitz, E. Fetaya, and S. Ullman, "Human pose estimation using deep consensus voting", in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., ser. Lecture Notes in Computer Science, Cham: Springer International Publishing, 2016, pp. 246–260, ISBN: 978-3-319-46475-6. DOI: 10.1007/978-3-319-46475-6_16.

[81] R. Josyula and S. Ostadabbas, *A review on human pose estimation*, Oct. 13, 2021. DOI: 10.48550/arXiv.2110.06877. arXiv: 2110.06877[cs]. [Online]. Available: http://arxiv.org/abs/2110.06877 (visited on 11/01/2023).

[82] T. v. Marcard, G. Pons-Moll, and B. Rosenhahn, "Human pose estimation from video and IMUs", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 8, pp. 1533–1547, Aug. 2016, ISSN: 1939-3539. DOI: 10.1109/TPAMI.2016.2522398. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7393844 (visited on 11/01/2023).

[83] D. Pavllo, C. Feichtenhofer, D. Grangier, and M. Auli, "3d human pose estimation in video with temporal convolutions and semi-supervised training", presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 7753–7762. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2019/html/Pavllo_3D_Human_Pose_Estimation_in_Video_With_Temporal_Convolutions_and_CVPR_2019_paper.html (visited on 11/01/2023).

[84] H. Qiu, C. Wang, J. Wang, N. Wang, and W. Zeng, "Cross view fusion for 3d human pose estimation", presented at the Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 4342–4351. [Online]. Available: https://openaccess.thecvf.com/content_ICCV_2019/html/Qiu_Cross_View_Fusion_for_3D_Human_Pose_Estimation_ICCV_2019_paper.html (visited on 11/01/2023).

[85] A. Marmol, T. Peynot, A. Eriksson, A. Jaiprakash, J. Roberts, and R. Crawford, "Evaluation of keypoint detectors and descriptors in arthroscopic images for feature-based matching applications", *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 2135–2142, Oct. 2017, ISSN: 2377-3766. DOI: 10.1109/LRA.2017.2714150. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7945244 (visited on 11/01/2023).

[86] Y. Zhang *et al.*, "Build your own hybrid thermal/EO camera for autonomous vehicle", in *2019 International Conference on Robotics and Automation (ICRA)*, Montreal, QC, Canada: IEEE, May 2019, pp. 6555–6560, ISBN: 978-1-5386-6027-0. DOI: 10.1109/ICRA.2019.8794320. [Online]. Available: https://ieeexplore.ieee.org/document/8794320/ (visited on 12/12/2023).

B Acronyms

AAS Appearance Analysis weighted Saliency 15

ABCD Absolute Basic Color Difference 24, 26–28

AHE Adaptive Histogram Equalization 23, 24, 26

ARBLOT Average Relative Bone Length Over Time 29, 35, 40, 41, 43, 44, 47

CDF Cumulative Distribution Function 23, 24, 26

CLAHE Contrast Limited Adaptive Histogram Equalization 23–26, 29, 43

CNN Convolutional Neural Network 5, 9–12, 16

DUC Dense Upsampling Convolution 13

HAR Human Action Recognition 16

HPE Human Pose Estimation 5, 6, 8–12, 16, 29, 32, 34, 35, 40, 47, 48

LAHE Local Area Histogram Equalization 24

LCGS Lossless CLAHE Grayscale 30, 41–46

LCHSV Lossless CLAHE HSV 30, 41–46

LCIRBL Lossless CLAHE Ironblack 31, 41–46

LLGS Lossless Linear Grayscale 30, 37, 41–46

LLHSV Lossless Linear HSV 30, 41–46

LLIRBL Lossless Linear Ironblack 30, 41–46

LUT LookUp Table 24, 26–29

LYCGS Lossy CLAHE Grayscale 31, 41–46

LYCHSV Lossy CLAHE HSV 31, 41–46

LYCIRBL Lossy CLAHE Ironblack 32, 41–46

LYLGS Lossy Linear Grayscale 31, 41–46

LYLHSV Lossy Linear HSV 31, 41–46

LYLIRBL Lossy Linear Ironblack 31, 41–46

M3DHPE Monocular Three-Dimensional Human Pose Estimation 6–9, 11, 12, 33, 37, 40, 41, 47, 48

MPJPE Mean Per Joint Position Error 34, 36, 41, 42, 44, 45, 47

NN Neural Network 12, 13, 16, 17, 22, 24

PCK Percentage of Correct Keypoints 34, 41–43, 45–47

PIR Percentage of Inference Results 34, 41, 42, 47

RBL Relative Bone Lengths 35

ROI Region of interest 9–13, 15, 21

SOTA State Of The Art 9, 15, 48

SSD Single Shot MultiBox Detector 9

TAS Thermal Analysis based Saliency 15

THR Thermographic Human Recognition 8, 15