Hochschule Karlsruhe
University of
Applied Sciences

Fakultät für
Informatik und
Wirtschaftsinformatik

Master Thesis

Name:	Matthias Möller
Topic:	Human Keypoint Detection from Partial Views
Place of work:	Intelligent Systems Research Group, Karlsruhe
Supervisor:	Prof. Dr.-Ing. Laubenheimer
Co-examiner:	Prof. Dr. Wölfel
Deadline:	14/03/2025

Karlsruhe, 15/09/2024

The Chairman of the examination committee

H. Kö

Prof. Dr. Heiko Körner

Statement of Authorship

I hereby declare that this work is my own and that I have not used any sources, tools, or assistance other than those explicitly acknowledged. All sources and materials taken from the work of others, whether quoted, paraphrased, or otherwise used, have been properly cited and referenced.

Karlsruhe, March 14, 2025

(Matthias Möller)

Abstract

Accurate human pose estimation in cases where the subject is partially truncated or occluded remains challenging. This study explores methods to improve human pose estimation under truncation conditions. Segmentation-guided attention and multi-layer segmentation conditioning (ControlNet) using body part segmentations as auxiliary information are evaluated in both HRNet, a convolutional neural network, and Sapiens-0.3B, a transformer-based model. A 3D scanner dataset, capturing only partially visible poses, was annotated in this study to serve as a test dataset for evaluating performance in challenging real-world scenarios. Results show that the chosen segmentation-based approaches, segmentation attention and ControlNet, yield moderate improvements in average precision by providing structural cues but do not consistently enhance fine-grained keypoint localization when significant portions of the body are missing. While no significant performance gains for human pose estimation under truncated conditions is achieved by either segmentation-guided attention or ControlNet, ControlNet offers slight advantages over single-stage segmentation attention, suggesting that multi-layer segmentation methods help models by incorporating the additional spatial context at multiple processing stages, allowing them to infer more plausible keypoint locations in partially visible poses.

In contrast, augmented training data, including truncated images, provides consistent performance gains by exposing the model to partial-pose inputs during training. These findings highlight the need to ensure that the training data closely reflects real-world conditions to improve model performance. While refining segmentation granularity and implementing adaptive weighting mechanisms could improve the model's ability to handle partial poses, significant truncation can still limit the usefulness of segmentation information. Further research should explore whether increasing the detail and precision of segmentation maps can provide more meaningful structural cues for pose estimation in truncated conditions and how this translates to human pose estimation under occlusion. Additionally, to achieve the most significant performance gains, future work should develop more advanced data augmentation techniques that better simulate partial poses and analyze the distribution of missing body parts due to truncation to optimize augmentation strategies accordingly.

Kurzfassung

Die exakte Schätzung menschlicher Posen stellt eine Herausforderung dar, insbesondere bei Personen, die teilweise abgeschnitten oder verdeckt sind. Diese Studie untersucht Methoden zur Optimierung der menschlichen Posenschätzung in solchen Fällen. Zu diesem Zweck werden segmentierungsbasierte Aufmerksamkeitsmechanismen (Segmentation-Guided Attention) und Multi-Layer-Segmentierungs-Konditionierung (ControlNet) mit Körperteil-Segmentierungen als zusätzliche Information evaluiert. Die Evaluierung erfolgt sowohl in HRNet, einem faltenden neuronalen Netzwerk, als auch in Sapiens-0.3B, einem transformerbasierten Modell. Zur Leistungsbewertung wird ein 3D-Scanner-Datensatz verwendet, der ausschließlich Bilder von teilweise sichtbaren Posen enthält und im Rahmen dieser Studie annotiert wurde.

Die Resultate demonstrieren, dass die segmentierungsbasierten Ansätze Segmentation Attention und ControlNet moderate Verbesserungen der durchschnittlichen Präzision ermöglichen, indem sie strukturelle Informationen bereitstellen. Jedoch resultieren sie nicht in einer konsistent besseren Keypoint-Lokalisierung, wenn wesentliche Körperregionen fehlen. In Fällen teilweiser Sichtbarkeit zeigen weder Segmentation-Guided Attention noch ControlNet signifikante Leistungssteigerungen, jedoch weist ControlNet leichte Vorteile gegenüber der einstufigen Segmentation Attention auf. Dies legt nahe, dass Multi-Layer-Segmentierungsmethoden durch zusätzliche räumliche Kontextinformationen plausiblere Keypoints ableiten können.

Die Verwendung von erzeugten oder erweiterten Trainingsdaten, die explizit abgeschnittene Posen enthalten, resultiert in einer konsistenten Leistungssteigerung, da das Modell gezielt auf partielle Posen trainiert wird. Dies betont die Relevanz einer realitätsnahen Verteilung der Trainingsdaten. Eine Verfeinerung der Segmentierungsgranularität sowie adaptive Gewichtungsmechanismen könnten die Verarbeitung partieller Posen weiter optimieren. Allerdings ist der Nutzen segmentierungsbasierter Informationen bei stark abgeschnittenen Posen begrenzt. Zukünftige Forschung sollte daher untersuchen, inwiefern präzisere Segmentierungsmasken relevante strukturelle Hinweise liefern und wie sich dies auf verdeckte Posen auswirkt. Darüber hinaus sollten fortschrittliche Datenverarbeitungstechniken entwickelt werden, um partielle Posen besser zu simulieren und die Verteilung fehlender Körperteile für optimierte Augmentationsstrategien zu analysieren.

Notation

Acronyms

AP	Average Precision
AR	Average Recall
BCA	Body-Cropping Augmentation
COCO	Common Objects in Context
CNN	Convolutional Neural Network
CPN	Cascaded Pyramid Network
FFN	Feed-forward Network
FPN	Feature Pyramid Networks
HPE	Human Pose Estimation
HRNet	High-Resolution Network
IoU	Intersection over Union
MAE	Masked Autoencoder
NLP	Natural Language Processing
OKS	Object Keypoint Similarity
PCK	Percentage of Correct Keypoints
PPNet	Position Puzzle Network
ReLU	Rectified Linear Unit
SOTA	State-of-the-Art
ViT	Vision Transformer

Contents

1 Introduction

Human pose estimation is a foundational task in computer vision, which involves detecting and estimating the spatial configurations of human bodies in images or videos. This field has gained significant attention in recent years due to advancements in machine learning and deep learning, leading to enhanced estimation of human postures, gestures, and movements [Gao+25]. Determining the positions of key anatomical body points and limbs enables the interpretation of human actions in a wide range of applications.

Pose estimation is crucial in several domains, including healthcare, sports, entertainment, and security. In healthcare, it enables remote physical therapy and rehabilitation by allowing practitioners to monitor patient movements [LVX20]. In sports analytics, pose estimation aids in performance evaluation and the biomechanical analysis of athletes [BM21]. The entertainment industry uses pose estimation to create realistic animations and improve virtual and augmented reality experiences by improving immersion and interactivity [AP22]. Moreover, human-computer interaction benefits from pose estimation by enabling more natural and intuitive user interfaces, leading to closer interaction between humans and machines [Bau+15]. In security and surveillance, enhanced pose estimation techniques contribute to activity recognition and threat detection by analyzing human actions more precisely [Zan+23].

Significant progress has been made in the past decade, but several challenges remain. One major obstacle is the variability in human appearances, including differences in clothing, body shapes, and sizes, which complicates model generalization to accommodate all differences and still make accurate predictions. Environmental factors, including different lighting conditions, camera angles, and cluttered backgrounds further impact accuracy [Jia+24]. Additionally, occlusions and truncated poses pose significant difficulties. While truncated poses result in the complete absence of body parts from the image, occluded poses may still be estimated using prior knowledge of human anatomy. However, accurately predicting occluded body parts remains challenging for state-of-the-art models [Han+25].

As research in human pose estimation progresses, addressing these challenges is important to enhance model robustness.

1.1 Objective and Motivation

The primary objective of this study is to enhance the performance of human pose estimation models in scenarios where poses are only partially visible due to truncation. This is realized by integrating body part segmentation, which provides anatomical context, into the model pipeline, to enhance model robustness. By incorporating segmentation, the models gain a structured understanding of human anatomy, enabling more accurate estimations even when key body parts are missing or occluded.

A central component of this research is the creation of annotations for a task-specific test dataset. The image data was previously captured using a specialized multi-camera setup consisting of 16 cameras arranged in a 3D scanning configuration. This configuration results in images where human poses are only partially visible in individual frames, as shown in Figure 1.1, which poses significant challenges for pose estimation models. As part of this study, manual annotations are created for these images to ensure accurate labeling of the test dataset.

The multi-camera 3D scanning setup allows capturing of human poses from multiple angles and perspectives, ensuring a comprehensive and challenging dataset for evaluating model performance. By systematically analyzing the impact of truncations, this study contributes to the development of more robust and generalizable human pose estimation techniques.

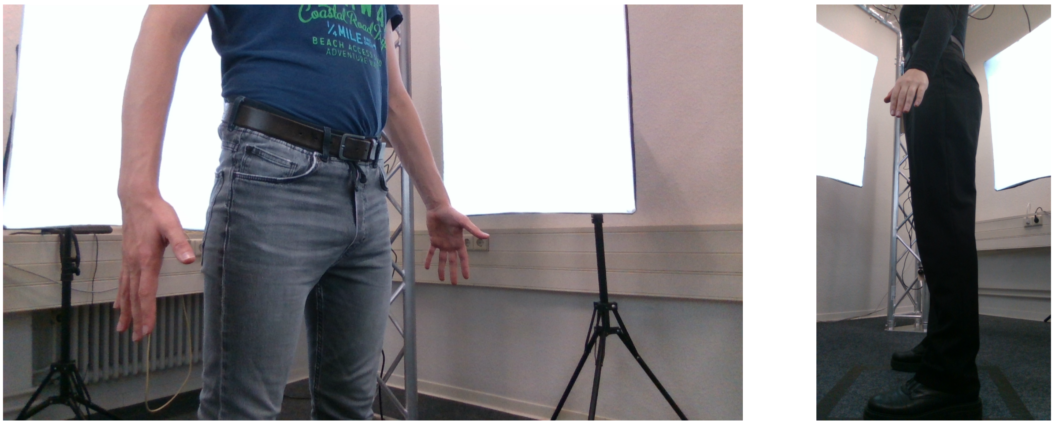


Figure 1.1: Images captured using the 3D scanner setup.

1.2 Environment

The research in this study is undertaken at the *Intelligent Systems Research Group* (ISRG), a research institute of the *Hochschule Karlsruhe - University of Applied Science.* The ISRG specializes in machine learning, computer vision and optimization, driving both theoretical advances and practical applications. Its research results are integrated into industrial and economic sectors, strongly focusing on 3D modeling, computer vision, materials science, and manufacturing processes. In addition, the group explores fault diagnosis, anomaly detection, and optimization to improve system efficiency and reliability.

1.3 Structure

This document is structured into five chapters beyond the introduction. Chapter 2 lays the foundational concepts of human pose estimation, introducing different deep learning based methods and data augmentation techniques to help model generalization. Chapter 3 reviews the state-of-the-art methods and related work important in this research. In Chapter 4, the methodology of this study is presented, detailing the experimental design and methods. Chapter 5 discusses the outcomes of the experiments, presenting both the raw results and an in-depth analysis to interpret the findings. The final chapter, Chapter 6, concludes the study, summarizing the key discoveries and proposing potential avenues for future research.

2 Basics

This chapter lays the foundation for understanding the core principles and methods central to this research and provides a comprehensive overview of key concepts in 2D human pose estimation and related areas. The chapter begins with an introduction to human pose estimation (HPE), focusing on keypoint-based 2D pose estimation from monocular images. This foundational section sets the stage for exploring different approaches to pose estimation, including top-down, bottom-up, regression-based, and heatmap-based methods. Each approach is discussed in detail, highlighting its strengths and applications. The chapter then delves into three deep learning-based methods commonly used in HPE: Convolutional Neural Networks (CNNs), Vision Transformers (ViTs) and Masked Autoencoders (MAEs). These methods are analyzed regarding their architectural designs, advantages and relevance to pose estimation tasks, providing insight into their contributions to the field. The discussion then shifts to segmentation, with particular emphasis on body part segmentation. Finally, the chapter concludes with an examination of data augmentation techniques. The importance of data augmentation in human pose estimation is highlighted, and selected techniques are presented to demonstrate their impact on model performance.

2.1 Human Pose Estimation

Human Pose Estimation (HPE) aims to estimate the pose of the human body, typically in images or video sequences. HPE finds applications in different fields, including action recognition, human-computer interaction, virtual and augmented reality, and the security sector. Generally, HPE can be classified into *2D pose human estimation* and *3D pose human estimation*, with the former focusing on detecting human poses within a two-dimensional plane, such as an image, while the latter aims to estimate poses in three-dimensional space [Zho+23].

Different representations of the human body have been proposed [Kna24]. These representations can be categorized into three primary models: the kinematic, planar and volumetric models [Zho+23]. The kinematic model, which is the most widely used, represents the human body through joint or keypoint positions and limb orientations, effectively capturing its structure. An example of such a representation is shown in Figure 2.1. These keypoints are most often anatomical joints of the human body (e.g., shoulders, elbows, wrists, hips, knees, and ankles), but can also include other points such as the nose, eyes, and more.

The keypoints and connections between them can be interpreted as a graph with nodes and edges connecting the nodes and allow for the modeling of different poses and configurations of the human body, providing a flexible and efficient representation.

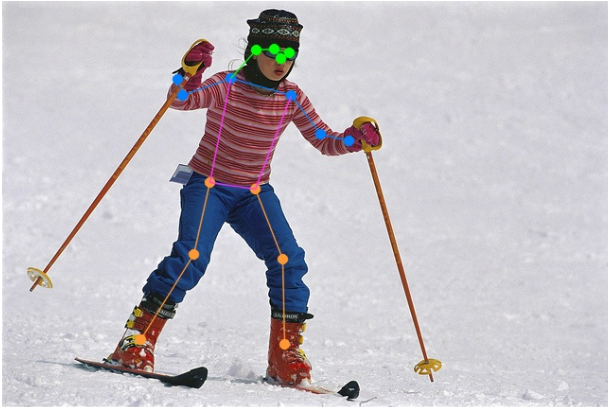


Figure 2.1: Kinematic model-based pose representation of an image from the MS COCO dataset [Lin+15].

Alternative representations include the planar model, which uses rectangles to approximate body shape and appearance, and the volumetric model, which employs mesh data to capture finer body shape details. While these approaches may provide more detailed representations, they also introduce higher computational complexity [Gao+25]. However, in many practical applications, the kinematic model remains the preferred choice due to its balance between simplicity and accuracy.

2D pose estimation methods can generally be categorized into *top-down* and *bottom-up* approaches [Lan+23; Zho+23]. In *top-down* methods, the process begins with detecting each person in the image by identifying a region, known as a bounding box, that closely encapsulates the individual. Once detected, keypoints are estimated within each bounding box to determine the person's pose. Conversely, *bottom-up* approaches first detect all keypoints across the entire image before grouping them into individual poses. This method avoids the need for an initial person detection step, instead focusing on grouping detected keypoints into coherent human figures.

Despite major breakthroughs, 2D HPE still faces challenges due to occlusion, challenging backgrounds, and variability in human appearances. Modern deep learning techniques, such as CNNs and Transformer-based architectures have shown substantially improved performance by learning robust feature representations from large datasets [Lan+23].

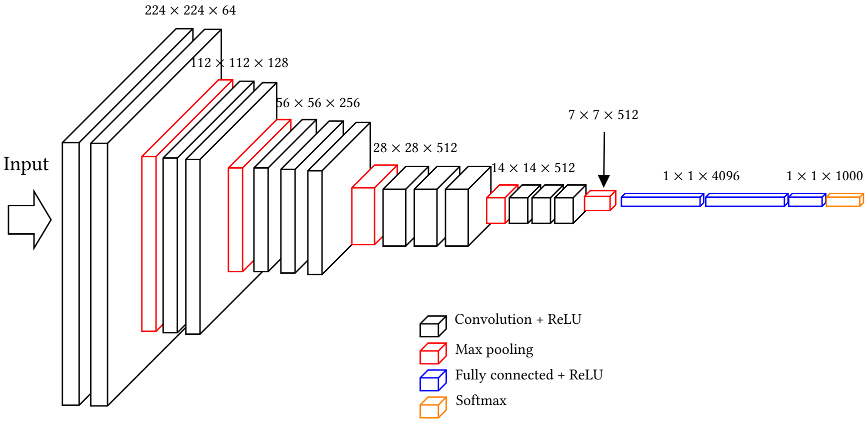


Figure 2.2: Illustration of the VGG16 [SZ15] architecture,[1] showcasing its sequential deep convolutional layers, pooling operations, and fully connected layers.

2.2 Deep Learning-Based Methods

Early approaches to HPE primarily relied on handcrafted features. However, recent advancements in deep learning have had a significant impact on the field, leading to more accurate and robust detection of keypoints in images [Kna24].

This section introduces three prominent deep learning-based methods for HPE: Convolutional Neural Networks, Vision Transformers, and Masked Autoencoders.

2.2.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) have demonstrated remarkable success for several computer vision tasks, including image classification, object detection and human pose estimation [Sun+19; NYD16]. They apply local receptive fields (convolutions) to learn spatially correlated features at different scales and depths [Lec+98; KSH12]. By hierarchically extracting relevant information, CNNs can detect complex structures in images while preserving spatial relationships. Figure 2.2 shows the VGG16 [SZ15] model architecture, showcasing its sequential convolutional layers, pooling operations, and fully connected layers. This model is widely used for image classification and feature extraction in deep learning applications.

Core Building Blocks of a CNN

A CNN consists of multiple hierarchical layers, each responsible for different levels of feature extraction. The most fundamental component is the convolutional layer, where filters (or

[1] VGG16 figure by hongvin: https://github.com/hongvin/Neural-Network-Architectures-in-LaTeX

Kernels) are applied to extract spatial patterns from the input. Mathematically, a convolution operation at layer l is defined as:

$$\mathbf{F}_l = \sigma(\mathbf{W}_l * \mathbf{F}_{l-1} + \mathbf{b}_l) \tag{2.1}$$

where $\mathbf{F}_l$ represents the feature map, $\mathbf{W}_l$ and $\mathbf{b}_l$ are the trainable filter weights and biases, respectively, and $\sigma(\cdot)$ is a non-linear activation function such as the Rectified Linear Unit (ReLU) [KSH12]. These filters allow CNNs to capture local spatial dependencies while maintaining translation invariance.

To introduce non-linearity, activation functions are applied after convolutions. The most commonly used function is ReLU, defined as:

$$f(x) = \max(0, x) \tag{2.2}$$

ReLU improves gradient propagation and helps mitigate the vanishing gradient problem, allowing for deeper network architectures.

Pooling and Downsampling

Pooling layers are used to progressively reduce the spatial dimensions of feature maps while preserving important information. The two most common types are max pooling and average pooling. Max pooling selects the highest value within a given window, while average pooling computes the mean value of all values in the window. Pooling ensures that small translations in the input image do not drastically change feature representations.

While pooling layers reduce computational complexity and help in generalization, excessive downsampling can lead to loss of information. To address this, alternative techniques such as strided and dilated convolutions have been introduced, allowing networks to maintain high-resolution feature maps while controlling the receptive field size.

Fully Connected Layers and Output

After a series of convolutional and pooling operations, the extracted feature maps are flattened and passed through fully connected layers, which perform high-level reasoning and classification. Each neuron in a fully connected layer is connected to every neuron in the previous layer. In human pose estimation, the output layer predicts keypoint coordinates representing body joints, often using regression or heatmap-based techniques.

Hierarchical Feature Learning in CNNs

One of the strengths of CNNs is their ability to learn hierarchical representations due to their architecture. Early layers typically detect simple features such as edges, textures, and small patterns. With increasing depth, the network captures more complex structures, including shapes and body parts, ultimately leading to high-level semantic understanding. This hierarchical feature extraction is beneficial for human pose estimation, where precise relationships between body joints must be inferred.

Architectural Variants and Their Relevance to Pose Estimation

While early CNN architectures like AlexNet [KSH12] and VGG [SZ15] demonstrated the potential of deep learning on large-scale datasets, their direct application to human pose estimation posed challenges. Standard CNNs struggle with capturing multi-scale information, handling occlusions, and ensuring efficient gradient flow in deeper network architectures.

To address these issues, several architectural improvements have been introduced. Increasing CNN depth often leads to vanishing gradients (where gradients become too small during backpropagation, preventing effective weight updates), limiting learning. ResNet [He+16] mitigates this issue by introducing identity-based skip connections:

$$\mathbf{y} = \mathbf{x} + f(\mathbf{x}) \tag{2.3}$$

which allows gradients to flow more effectively through the network, enabling deeper architectures.

Another challenge in pose estimation is efficient feature reuse. DenseNet [Hua+17] addresses this by connecting each layer to all preceding layers. This promotes feature reuse, helping pose estimation networks learn shared representations across joints and reducing overfitting, but it also increases the computational complexity.

Since body parts appear at varying scales, multi-scale feature learning is essential. Feature Pyramid Networks (FPN) [Lin+17] and Hourglass Networks [NYD16] tackle this by aggregating information across different resolutions, improving robustness in detecting keypoints under varying conditions.

Finally, retaining high-resolution spatial details is important for precise keypoint localization. HRNet [Sun+19] uses multiple branches with different resolutions, maintaining high-resolution feature maps throughout the network instead of downsampling early in the architecture. This approach preserves fine-grained spatial information, improving the detection of small joints.

These advancements have significantly enhanced CNN-based human pose estimation, making them more robust and accurate in real-world applications.

2.2.2 Vision Transformer

Transformer [Vas+23] were originally designed for sequence modeling in natural language processing (NLP). The success in language tasks, coupled with the increased compute capabilities of GPUs have paved the way for Vision Transformer (ViT) [Dos+21], which use global self-attention instead of convolutions.

Transformer Architecture

The Transformer model, introduced by Vaswani et al. in *Attention is All You Need* [Vas+23] on which the following introduction to Transformer is based on, has revolutionized various fields, including NLP and computer vision through its self-attention mechanism.

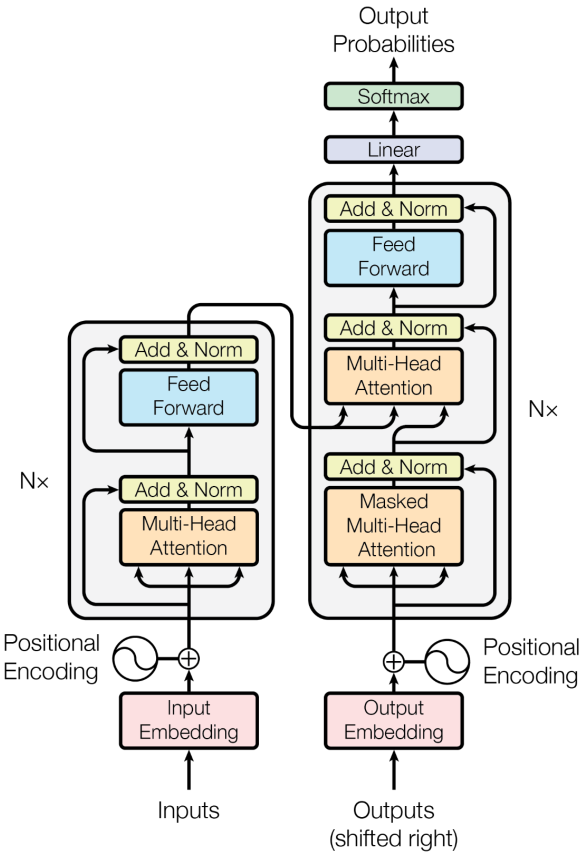


Figure 2.3: Architecture of the Transformer model, adapted from [Vas+23].

A Transformer, as illustrated in Figure 2.3, consists of two main components: an encoder (left) and a decoder (right), each composed of a stack of N identical layers.

The encoder layers have three main components: a multi-head self-attention mechanism, a fully connected feed-forward network (FFN), and layer normalization with residual connections. Each encoder layer processes input embeddings using self-attention to capture relationships between tokens, followed by the FFN to enhance feature representations. The decoder layers introduce an additional component: a masked multi-head attention mechanism that attends to the encoder's output while ensuring that predictions at position i only depend on previously generated outputs.

Input Embeddings and Positional Encodings

Before being processed by the Transformer, the raw input data must converted into a numerical representation that the model can process. This is achieved through *input embeddings*, which transform discrete tokens (e.g., words in NLP or patches in vision tasks) into continuous vector representations in a high-dimensional space. Given an input sequence of tokens, each token is mapped to a corresponding embedding vector using an embedding matrix learned during training. Mathematically, for an input sequence $\mathbf{X}$, the embeddings are obtained as follows:

$$\mathbf{E} = \mathbf{X}\mathbf{W}_E \tag{2.4}$$

where $\mathbf{W}_E$ is the learned embedding matrix, and $\mathbf{E}$ represents the resulting sequence of embeddings. These embeddings capture semantic relationships between tokens, allowing the model to process complex patterns in the data.

Since the Transformer processes tokens in parallel rather than sequentially, it does not inherently encode the order of elements in a sequence. To address this, *positional encodings* are added to the input embeddings, providing explicit information about token positions in the sequence. The positional encoding vector $\mathbf{PE}(pos)$ is computed using sinusoidal functions as follows:

$$\mathbf{PE}_{(pos,2i)} = \sin\left(\frac{pos}{10000^{2i/d_{\text{model}}}}\right) \tag{2.5}$$

$$\mathbf{PE}_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{2i/d_{\text{model}}}}\right) \tag{2.6}$$

where pos is the position of the token in the sequence, i is the dimension index, and d_{model} is the total dimensionality of the embeddings. The alternating sine and cosine functions ensure that the positional encodings produce unique representations for different positions while allowing the model to generalize to longer sequences than those seen during training. The final input to the Transformer is obtained by adding the positional encodings to the corresponding input embeddings:

$$\mathbf{Z} = \mathbf{E} + \mathbf{PE} \tag{2.7}$$

This allows the model parallel computation while still maintaining information about the token positions and enables the use of self-attention.

Self-Attention Mechanism

The self-attention mechanism is the core of Transformer networks. It allows each element to *pay attention* to all other elements, capturing dependencies independent of their distance to each other.

This is achieved by computing attention scores (how important elements are for each other) between elements using three components: queries ($\mathbf{Q}$), keys ($\mathbf{K}$), and values ($\mathbf{V}$).

Queries, keys, and values represent different projections of the input sequence, each calculated by applying learned linear transformations. For each token in a sequence, its own query, key, and value vectors are calculated. Query vectors determine the relevance of other tokens, key vectors represent the tokens to be compared, and value vectors contain the actual data of the tokens.

The self-attention mechanism operates as follows:

$$\mathbf{Q} = \mathbf{Z}\mathbf{W}_Q, \quad \mathbf{K} = \mathbf{Z}\mathbf{W}_K, \quad \mathbf{V} = \mathbf{Z}\mathbf{W}_V \tag{2.8}$$

where $\mathbf{Z}$ represents the sum of the input embeddings and positional encodings, and $\mathbf{W}_Q$, $\mathbf{W}_K$, and $\mathbf{W}_V$ are learned weight matrices. The attention scores are computed using the scaled dot-product of the queries and keys:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \tag{2.9}$$

Here, d_k is the dimensionality of the keys. The scaling factor $\sqrt{d_k}$ helps to mitigate the impact of large dot-product values, which can make the softmax function produce very small gradients.

Multi-Head Attention

Multi-head attention is used to further increase the model's ability to focus on different parts of the input sequence. Several self-attention operations are applied in parallel, as shown in Figure 2.4. Each of these attention heads has its own learned projections, allowing the model to capture different relationships between elements.

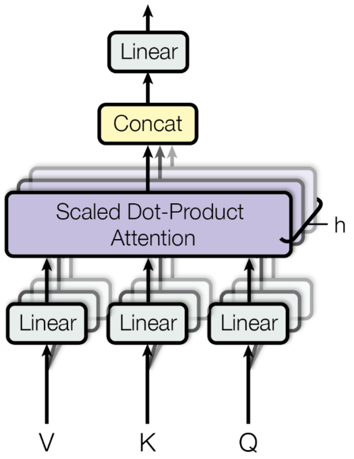


Figure 2.4: Illustration of the multi-head attention mechanism from [Vas+23].

Each head i computes its own attention function:

$$\text{head}_i = \text{Attention}(\mathbf{Q}\mathbf{W}_{Q_i}, \mathbf{K}\mathbf{W}_{K_i}, \mathbf{V}\mathbf{W}_{V_i}), \quad i = 1, \dots, h \tag{2.10}$$

where $\mathbf{W}_{Q_i}$, $\mathbf{W}_{K_i}$, and $\mathbf{W}_{V_i}$ are the weight matrices for the i-th head. The outputs of all heads are concatenated and transformed by a final weight matrix $\mathbf{W}_O$:

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)\mathbf{W}_O \tag{2.11}$$

Feed-Forward Neural Network

After self-attention is applied, each token representation is processed through a fully connected FFN. The FFN consists of two linear transformations with a non-linear activation function between them:

$$\text{FFN}(\mathbf{x}) = \max(0, \mathbf{x}\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2 \tag{2.12}$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ are learned weight matrices, and $\mathbf{b}_1$ and $\mathbf{b}_2$ are bias terms. The activation function, typically ReLU, introduces non-linearity into the model.

Patch Embeddings for Vision Transformers

A fundamental challenge in applying Transformers to image data stems from the large number of pixels in images. Unlike text, which in comparison consists of a relatively short sequence of words, an image at standard resolution contains millions of pixels. Directly treating each pixel as an input token would make the computational cost of self-attention infeasible.

To address this, Dosovitskiy et al. proposed the *Vision Transformer (ViT)* [Dos+21]. Figure 2.5 illustrates the ViT architecture and the used encoder. Instead of using a stack of decoder blocks, ViT uses the encoder part of Transformers to learn a rich latent representation of the input image, followed by a multi-layer perceptron, which is another name for an FFN, for classification. The ViT architecture can be adapted for different vision tasks by using different *heads* to make predictions using the latent representation.

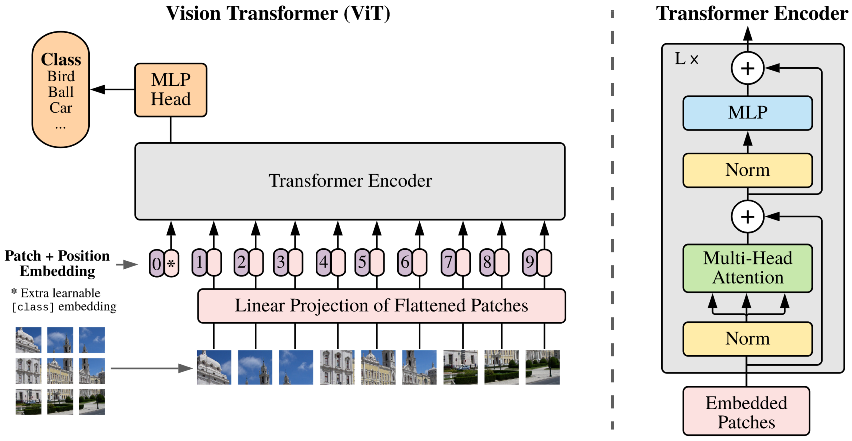


Figure 2.5: Vision Transformer architecture (left) [Dos+21] and Transformer encoder used in ViT (right) [Dos+21; Vas+23].

ViT segments an image $\mathbf{I} \in \mathbb{R}^{H \times W \times C}$ into N non-overlapping patches of size $P \times P$:

$$N = \frac{HW}{P^2} \tag{2.13}$$

Each patch is then flattened into a vector $\mathbf{x}_i \in \mathbb{R}^{P^2 C}$ by concatenating its pixel values. These patch vectors are subsequently projected into a D-dimensional embedding space using a trainable linear transformation:

$$\mathbf{z}_i = \mathbf{W}_e \mathbf{x}_i + \mathbf{b}_e, \quad i = 1, \ldots, N \tag{2.14}$$

The resulting vectors, known as patch embeddings, effectively reduce the sequence length from the total number of pixels to N patches. This transformation makes self-attention computationally feasible, as the number of tokens processed by the Transformer is now significantly lower.

Since Transformers lack an inherent sense of spatial structure, a learnable positional embedding $\mathbf{p}_i$ is added to each patch embedding $\mathbf{z}_i$ to provide positional information:

$$\mathbf{z}_i^0 = \mathbf{z}_i + \mathbf{p}_i \tag{2.15}$$

The positionally-encoded patch embeddings then serve as the input token sequence for the Transformer encoder. This method ensures that spatial relationships within the image are preserved, enabling the model to learn spatial hierarchies despite the absence of convolutional operations effectively.

2.2.3 Masked Autoencoders

CNNs and ViTs follow the supervised learning paradigm, requiring labeled data for training. However, collecting labeled datasets is time-consuming and expensive. As the demand for annotated data increases, research in self-supervised learning has gained significant attention. Masked Autoencoders (MAEs) [He+22] extend the Transformer paradigm to self-supervised image modeling, drawing inspiration from masked language modeling in NLP, such as BERT [Dev+19].

Basic Concept and Motivation

Like ViTs, MAEs divide an image into a series of non-overlapping patches that serve as visual tokens. Unlike standard ViTs, where all patches are processed equally, MAEs employ random masking during training, discarding a significant portion of the patches (commonly 75%) while retaining only a subset (25%) for processing. The model is then tasked with reconstructing the missing patches from the remaining visible ones. This process is visualized in Figure 2.6.

This self-supervised pretraining strategy forces the model to learn a compact latent representation of the visible content, capturing local and global semantics necessary to infer the missing regions. By relying on partial information, MAEs develop strong feature representations, which can later be fine-tuned for downstream vision tasks such as classification, object detection, and pose estimation.

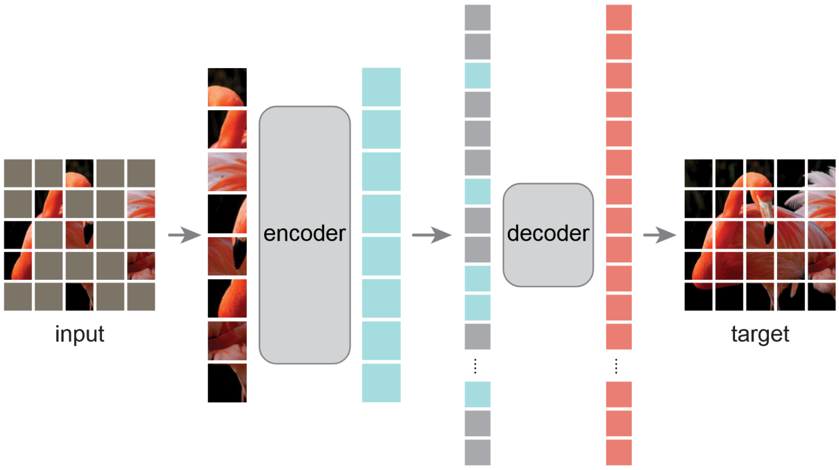


Figure 2.6: Approach of the Masked Autoencoder (MAE) proposed by [He+22].

A major advantage of MAEs is computational efficiency during pretraining. Since the encoder processes only a fraction of the total patches, the computational overhead is significantly lower than fully supervised ViTs. This reduction in complexity makes MAEs highly scalable for large-scale image modeling while maintaining high representational capacity.

MAE Architecture

The MAE architecture consists of two main components: an encoder and a decoder. Both components follow the ViT framework, but their roles differ. The encoder is responsible for learning high-level representations from the visible patches while the decoder reconstructs the missing image content.

The encoder processes only the unmasked patches, leading to lower computational requirements than models that process all image patches. Let $\mathbf{Z} \in \mathbb{R}^{N \times D}$ be the complete set of patch tokens extracted from an image, where N represents the number of patches, and D is the embedding dimension. After applying random masking, only a subset of patches, denoted as $\mathbf{Z}_{\text{visible}}$ is retained and fed into the encoder:

$$\mathbf{H} = f_{\text{encoder}}(\mathbf{Z}_{\text{visible}}) \tag{2.16}$$

Each retained patch token consists of a patch embedding along with a positional encoding, which preserves spatial relationships between patches. The masked patches are discarded during encoding and do not contribute to the feature extraction. After obtaining the latent representation **H** from the encoder, a small, lightweight Transformer-based decoder reconstructs the missing patches. The decoder takes both the latent representation and a set of mask tokens representing the missing patches. These mask tokens are learnable parameters that provide a placeholder for the missing information, allowing the decoder to infer the original image content.

The reconstruction objective is formulated as a mean squared error (MSE) loss, applied only to the masked patches:

$$\mathcal{L}_{\text{reconstruction}} = \frac{1}{N_{\text{masked}}} \sum_{i \in \text{masked}} \|\mathbf{x}_{p,i} - \hat{\mathbf{x}}_{p,i}\|^2 \tag{2.17}$$

where $\mathbf{x}_{p,i}$ represents the original pixel values of the masked patches, and $\hat{\mathbf{x}}_{p,i}$ denotes the predicted reconstructions. By optimizing this loss function, the model learns to predict missing information based on its understanding of the visible patches.

Advantages and Applications

MAEs provide a computationally efficient approach to self-supervised learning by processing only a fraction of the input patches in the encoder. This significantly reduces training time and memory consumption compared to standard self-supervised learning techniques, which process the entire image. Furthermore, the need to reconstruct missing content forces the model to learn meaningful feature representations, which improves generalization in downstream tasks.

Self-supervised pretraining with MAEs has shown strong performance when fine-tuned on various vision tasks, including body part segmentation, depth estimation, normal estimation and human pose estimation [Khi+24]. The ability to learn from unlabeled data reduces the dependency on large labeled datasets, making MAEs an attractive approach for real-world applications where data annotation is costly.

2.2.4 Pose Estimation Heads

All three presented architectures, CNN, ViT and MAE, can be used as backbone networks for feature extraction from images. Two main approaches are commonly used to predict keypoint locations from the features extracted by these backbone networks: coordinate regression and heatmap generation.

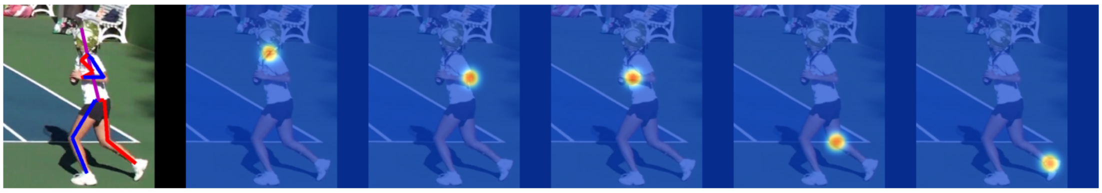


Figure 2.7: Example heatmap from the Stacked Hourglass Network, as presented in [NYD16].

Coordinate Regression

In this approach, the network directly regresses the coordinates of each joint. The final feature maps from the backbone are flattened and processed by fully connected layers to obtain keypoint predictions:

$$\hat{\mathbf{y}} = \mathbf{W}_{\text{fc}}\mathbf{F}_{\text{flattened}} + \mathbf{b}_{\text{fc}} \tag{2.18}$$

where $\hat{\mathbf{y}}$ represents the predicted keypoint locations, $\mathbf{F}_{\text{flattened}}$ is the flattened feature representation, and $\mathbf{W}_{\text{fc}}, \mathbf{b}_{\text{fc}}$ are the learnable weights and biases of the fully connected layer.

While coordinate regression offers a straightforward approach, it often struggles to capture spatial dependencies effectively, particularly in complex poses or when occlusions are present. This limitation stems from the fact that direct coordinate regression lacks the spatial structure that is inherently preserved by convolutional layers [Zha+19].

Heatmap Generation

A more widely adopted approach, especially in state-of-the-art methods, involves predicting a heatmap for each keypoint rather than regressing coordinates directly [XWW18; Sun+19; NYD16]. Instead of predicting coordinates directly, the model outputs a probability distribution over possible keypoint locations in the form of a heatmap, where each assigned value to a pixel corresponds to the probability that the keypoint is at that pixel. Such heatmaps are illustrated in Figure 2.7 for five keypoints. This approach enables more robust and spatially aware keypoint localization. This enhancement in performance is attributed to the incorporation of spatial information inherent in the image itself, preserving the relative positioning and the structure of body parts. Generating heatmaps in the same spatial domain as the input image inherently encodes contextual and geometric relationships between keypoints, increasing the approach's effectiveness in handling occlusions, perspective distortions, and varying poses [NYD16]. Consequently, networks adopting heatmap-based keypoint detection can learn spatial dependencies more effectively than direct coordinate regression methods.

The ground-truth heatmap for a keypoint k is typically modeled as a Gaussian distribution centered at the actual joint location:

$$\mathbf{H}_k(x, y) = \exp\left(-\frac{(x - x_k)^2 + (y - y_k)^2}{2\sigma^2}\right) \tag{2.19}$$

where (x_k, y_k) is the ground-truth keypoint position, (x, y) the estimated keypoint position, and σ controls the spread of the Gaussian. During training, the network learns to generate similar heatmaps, and during inference, the joint location is determined by identifying the peak response in the predicted heatmap.

Heatmap-based approaches have been shown to outperform direct regression methods because they use spatial context and local structures, making them more robust in complex scenarios [XWW18; Sun+19; NYD16]. In contrast to direct regression, which imposes precise coordinate predictions without accounting for spatial uncertainty, heatmaps distribute the probability of a keypoint over a region. This allows the model to handle occlusions and ambiguous poses more effectively.

However, heatmap estimation has its limitations. One significant issue arises from quantization errors, as keypoint locations are restricted to discrete pixel coordinates in the output heatmap. This constraint prevents sub-pixel precision without additional post-processing techniques. Various methods, such as Differentiable Soft-Argmax [Sun+19] and DARK (Distribution-Aware Refinement of Keypoints) [Zha+19], have been proposed to mitigate this limitation by refining keypoint predictions beyond pixel-level accuracy.

2.3 Segmentation

Segmentation techniques in computer vision are generally categorized into three primary types: semantic segmentation, instance segmentation, and panoptic segmentation.

Semantic segmentation assigns a class label, such as person, background, or object, to each pixel in an image. However, it does not differentiate between multiple instances of the same class, treating all objects of a given class as a single entity [LSD15].

Instance segmentation builds on the idea of semantic segmentation by classifying pixels and distinguishing between individual instances of the same class. For instance, in an image containing multiple people, instance segmentation labels each detected person separately (e.g., person 1, person 2), enabling object separation [He+17].

Panoptic segmentation integrates both semantic and instance segmentation by assigning every pixel in an image a class while differentiating between *stuff* (amorphous regions such as sky or grass, which aligns with semantic segmentation principles) and *things* (countable objects such as people and cars, which require instance-level differentiation) [Kir+19].

Body Part Segmentation

Body part segmentation is a specialized variant of semantic segmentation, where each pixel is assigned to a specific anatomical region (e.g., head, torso, arms, legs). Unlike instance segmentation, which identifies separate objects of the same category, body part segmentation provides a fine-grained delineation of the human body, as shown in Figure 2.8.

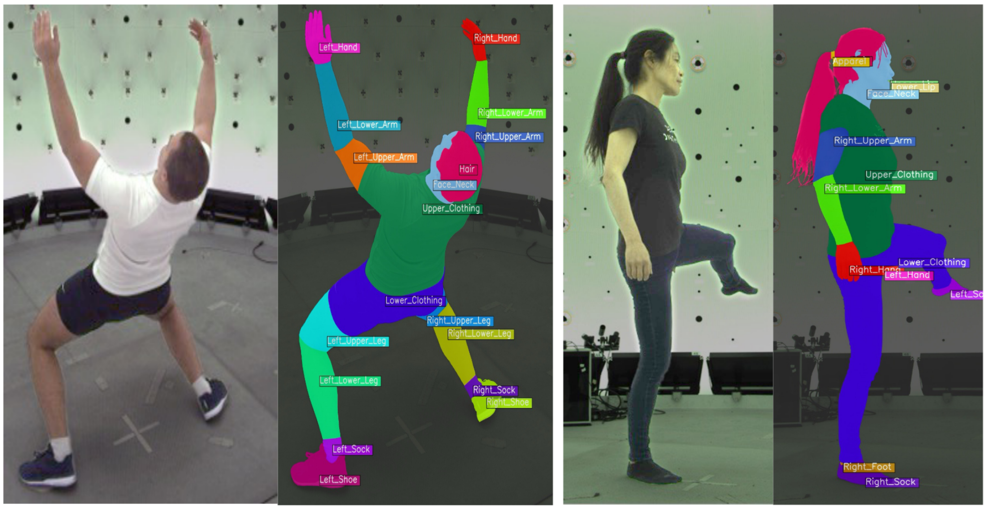


Figure 2.8: Body part segmentation from Sapiens, as presented in [Khi+24].

Despite its advantages, body part segmentation poses significant challenges due to pose variations, occlusions, and complex backgrounds. However, advancements in deep learning architectures have greatly improved segmentation accuracy and robustness, enabling the generation of high-quality segmentation maps suitable for downstream tasks [Khi+24].

2.4 Data Augmentation in Human Pose Estimation

Data augmentation is a crucial strategy for enhancing the robustness and generalization of HPE models. By artificially diversifying training datasets, data augmentation significantly reduces the risk of overfitting and enhances the models' ability to generalize to unseen scenarios [Jia+24; SK19; TN18]. Effective augmentation approaches for human pose estimation can be categorized broadly into geometric transformations, appearance-based adjustments, occlusion simulations, and advanced automated augmentation strategies. This section will summarize augmentation techniques for human pose estimation in accordance with the surveys outlined by [Jia+24; SK19].

2.4.1 Geometric Augmentations

Geometric transformations such as scaling, rotation, translation, cropping, and horizontal flipping are widely employed to simulate viewpoint variations and spatial changes [SSP03; KSH12]. These methods increase invariance to different camera setups and subject orientations, allowing pose estimation models to maintain accuracy under diverse conditions. Additionally, specialized approaches like half-body augmentation–where only upper or lower body parts are selectively zoomed and cropped–have shown empirical improvements by enhancing model sensitivity to fine-grained keypoint details [Jia+24].

2.4.2 Appearance-based Augmentations

Appearance-based augmentations change photometric properties, including brightness, contrast, saturation, and color jitter, to improve robustness under varied lighting and environmental conditions [SK19]. Advanced techniques such as neural style transfer further diversify training data by applying styles from different visual domains, effectively preparing models to handle unseen texture and color variations [GEB16].

2.4.3 Occlusion and Information Dropping

Simulating occlusions via methods such as Cutout [DT17], CutMix [Yun+19], and random erasing introduces robustness to partial visibility, which is commonly encountered in real-world images. By deliberately obscuring body parts or regions within images, these augmentations compel models to infer occluded joints from visible context, thus improving contextual reasoning and accuracy in occluded scenarios [Jia+24].

2.4.4 Synthetic Data Generation

Synthetic data augmentation generates new examples using graphical rendering or recombination techniques. Using computer-generated people or compositing real people onto new backgrounds significantly increases data diversity and fills gaps in real-world data distributions [DMS18]. The realism and variety offered by synthetic augmentations can increase generalization, particularly for rare or difficult-to-capture poses and scenarios [Jia+24].

2.4.5 Automated and Advanced Augmentation

Automated augmentation techniques, including AutoAugment [Cub+19] and RandAugment [Cub+20] dynamically identify optimal augmentation strategies through machine learning methods. Recent adversarial augmentation frameworks and differentiable augmenters such as PoseAug [GZF21] further advances this field by adaptively generating challenging training samples tailored specifically to pose estimation tasks [Jia+24].

3 State-of-the-Art

Human pose estimation has undergone significant progress in the past decade, driven mainly by advancements in deep neural networks and improved computational capabilities. This chapter discusses the state-of-the-art (SOTA) methods in 2D HPE, focusing on CNN-based methods, transformer-based methods, and emerging foundation models. Furthermore, it extensively reviews approaches targeting the challenges posed by occluded and truncated human body parts and examines keypoint dependencies critical for robust estimation in partially observable scenarios.

3.1 CNN-Based Human Pose Estimation Methods

CNN-based methods have been the cornerstone of human pose estimation for the past decade due to their powerful feature extraction capabilities and computational efficiency. One of the foundational works is the Stacked Hourglass Networks proposed by Newell et al. [NYD16]. This architecture uses repeated downsampling and upsampling operations (hourglass modules) to capture multi-scale contextual information effectively. The model iteratively refines predictions by stacking several hourglass modules, integrating global structural information and fine-grained local details. The Stacked Hourglass Network demonstrated superior accuracy on benchmark datasets such as MPII [And+14] and has become a baseline for many subsequent studies.

Building on this idea, Sun et al. [Sun+19] introduced the High-Resolution Network (HRNet), designed explicitly to maintain high-resolution feature maps throughout the entire network structure. HRNet avoids aggressive downsampling, thereby preserving spatial precision and enabling the accurate localization of keypoints. This strategy substantially improved performance on standard benchmarks, such as the COCO Keypoint dataset [Lin+15], without complicated post-processing or multi-stage refinement. Similarly, methods like the Cascaded Pyramid Network (CPN) [Che+18] and SimpleBaseline [XWW18] achieved significant gains by effectively merging multi-scale features. In particular, CPN utilizes a two-stage framework, where an initial global prediction guides a subsequent refinement network that specifically addresses difficult-to-predict keypoints.

3.2 Transformer-Based Approaches and Foundation Models

Transformer architectures have recently demonstrated remarkable potential in human pose estimation by effectively capturing long-range dependencies between keypoints—an advantage over traditional CNNs with limited receptive fields. One of the earliest successful transformer-based methods, TransPose [Yan+21], introduced self-attention mechanisms to model the relationships between joints explicitly. Its attention maps highlight how predictions rely on adjacent and symmetric joints, which is particularly beneficial when handling occlusions. Building on this concept, TokenPose [Li+21] went further by representing each keypoint as a learnable token and then modeling interactions between these tokens. This token-based approach adapts attention to context-specific visual cues, thus enhancing robustness under partial visibility.

Subsequent approaches have adapted the DETR (DEtection TRansformer) [Car+20] framework, initially designed for object detection, to keypoint prediction. By exploiting DETR's end-to-end formulation, these methods jointly perform bounding-box detection and pose estimation, streamlining the overall pipeline. Meanwhile, HRFormer [Yua+21] merges HRNet's high-resolution approach with transformer blocks to retain high-resolution spatial information while incorporating global self-attention. Similarly, hierarchical vision transformers such as Swin Transformer [Liu+21] have been adapted for pose tasks, where window-based attention efficiently captures multi-scale structures crucial for precise keypoint localization.

Another notable development is ViTPose [Xu+22], which employs a ViT backbone pretrained on large-scale image datasets. While ViTPose is not strictly a foundation model, its extensive pretraining provides strong generalization across diverse scenarios. By learning universal visual representations of the human anatomy, ViTPose achieves state-of-the-art accuracy on multiple standard benchmarks. More recently, the field has begun to explore genuine foundation models. Foundation models are large-scale, self-supervised systems that learn broadly applicable representations for various downstream tasks. Sapiens [Khi+24], for instance, was trained on millions of images using self-supervised learning strategies explicitly tailored to human-centric tasks. This extensive pretraining has been shown to result in impressive robustness against partial visibility and occlusions, enabling Sapiens to outperform previous models on challenging benchmarks like Humans-5K. These findings demonstrate the merits of foundation models, which apply extensive data and computing resources to learn highly generalizable representations of human pose.

3.3 Occlusion and Truncation in Human Pose Estimation

Occlusion and truncation remain two of the most challenging problems in human pose estimation. They create partially visible or missing keypoints, which complicate inference. Researchers have addressed these obstacles through a variety of techniques, including both data augmentation and specialized network architectures. A prominent example is Body-Cropping Augmentation (BCA) [PLP20] simulates real-world scenarios by systematically cropping sections of human figures in training images. This strategy compels models to focus on contextual cues to infer missing information, ultimately reducing false-negative detections and improving generalizability.

Position Puzzle Network (PPNet) [PP21] takes a more direct approach to truncation issues, which often arise from inaccurate bounding box estimation, where the bounding box does not fully capture the person. PPNet predicts the likely full-body position and then expands bounding boxes on the fly to accommodate joints that initially fall outside the detected region, thereby boosting keypoint localization in occluded or truncated settings. In parallel, MeTRo (Metric-scale Truncation-Robust heatmaps) [Sár+21] shifts the representation space from pixels to a learned metric scale, allowing for robust joint predictions even when body parts extend beyond the image boundary. Although MeTRo was initially designed for 3D pose estimation, similar principles could be adapted to 2D contexts, mitigating occlusion and truncation effects.

3.4 Keypoint Dependencies and Robustness under Partial Observations

One critical aspect determining model robustness to occlusions and truncations is the interdependency between keypoints. Research has extensively documented how deep neural networks use relational information between keypoints, especially under partial observations. TransPose [Yan+21] and TokenPose [Li+21] demonstrated through attention analysis that models rely heavily on adjacent and symmetrical keypoints for inferring obscured joints. These transformer-based networks progressively refine their attention, initially using broad contextual information and gradually focusing on local anatomical details or symmetrically related joints.

Tang and Wu [TW19] analyzed grouping strategies of keypoints, comparing handcrafted versus statistical (data-driven) approaches. Their study revealed that data-driven grouping strategies consistently outperformed manual ones due to better alignment with learned anatomical relationships and joint dependencies. These findings emphasize the importance of accurately modeling keypoint groups to improve prediction accuracy under challenging visibility conditions.

Moreover, explicit modeling skeletal constraints through graphical structures or losses can further enhance occlusion robustness. For instance, models employing limb-graph consistency losses ensure anatomically plausible predictions even under severe truncations or occlusions by implicitly encoding anatomical priors [Han+24].

This chapter provided a detailed examination of current methodologies in human pose estimation, focusing on the challenges arising from occluded and truncated poses. CNN-based, transformer-based, and foundation models were analyzed, highlighting their respective contributions to state-of-the-art accuracy. Additionally, strategies for explicitly handling partial visibility and keypoint dependencies were explored, emphasizing their role in improving robustness. Despite significant advancements, challenges remain, particularly in pose estimation under truncation, which has received comparatively little research attention.

4 Methodology

Keypoint detection aims to localize specific anatomical landmarks on human figures within an image. Despite significant advancements through deep learning, challenges such as occlusions, truncated body regions, and varied viewpoints still degrade the reliability of standard pose estimation pipelines [Che+18; Li+21; Ma+22]. This chapter presents augmentation strategies selected and designed explicitly for truncated poses and segmentation-based conditioning approaches to address these limitations.

4.1 Dataset Selection and Preprocessing

The proposed approach uses the Microsoft Common Objects in Context (MS COCO) [Lin+15] dataset. MS COCO is a widely recognized and used benchmark dataset for computer vision tasks, including object detection, segmentation, and human pose estimation. The MS COCO dataset contains approximately 328,000 images, with over 2.5 million annotated object instances spanning 91 object categories. The dataset is designed to represent complex real-world scenarios, capturing a wide range of everyday objects in varied environmental conditions. MS COCO employs a multi-step annotation process for accurate and high-quality annotations. This includes instance segmentation, which delineates object boundaries precisely, allowing for more reliable object localization and classification. The annotation process also incorporates keypoint-based human pose labeling, making it particularly suitable for human-centric vision tasks.

4.1.1 Training Dataset

For this study, the 2017 training and validation subsets of MS COCO were selected. These datasets comprise 118,000 images for training and 5,000 for validation. However images without at least one human figure were excluded, since the focus was on human-related tasks. This filtering process resulted in a refined dataset of 25,466 training images and 1,033 validation images.

MS COCO provides both bounding box and keypoint annotations for human instances. The keypoint annotations follow the COCO topology, which consists of 17 keypoints defining the

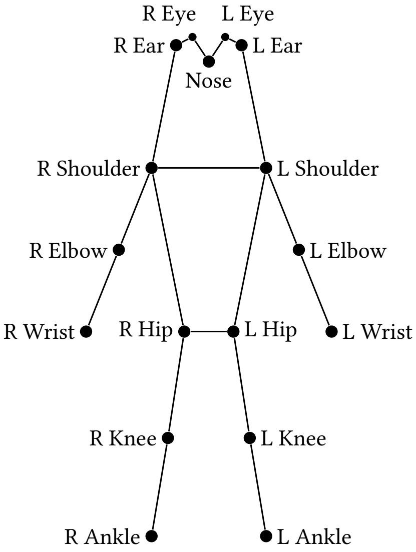


Figure 4.1: Keypoint topology of the MS COCO dataset.

human pose, as illustrated in Figure 4.1. These keypoints represent anatomical landmarks such as facial features, shoulders, elbows, wrists, hips, knees, and ankles.

4.1.2 Test Dataset

A custom test dataset of approximately 500 images was previously collected using a 16-camera multi-camera system. This setup employed 16 *Intel RealSense D415*[1] cameras, strategically arranged and calibrated to function as a 3D person scanner, as shown in Figure 4.2.

In this configuration, the top and bottom cameras were mounted in portrait orientation, while the remaining cameras were positioned in landscape. This arrangement captured natural, multi-perspective views of each subject, ensuring diverse representations. By recording individuals from various angles simultaneously, the dataset contains a wide range of partial-body views, enhancing its applicability for human pose estimation tasks.

For each image, bounding box and keypoint annotations were manually annotated during this study. Each image was labeled with a bounding box and keypoint locations following the COCO keypoint topology, allowing for accurate evaluation of pose estimation models.

Conventional datasets primarily contain full-body poses; however, this dataset naturally includes truncated poses due to the multi-camera setup. Depending on the camera angle, some views capture only upper-body, lower-body, or side-body perspectives, mimicking real-world

[1] https://www.intelrealsense.com/depth-camera-d415/

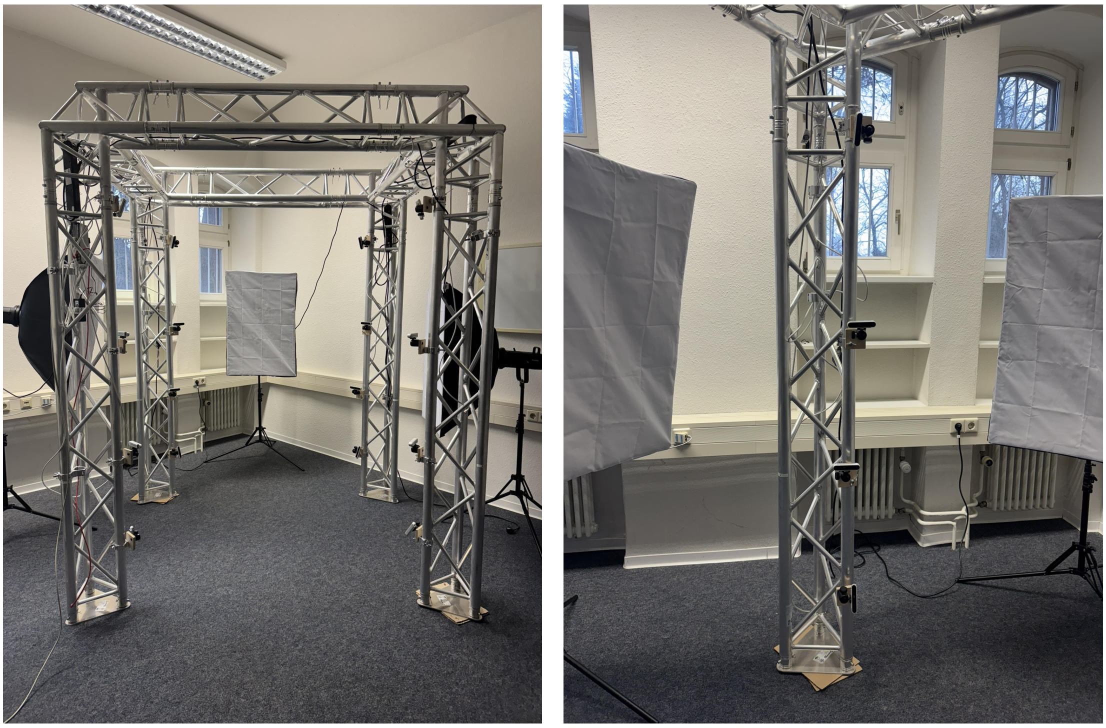


Figure 4.2: Multi-camera setup consisting of 16 RealSense D415 cameras. The top and bottom cameras are mounted vertically, while the remaining cameras are positioned horizontally.

scenarios where individuals may be partially visible at the image boundaries or under non-standard framing conditions, as depicted in Figure 4.3. This dataset offers an ideal testbed for assessing the robustness of models trained to infer poses from incomplete visual information.

4.1.3 Body Part Segmentation

To introduce structural context for each image, body part segmentation maps are generated using the Sapiens-1B model [Khi+24]. These segmentation maps provide a detailed decomposition of the human silhouette into distinct anatomical regions, such as upper and lower arms, legs, torso, and head. Figure 4.4 depicts two images with corresponding segmentation masks. By explicitly encoding spatial priors, these maps supplement standard keypoint-based annotations, aiding pose estimation in cases where portions of the body are missing due to truncation.

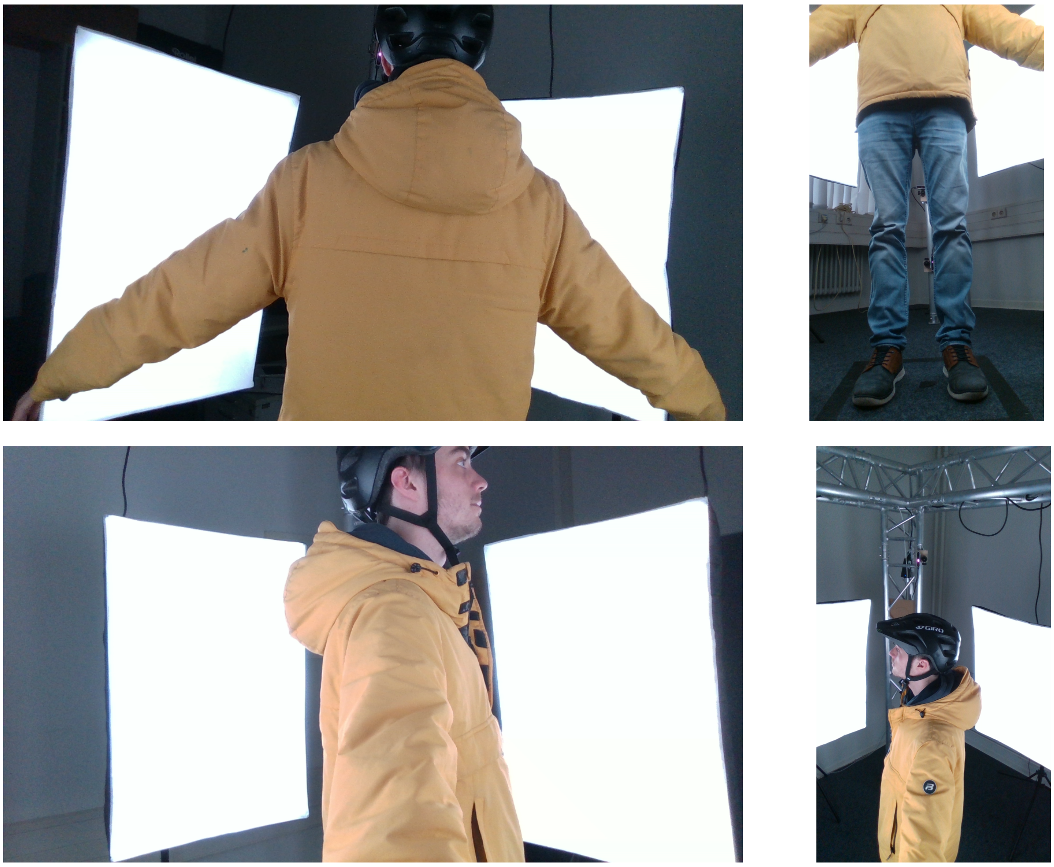


Figure 4.3: Four images captured from different cameras and perspectives using the multi-camera 3D scanning setup.

Sapiens-1B Segmentation Model

The Sapiens-1B model [Khi+24] is a transformer-based segmentation framework designed for pixel-accurate body part segmentation. It employs an MAE pre-training approach to learn robust feature representations, improving generalization across diverse environments.

Encoder The encoder follows a ViT backbone and processes input images by dividing them into non-overlapping patches. These are projected into an embedding space and passed through a multi-layer transformer network, capturing long-range dependencies and spatial relationships. The encoder outputs a structured feature map that serve as input for the segmentation decoder.

Decoder The decoder, known as VitHead, reconstructs high-resolution segmentation masks from the encoded feature representations. It employs deconvolution layers to upsample feature

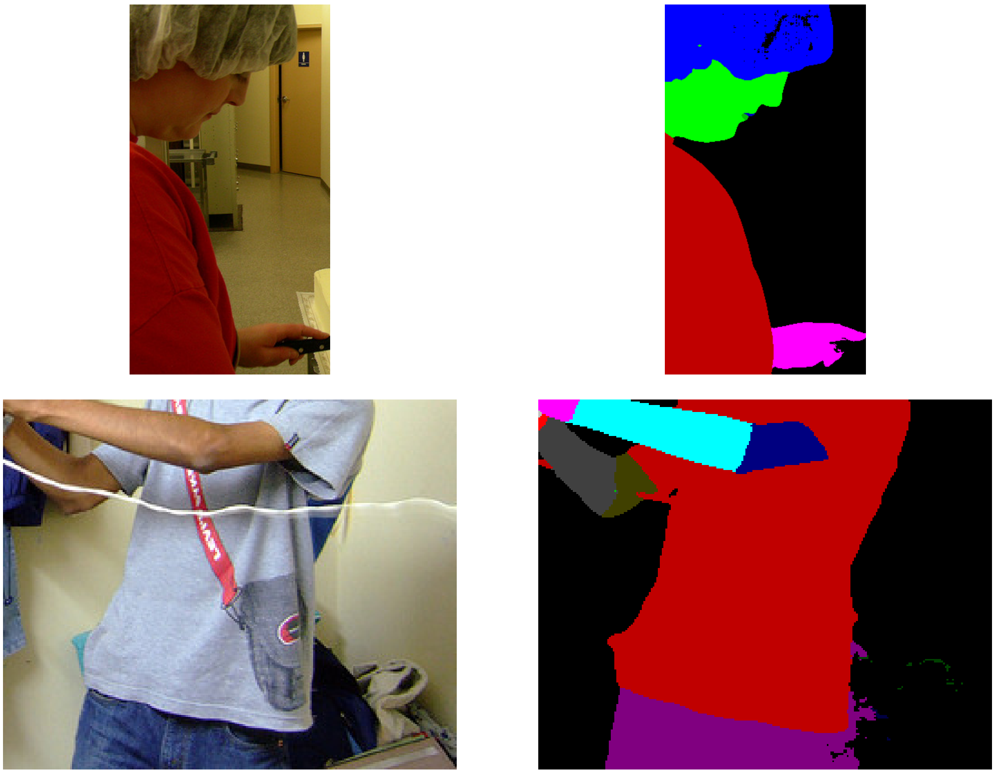


Figure 4.4: Body part segmentation maps (right) generated using the Sapiens-1B model [Khi+24] and aligned with the corresponding augmented RGB images (left) from the MS COCO [Lin+15] dataset.

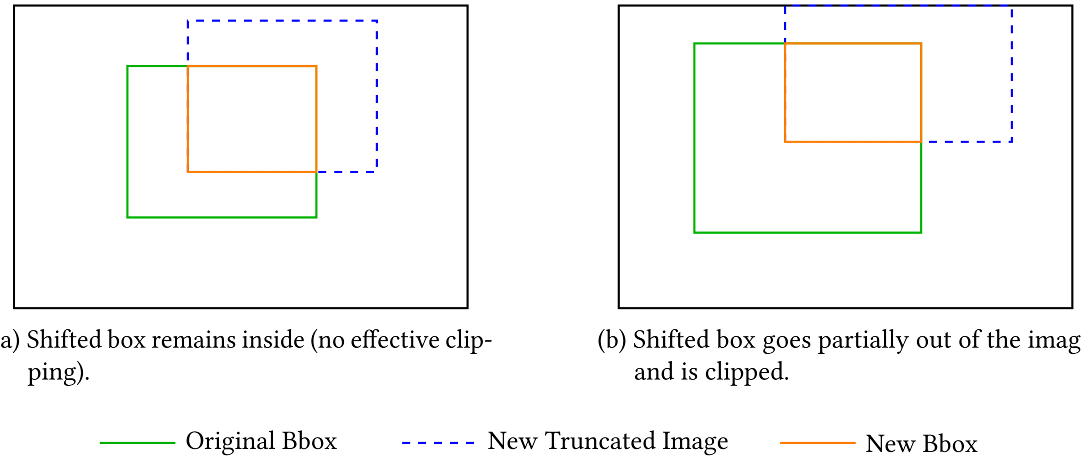
maps, restoring spatial resolution while preserving structural details. Convolution layers refine these features to enhance the accuracy of body part segmentation by capturing finer anatomical details. As a last step, a 1×1 convolution layer maps the refined features to pixel-wise segmentation labels across 28 classes, which include 27 body parts and a background class.

Role in Keypoint Detection

In HPE from partial observations, segmentation maps serve as auxiliary input by providing region-specific cues that compensate for missing body parts. When keypoints are missing due to truncation at the image boundary, the segmentation mask retains information about the spatial extent of visible body regions. This additional structural context helps the model constrain its predictions, improving localization accuracy by enhancing anatomical consistency.

4.2 Synthetic Augmentation

A synthetic augmentation technique was developed to address truncated poses more effectively. This method randomly shifts bounding boxes and clips them to the image boundary, providing a realistic simulation of truncation scenarios that enhances training for partial pose visibility. Figure 4.5 illustrates how the bounding box is shifted and clipped.



(a) Shifted box remains inside (no effective clipping).

(b) Shifted box goes partially out of the image and is clipped.

Figure 4.5: The shifted bounding box (blue, dashed) is clipped to stay within the image boundary. The intersection (orange) is the overlap between the original (green) and the clipped shifted region.

The augmentation is applied with a probability of $p = 0.3$, ensuring controlled exposure to truncated samples. When triggered, the bounding box of a person is extracted in the format (x, y, w, h), where (x, y) are the top-left coordinates, and w, h denote width and height.

Truncation is simulated by shifting the bounding box via offsets Δ_x and Δ_y:

$$\Delta_x \sim U(-0.5w, 0.5w), \quad \Delta_y \sim U(-0.5h, 0.5h) \tag{4.1}$$

where $U(a, b)$ represents a uniform distribution. The shifted bounding box coordinates are:

$$\tilde{x} = x + \Delta_x, \quad \tilde{y} = y + \Delta_y \tag{4.2}$$

while the original width and height remain unchanged:

$$\tilde{w} = w, \quad \tilde{h} = h \tag{4.3}$$

To prevent exceeding the image boundaries, coordinates are clipped:

$$\tilde{x} = \max(0, \min(\tilde{x}, W - \tilde{w})) \tag{4.4}$$

$$\tilde{y} = \max(0, \min(\tilde{y}, H - \tilde{h})) \tag{4.5}$$

Bounding box dimensions are also clipped if necessary:

$$\tilde{w} = \min(\tilde{w}, W - \tilde{x}) \tag{4.6}$$

$$\tilde{h} = \min(\tilde{h}, H - \tilde{y}) \tag{4.7}$$

A final validity check ensures that the truncated region remains large enough (at least 100 pixels in width and height). Any keypoints falling outside the new bounding box are set to zero. If no valid keypoints remain, the augmentation is discarded and the original image is used; otherwise, the augmented image is retained for training.

4.3 Model Architecture

This work employs two primary backbone architectures: HRNet [Sun+19], a CNN-based model, and Sapiens [Khi+24], a Transformer-based approach. Body part segmentation is integrated into these backbones via two distinct methods: spatial attention and ControlNet [ZRA23].

4.3.1 HRNet

HRNet (High-Resolution Network) [Sun+19] is a convolutional architecture designed to maintain high-resolution representations throughout its layers, making it well-suited for tasks that require detailed spatial precision, such as human keypoint localization. In contrast to conventional CNNs that aggressively downsample the spatial dimension, HRNet preserves multiple parallel streams at different resolutions and employs fusion layers to exchange information across these streams. This strategy retains fine-grained features while also incorporating a broader semantic context. The overall HRNet structure is visualized in Figure 4.6, adapted from Sun et al. [Sun+19].

In this study, the HRNet-W48 variant is used. Figure 4.7 visualizes a simplified version of the architecture to demonstrate the different stages and fusion layers. Its design comprises four sequential stages. The first stage includes a single high-resolution branch of bottleneck blocks that produces 64-channel feature maps. The second stage introduces a second parallel

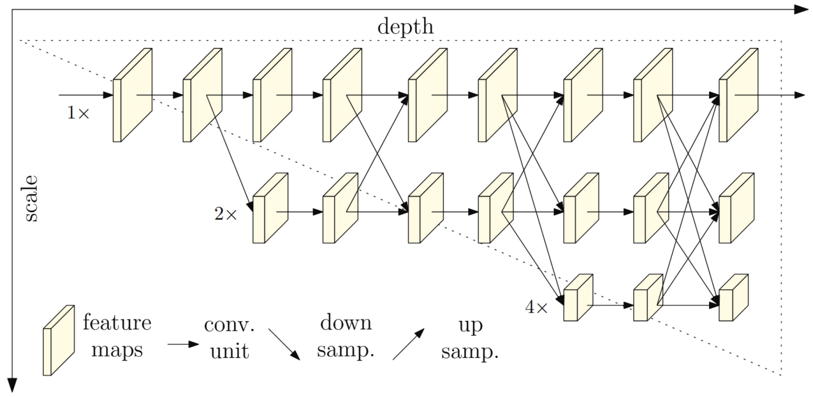


Figure 4.6: HRNet architecture illustrating multi-resolution subnetworks, adapted from [Sun+19].

branch, resulting in two concurrent streams with 48 and 96 feature channels. The third stage expands to three parallel streams carrying 48, 96, and 192 channels. The fourth stage adds a fourth branch, culminating in streams of 48, 96, 192, and 384 channels. Figure 4.7 depicts a simplified schematic of these stages.

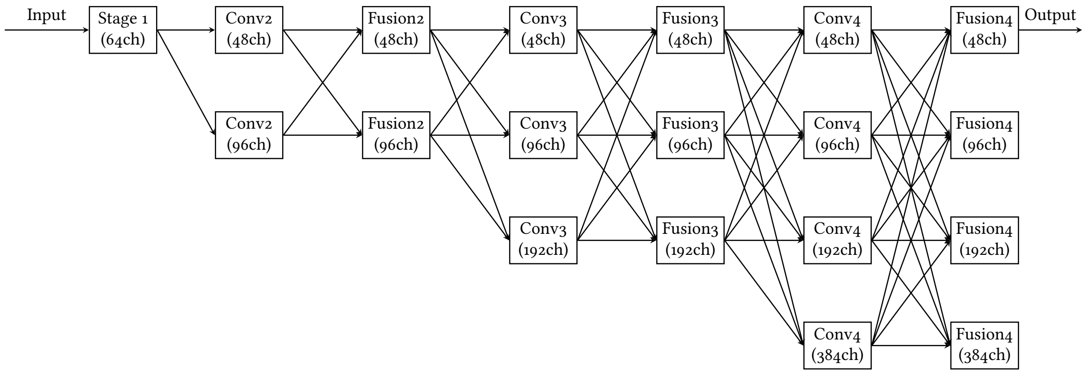


Figure 4.7: Simplified HRNet architecture visualization showing the multi-resolution parallel subnetworks, feature channel counts and feature fusion.

Each stage incorporates cross-resolution fusion layers that continuously exchange spatially-detailed features with broader semantic information. After the final stage, HRNet applies a *HeatmapHead* to the highest-resolution (48-channel) feature maps, producing 17 heatmaps for keypoint prediction. Owing to HRNet's inherently high-resolution design, no additional upsampling or deconvolution layers are required. Optimization is performed via mean squared error (MSE) loss on the predicted heatmaps. By consistently retaining high-resolution features, HRNet facilitates precise human pose localization.

4.3.2 Sapiens 0.3B

Sapiens [Khi+24] is a Transformer-based model explicitly fine-tuned for human pose estimation. The *Sapiens 0.3B* variant, containing approximately 300 million parameters, is employed in this work. Figure 4.8 illustrates the Sapiens architecture (left) and the transformer encoder (right) that is used. The core of this architecture is a ViT backbone enhanced by MAE pretraining.

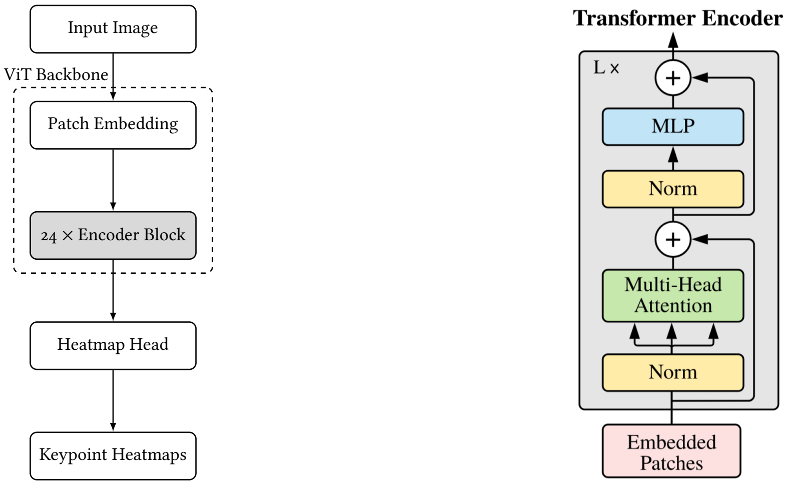


Figure 4.8: Simplified Sapiens Pose Architecture (left) und used Transformer Encoder Architecture (right) [Vas+23; Dos+21].

At the input stage, images are divided into 16 × 16 non-overlapping patches, which are than flattened and transformed into 1,024-dimensional token embeddings. In contrast to conventional ViT designs, Sapiens omits the class token, instead generating dense feature maps that are directly applicable to keypoint-level predictions. A total of 24 Transformer encoder layers compose the primary backbone, with layer normalization applied immediately before the subsequent prediction modules.

Following the Transformer backbone, the model introduces a dedicated *HeatmapHead*. This head progressively restores spatial resolution using two transposed convolution operations, each with a 4 × 4 kernel size, resulting in an overall fourfold upsampling. Intermediate convolutions further refine these upsampled feature maps, culminating in a series of heatmaps, one per human keypoint.

Both models are initialized with pre-trained weights. HRNet utilizes the publicly available "td-hm_hrnet-w48_8xb32-210e_coco-384x288" checkpoint from MMPose [Con20], while Sapiens relies on weights provided by its original authors [Khi+24].

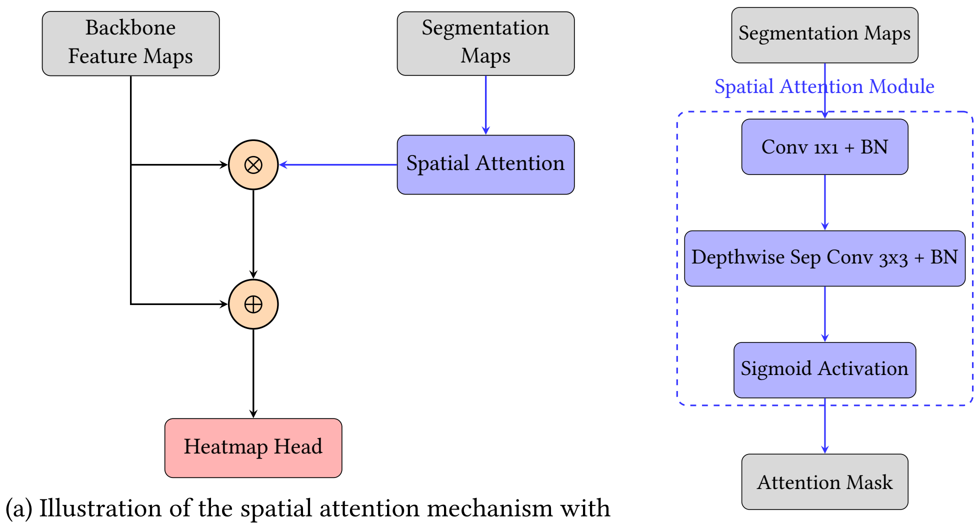
4.4 Segmentation-Based Methods

Two approaches for incorporating body part segmentation into the HPE pipeline are explored: spatial attention, a backbone-independent method that enhances relevant spatial features, and ControlNet, which integrates segmentation information directly within the backbone architecture.

4.4.1 Method 1: Spatial Attention

The first method applies spatial attention to enhance regions of interest while suppressing irrelevant areas selectively. This approach aims to refine keypoint detection, particularly in cases of truncated or occluded poses, by directing the model's focus toward visible body parts and away from background clutter.

Segmentation-Driven Attention Computation



(a) Illustration of the spatial attention mechanism with backbone feature maps and segmentation maps.

(b) Spatial attention architecture.

Figure 4.9: Comparison between the spatial attention mechanism (left) and its architectural details (right). The spatial attention module and its components are highlighted in blue.

Segmentation maps obtained from Sapiens Segmentation provide body part information indicating which regions are likely to contain visible limbs or torsos. As illustrated in Figure 4.9, these maps are passed through a lightweight set of convolutions to produce an attention mask $\mathbf{A}$. A 1×1 convolution with batch normalization matches the channel dimensions of the backbone features and a subsequent depthwise separable convolution refine local spatial responses while reducing parameter overhead. A sigmoid activation assigns attention values between 0 and 1, leading to partial emphasis on important anatomical areas.

The refined mask highlights visible body parts at each spatial location and suppresses occluded or background regions. The resulting mask modulates the backbone feature maps **F** through elementwise multiplication, scaling the contribution of each spatial position. A residual connection adds **F** back to the modulated features, helping preserve the original representation while selectively focusing on the most informative regions. Figure 4.9a illustrates the applied architecture. Formally,

$$\mathbf{F}' = \mathbf{F} + \mathbf{F} \otimes \sigma(\mathbf{A}) \tag{4.8}$$

where $\sigma(\cdot)$ denotes the sigmoid function, **F** the backbone features, and **F**′ the attention-weighted output. This formulation mitigates over-dependence on the segmentation signal and stabilizes learning by retaining direct access to unmodified features.

Integration into HRNet and Sapiens. The proposed mechanism is integrated after feature extraction, enabling seamless incorporation into various top-down pose estimation pipelines. The HRNet and Sapiens Pose models leverage segmentation-driven attention to enhance keypoint prediction. In HRNet, the attention mask is derived from the Sapiens Segmentation output and integrated into the high-resolution feature stream obtained from the HRNet backbone. This refined feature stream is then processed by the HeatmapHead, improving keypoint localization by emphasizing relevant anatomical structures while suppressing background noise. Similarly, the Sapiens Pose model utilizes segmentation maps from the independent Sapiens Segmentation network to guide spatial attention. This mechanism directs the network toward discriminative regions, enhancing keypoint accuracy. It is important to note that the segmentation model operates independently of the pose estimation pipeline, ensuring modularity while effectively leveraging segmentation-based spatial priors.

Implementation Considerations Integrating segmentation attention requires minimal architectural modifications and introduces only a small number of additional parameters. However, generating segmentation maps as a preprocessing step adds computational overhead, increasing inference time compared to pipelines that rely solely on RGB input. Despite this, since feature modulation occurs at the final stage of feature extraction, this method remains compatible with various pose estimation pipelines while improving robustness to occlusions and challenging poses.

Benefits and Limitations

Segmentation-driven attention adaptively focuses the network on relevant parts of the human body, which proves valuable when certain limbs are obstructed or truncated. This localized emphasis serves to mitigate the loss of information due to truncation by introducing information about the presence or absence of body part parts. However, it should be noted that the effectiveness of the attention mask is highly dependent on the accuracy of the segmentation process [Yan+21]. Incorrect or imprecise segmentation maps can propagate misleading cues, degrading the overall pose estimation. Furthermore, exclusively local attention may fail to capture global dependencies, such as severely out-of-frame limbs.

4.4.2 Method 2: ControlNet

ControlNet was introduced in [ZRA23] to enable generative models to incorporate auxiliary conditioning signals, such as segmentation masks, for more structured outputs. In this study, the ControlNet framework is extended beyond its conventional applications and adapted to human pose estimation, specifically within the feature extraction process. This approach introduces segmentation-based conditioning to enhance the spatial understanding of articulated body structures, improving keypoint localization even in challenging scenarios such as occlusions or truncations.

Traditional pose estimation models rely primarily on raw image features, which often struggle to resolve occluded or overlapping limb position ambiguities. By integrating ControlNet, segmentation cues act as additional control signals that explicitly highlight anatomical boundaries. This facilitates the disambiguation of complex poses, providing a structured mechanism to inject spatial priors into the feature extraction pipeline. The goal is to enhance both the semantic richness and geometric consistency of the learned pose representations while preserving the generalization capacity of the backbone network.

ControlNet Architecture

ControlNet follows a parallel-branch design, introducing trainable control pathway alongside a frozen pre-trained backbone. As shown in Figure 4.10, a new ControlNet block is added parallel to each existing network block. This new branch processes control signals, such as segmentation-based features, while the original network block remains unaltered. The control pathway consists of zero-convolution layers, which allow the segmentation features to be projected into a latent space compatible with the backbone's feature maps. The processed features are then merged back into the main network using residual addition. This architecture enables the model to incorporate external spatial cues without altering the core structure of the pre-trained backbone.

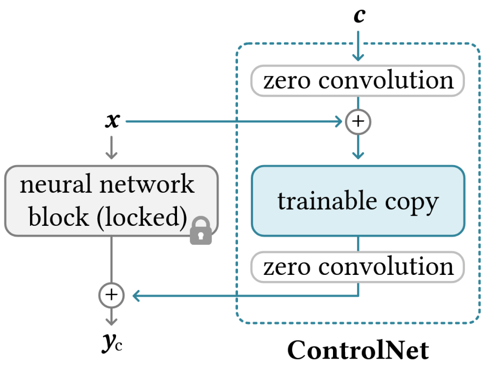


Figure 4.10: Illustration of ControlNet from [ZRA23]. A parallel, trainable copy of the original network block processes segmentation-based control signals, while the original block (frozen) remains untouched.

Integration into HRNet and Sapiens

To integrate ControlNet into existing pose estimation architectures, body part segmentation maps from Sapiens Segmentation are introduced at multiple levels within the feature extraction backbone. In HRNet, where a block corresponds to an entire stage, ControlNet modules are added in Stages 2–4. In the Sapiens Pose model, each encoder block is augmented with a parallel ControlNet branch. The additional ControlNet branch follows the standard design, where segmentation features are processed separately and fused back into the backbone through residual connections.

This integration allows segmentation-driven conditioning to refine the raw feature representations at multiple feature extraction stages. By reinforcing anatomical boundaries and spatial cues, the approach enhances the network's ability to localize keypoints accurately, particularly in cases of occlusion, truncation, or complex pose variations.

Only the newly added ControlNet branches are updated during training, while the core feature extraction backbone remains frozen. This selective updating strategy, inspired by prior work [Zha+24], helps retain the pose-estimation capabilities of the original network while preventing catastrophic forgetting. At the same time, it enables the effective incorporation of segmentation cues to refine keypoint localization.

Benefits and Limitations

The integration of ControlNet into HPE provides several advantages. By incorporating segmentation-based conditioning, the model gains a more explicit representation of articulated body structures, particularly in cases of occlusion, truncation, or highly overlapping limbs. Furthermore, ControlNet's modular design allows for seamless integration with existing architectures while requiring minimal modifications to the backbone.

However, this approach introduces additional computational overhead and memory usage due to the parallel processing branches. To mitigate this, Sapiens 0.3B is selected as a lightweight segmentation network. Another limitation is the reliance on external segmentation maps, which may propagate errors into the pose estimation pipeline if the segmentation output is inaccurate. Additionally, while ControlNet enhances local feature refinement, it does not inherently capture long-range dependencies between body parts, which integration into transformer-based backbones might solve. Addressing such limitations may require supplementary mechanisms.

4.5 Training Strategy

The training strategy is designed to enhance model robustness and generalization by incorporating targeted data augmentation, carefully selected loss functions, and optimization techniques specifically tailored to each network architecture.

4.5.1 Data Augmentation

To improve resilience against real-world challenges such as occlusions, variations in viewpoint, and imaging noise, a structured augmentation pipeline is employed on top of the training dataset prepocessing. This pipeline consists of three primary types of transformations: bounding box modifications, half-body occlusion simulations, and image-level manipulations. All augmentations are implemented using the MMPose framework [Con20].

Bounding Box Transformations. Bounding box augmentations are applied to improve robustness in the detection of inaccuracies and variations in subject positioning. The following transformations are performed:

- **Shifting:** With a probability of 30%, the bounding box is randomly translated within ±16% of its original scale along both the x- and y-axes. This helps simulate minor misalignments and off-center detections.

- **Scaling:** Each bounding box is resized by a random factor uniformly sampled between 50% and 150% of its original size to account for variations in subject distance and frame coverage.
- **Rotation:** To introduce pose variations caused by different camera angles, bounding boxes are rotated by a random angle up to ±80° with a 60% probability.

Half-Body Transformations. To mimic severe occlusion and truncation scenarios, half-body transformations are applied to a subset of training samples. Specifically, in 30% of the cases where at least eight keypoints are detected and at least two belong to a specific body region (upper or lower body), the bounding box is recalculated to enclose only that region. A padding scale of 1.5 is added around the cropped area to maintain context. This transformation forces the model to infer complete poses from partial visual information.

Image-Based Transformations. In addition to spatial modifications, image-level perturbations are introduced to simulate diverse real-world imaging conditions. These include:

- **Horizontal Flipping:** Applied randomly to enhance pose invariance to lateral reflections.
- **Blurring:** Gaussian and median blur are each applied with a 5% probability to simulate out-of-focus or motion-blurred images.
- **Coarse Dropout:** To enhance robustness against missing visual information, 40% of the images undergo coarse dropout, where rectangular regions are randomly masked. This compels the model to rely on global context rather than individual keypoints.

All image transformations are implemented using MMPose framework [Con20] and extended to segmentation masks, ensuring consistency between images and any associated segmentation masks.

By randomly applying these augmentations throughout training, the model is exposed to diverse conditions, improving its ability to generalize to challenging real-world scenarios, such as extreme occlusions, varying scales, and diverse illumination conditions.

4.5.2 Loss Function and Optimization

Pose estimation is framed as a heatmap regression task, where a Gaussian heatmap is predicted for each joint location. The loss function is defined as the mean squared error (MSE) between the predicted and ground truth heatmaps:

$$\mathcal{L}_{\text{heatmap}} = \frac{1}{N} \sum_{i=1}^{N} \left\| \hat{H}_i - H_i \right\|^2 \tag{4.9}$$

where $\hat{H}_i$ represents the predicted heatmap for the i-th joint, H_i is the corresponding ground truth heatmap, and N denotes the total number of keypoints.

Optimization Strategy

Both HRNet and Sapiens are optimized using adaptive gradient-based methods with warm-up and cosine annealing [LH17] learning rate schedules to ensure stable convergence during fine-tuning. HRNet employs the Adam optimizer [KB17], while Sapiens utilizes AdamW [LH19] with additional weight decay adjustments.

Training begins with a warm-up phase over the first 500 iterations, where the learning rate is gradually increased from 0.001 to the base value of 1×10^{-4}, ensuring stable optimization. A cosine annealing schedule is applied, progressively reducing the learning rate to a minimum of 1×10^{-6}. For HRNet, the decay occurs over 50 epochs, whereas for Sapiens, it extends over 30 epochs. Unlike traditional multi-step decay, which introduces abrupt learning rate reductions at predefined epochs, cosine annealing allows for a smooth and progressive decrease, preventing sudden optimization shifts that may destabilize fine-tuning. This approach enhances convergence stability by maintaining higher learning rates in the early stages for broader exploration while enabling precise fine-tuning in later epochs, making it particularly suitable for high-resolution networks and transformer-based models.

Sapiens fine-tuning follows the model's original weight decay and layer-wise adaptation strategy, incorporating a weight decay of 0.05 along with a layer-wise learning rate decay strategy. A decay rate of 0.85 per layer is applied across 24 transformer layers, ensuring that lower layers retain higher learning rates while deeper layers are gradually regularized. To prevent unnecessary regularization, decay multipliers for bias terms, positional embeddings, and relative position bias tables are set to zero, ensuring stable adaptation of model parameters.

To further stabilize training, gradient clipping is enforced with a maximum norm of 1.0 under the L_2 norm, preventing gradient explosion and ensuring controlled weight updates. After each training epoch, model performance is evaluated, and the checkpoint with the highest average precision is retained for final evaluation.

This optimization strategy, integrating structured learning rate adjustments, adaptive weight decay, and stabilization techniques, enhances the model's robustness and efficiency in real-world pose estimation scenarios.

4.6 Performance Metrics

To evaluate the effectiveness of the proposed methods, three key metrics are employed: Average Precision (AP), Average Recall (AR), and Percentage of Correct Keypoints (PCK). These metrics quantify different aspects of keypoint detection performance, including precision, recall, and localization accuracy.

4.6.1 Object Keypoint Similarity

The Object Keypoint Similarity (OKS) metric [Lin+15; RP17] is analogous to Intersection over Union (IoU) for evaluating keypoint localization. It quantifies the similarity between predicted keypoints and ground truth annotations while accounting for object scale and keypoint-specific localization tolerances. OKS is computed as follows:

$$\text{OKS} = \frac{\sum_i e^{-\frac{d_i^2}{2s^2k_i^2}} \cdot \delta(v_i > 0)}{\sum_i \delta(v_i > 0)} \tag{4.10}$$

where:

- d_i is the Euclidean distance between the predicted and ground truth locations for keypoint i.
- s is the object scale, defined as the square root of the bounding box area.
- k_i is a per-keypoint constant that normalizes localization error based on anatomical variability (e.g., head keypoints have lower tolerance than ankle keypoints).
- v_i is the visibility flag for keypoint i, where $v_i = 0$ means the keypoint is unannotated and ignored in evaluation.
- $\delta(v_i > 0)$ ensures that only annotated keypoints contribute to the similarity score.

The OKS value ranges from 0 (no similarity) to 1 (perfect alignment). This formulation allows for a fair comparison of keypoint localization across different object scales and body parts.

4.6.2 Average Precision

Average Precision (AP) [Lin+15] is a widely adopted metric for HPE that evaluates the accuracy of predicted keypoints by computing precision-recall curves at multiple OKS thresholds. The key AP variants in this study include:

- **AP@0.5:** Measures detection quality under a lenient criterion, requiring at least OKS ≥ 0.50 for a keypoint to be considered correct.

- **AP@0.75:** Employs a stricter threshold (OKS ≥ 0.75), emphasizing a model's ability to localize keypoints with high precision.

- **AP@[0.5:0.95]:** Averages AP across ten OKS thresholds (0.50, 0.55, 0.60, …, 0.95), providing a comprehensive measure of localization performance.

The AP metric is calculated by computing the area under the precision-recall curve for each OKS threshold. Precision is proportion of correctly predicted keypoints relative to all predictions, while recall represents the proportion of ground truth keypoints successfully detected. A prediction is classified as a true positive if its OKS exceeds the specified threshold.

Higher AP values indicate that the model produces more accurate keypoint detections with fewer false positives. Models with high AP@0.75 or AP@[0.5:0.95] scores excel in fine-grained localization, whereas AP@0.5 primarily reflects coarse detection ability.

4.6.3 Average Recall

Average Recall (AR) [Lin+15] evaluates the model's ability to detect ground truth keypoints across different OKS thresholds. In this work, AR is reported for the following thresholds:

- **AR@0.5:** Measures recall when keypoints are considered correctly detected if OKS ≥ 0.50.

- **AR@0.75:** Evaluates recall under a stricter threshold (OKS ≥ 0.75), assessing the model's ability to recover keypoints with high localization accuracy.

- **AR@[0.5:0.95]:** Averages AR across ten OKS thresholds (0.50, 0.55, 0.60, . . . , 0.95), offering a balanced measure of detection comprehensiveness.

AR is computed by aggregating recall over the specified OKS thresholds. A keypoint is considered correctly detected if at least one predicted keypoint meets the OKS threshold. Unlike AP, AR prioritizes the detection of all keypoints, placing less emphasis on precision.

Higher AR values indicate that the model successfully detects a larger proportion of ground truth keypoints across different OKS thresholds. When coupled with high AP, an elevated AR suggests that the model is both precise and comprehensive in keypoint localization.

4.6.4 Percentage of Correct Keypoints

Percentage of Correct Keypoints (PCK), introduced by Yang et al. [YR13], quantifies keypoint localization accuracy based on a normalized distance criterion rather than OKS. Unlike AP and AR, which rely on OKS thresholds, PCK measures whether keypoints are within a certain proportion of a reference distance (e.g., person bounding box or head bounding box).

A keypoint is considered correct if its Euclidean distance from the ground truth location falls within a threshold proportional to the selected bounding box size:

$$\sqrt{(x_p - x_g)^2 + (y_p - y_g)^2} \leq \alpha \times \text{bbox\_size} \tag{4.11}$$

where:

- (x_p, y_p) is the predicted keypoint location.
- (x_g, y_g) is the ground truth keypoint location.
- α is a scaling factor (e.g., 0.05 for a 5% tolerance as used in this work).
- bbox_size is the maximum of the bounding box width or height.

PCK is then computed as the percentage of keypoints meeting this criterion over the entire dataset.

PCK provides a scale-invariant assessment of localization accuracy, which is particularly useful when the quality of the bounding box varies. Smaller values of α enforce stricter localization criteria, while larger values allow more flexibility. PCK is particularly interesting for evaluating keypoints in challenging scenarios, such as occlusions or extreme poses.

4.7 Implementation Details

The development process utilizes the MMPose [Con20] and PyTorch framework [Pas+17] to provide a flexible and modular environment for integrating different pose estimation architectures. These frameworks allow for seamless incorporation of various backbone networks, including HRNet and Sapiens, along with custom Spatial Attention and ControlNet modules. The implementation's modular nature allows for independent activation or deactivation of segmentation attention and ControlNet functionalities, enabling controlled ablation studies to assess their contributions to model performance.

All experiments are conducted on an NVIDIA RTX 4090 GPU paired with an AMD Ryzen 9 7950X CPU and 64 GB of RAM. While HRNet operates efficiently with the full augmentation pipeline, fine-tuning large-scale transformer-based architectures, such as Sapiens-2B with approximately two billion parameters, introduces a substantial computational overhead where the training does not fit onto a single GPU. Consequently, Sapiens 0.3B with approximately 300 million parameters has been selected for its smaller size and more manageable parameter count.

The methodology integrates robust model architectures, segmentation-driven spatial attention, and ControlNet methods supported by tailored augmentation strategies and comprehensive training protocols. The subsequent chapter presents comparative results validating these methods, analyzing their effectiveness in handling human poses estimation in cases of partial observations.

5 Results

Building upon the methodology described in the previous chapter, this chapter thoroughly evaluates human pose estimation on the 3D Scanner Dataset, where body parts frequently fall outside the camera's field of view. These truncated views introduce missing information that can degrade performance for other visible joints, creating a realistic and challenging testbed for incomplete pose detection.

Four configurations are compared for HRNet [Sun+19] and Sapiens 0.3B [Khi+24]: Baseline (pretrained, no additional training), Fine-tuned (trained exclusively with a custom augmented dataset that includes artificially truncated images), Segmentation Attention (Seg. Attn.), and ControlNet. The primary metrics assessed are Average Precision, Average Recall, and Percentage of Correct Keypoints. Since the dataset commonly presents body parts partially cut off at image boundaries, these evaluations reveal how effectively each method handles incomplete data while aiming for accurate estimates of the visible keypoints.

5.1 Overall Performance Comparison

Two models, HRNet and Sapiens-0.3B, are examined to systematically evaluate the impact of these different conditioning strategies and training schemes under truncated conditions. They are compared in terms of AP, AR, and PCK. Table 5.1 shows how Baseline, Fine-tuned, Seg. Attn., and ControlNet influence performance for each model, with bolded values indicating each metric's best score.

5.1.1 HRNet Performance Analysis

HRNet exhibits strong AP and AR across all examined configurations, indicating robust keypoint detection despite missing image regions. The Baseline model attains AP@0.5 = 0.970 and AR@0.5 = 0.977, showing that even without specialized training, it handles partial views effectively. Its AP@[0.5:0.95] of 0.911 further underscores consistent precision under varying threshold constraints.

Fine-tuning with truncation-oriented augmentation leads to a notable AP@0.75 increase (from 0.937 to 0.948), indicating that exposing the network to augmented images during

Model	Method	AP			AR			PCK@0.05
		AP@0.5	AP@0.75	AP@[0.5:0.95]	AR@0.5	AR@0.75	AR@[0.5:0.95]	
HRNet	Baseline	0.970	0.937	0.911	0.977	0.949	0.925	0.907
	Fine-tuned	**0.979**	**0.948**	0.918	**0.980**	0.951	0.931	0.907
	Seg. Attn.	**0.979**	0.937	0.911	**0.980**	0.949	0.925	0.907
	ControlNet	0.978	0.936	0.916	**0.980**	0.949	0.930	0.914
Sapiens-0.3B	Baseline	0.950	0.905	0.881	0.975	0.949	0.933	0.930
	Fine-tuned	0.959	0.927	0.914	0.965	0.937	0.925	0.924
	Seg. Attn.	0.960	0.938	0.915	0.967	0.941	0.926	0.916
	ControlNet	0.969	**0.948**	**0.928**	0.973	**0.955**	**0.937**	**0.934**

Table 5.1: Comparison of Baseline (pretrained), Fine-tuned, Segmentation-Guided Attention, and ControlNet for HRNet and Sapiens-0.3B on the 3D Scanner Dataset. Bold values represent the best result in each column.

training helps it navigate diminished visual cues more effectively. AR remains essentially unchanged, indicating that the model's ability to detect keypoints has not increased but has become more precise where detection is successful.

Segmentation Attention (Seg. Attn.) does not substantially alter HRNet's performance, implying that coarse segmentation cues add little when the network is already capable of localizing keypoints in truncated settings. ControlNet yields a moderate boost in PCK@0.05 (from 0.907 to 0.914), suggesting slight benefits for fine-grained localization via multi-layer segmentation conditioning, though these gains fall behind those from fine-tuning.

5.1.2 Sapiens-0.3B Performance Analysis

Sapiens-0.3B, although somewhat lower in AP than HRNet, achieves a higher PCK@0.05. Its Baseline configuration reaches AP@0.5 = 0.950, AR@0.5 = 0.975, and PCK@0.05 = 0.930, reflecting strong localization of visible joints. However, its AP@[0.5:0.95] of 0.881 suggests that Sapiens-0.3B is more sensitive to truncated limbs when the evaluation criteria become stricter.

Fine-tuning raises AP@[0.5:0.95] from 0.881 to 0.914 but slightly reduces AR, implying a precision-recall trade-off in detecting truncated keypoints. Seg. Attn. moderately increases precision without eroding recall, pushing AP@[0.5:0.95] to 0.915. The results indicate that Sapiens-0.3B makes better use of segmentation cues for missing limb regions than HRNet, likely due to its transformer-based global attention, but Seg. Attn. does not lead to any significant gains over fine-tuning.

ControlNet emerges as the most effective method for Sapiens-0.3B, lifting AP@[0.5:0.95] to 0.928 and AR@0.75 to 0.955 while raising PCK@0.05 to 0.934. Although segmentation-guided conditioning does not substantially outperform fine-tuning, these results underscore that using ControlNet for segmentation-guided conditioning still provides notable benefits to HPE, producing higher precision and recall under stricter thresholds.

5.1.3 Key Observations

HRNet remains robust in truncated scenarios, benefiting primarily from targeted fine-tuning, while Seg. Attn. provides minimal additional improvement. ControlNet offers slight boosts in PCK but does not significantly enhance overall performance. Sapiens-0.3B, though slightly weaker in raw AP, achieves higher PCK for visible joints. It gains moderate precision improvements from Seg. Attn. and reaches its best overall performance with ControlNet.

These findings confirm that (1) truncation-specific image augmentation is essential for improving pose estimation in missing-limb scenarios, (2) model architectures differ in how effectively they integrate segmentation-based conditioning, and (3) while no substantial performance gains were observed, the integration of ControlNet into Sapiens-0.3B still led to stronger performance, indicating that segmentation guidance can enhance HPE model performance.

5.2 Analysis by Body Region

A region-specific breakdown clarifies whether each method resolves truncation consistently across different body parts. Tables 5.2, 5.3, and 5.4 detail PCK for facial landmarks, upper-body keypoints, and lower-body keypoints, respectively.

5.2.1 Facial Keypoints

Table 5.2 shows the PCK for the measured facial landmarks – nose, eyes, and ears.

Model	**Method**	**PCK@0.05**		
		Nose	**Eyes**	**Ears**
HRNet	Baseline	**0.995**	0.984	0.990
	Fine-tuned	**0.995**	0.988	**0.993**
	Seg. Attn.	**0.995**	0.984	0.990
	ControlNet	**0.995**	0.988	0.987
Sapiens-0.3B	Baseline	0.990	**0.996**	**0.994**
	Fine-tuned	0.990	0.987	0.990
	Seg. Attn.	0.990	0.992	0.987
	ControlNet	0.990	0.992	**0.994**

Table 5.2: Percentage of Correct Keypoints (PCK) for facial landmarks: nose, eyes, and ears.

HRNet achieves near-perfect nose estimates (0.995) regardless of approach, and segmentation-based methods do not appreciably elevate performance. Fine-tuning yields small gains for eye and ear detection, while ControlNet mildly reduces ear accuracy. Sapiens-0.3B similarly shows limited variation for facial landmarks, with only minor changes in ear PCK between configurations. Because the face region in the 3D Scanner Dataset is typically either fully visible or missing at the top boundary, segmentation conditioning delivers minimal additional benefit.

5.2.2 Upper-Body Keypoints

Table 5.3 focuses on shoulders, elbows, and wrists.

Model	**Method**	**PCK@0.05**		
		Shoulders	**Elbows**	**Wrists**
HRNet	Baseline	0.933	0.918	0.967
	Fine-tuned	0.926	0.909	0.960
	Seg. Attn.	0.931	0.920	0.967
	ControlNet	0.935	0.909	0.965
Sapiens-0.3B	Baseline	0.937	**0.950**	0.974
	Fine-tuned	**0.940**	0.941	0.966
	Seg. Attn.	0.928	0.946	0.979
	ControlNet	0.928	0.942	**0.981**

Table 5.3: Percentage of Correct Keypoints (PCK) for upper-body landmarks: shoulders, elbows, and wrists.

HRNet remains stable in the upper body, with shoulders around 0.93 PCK and wrists exceeding 0.96. Gains from segmentation approaches are modest but consistent. Sapiens-0.3B displays more visible improvements. Wrists, which may be partially cropped or poorly visible when arms are extended, benefit from ControlNet's segmentation-based cues, reaching a PCK of 0.981. The transformer-based architecture appears to leverage multi-layer conditioning more effectively here.

5.2.3 Lower-Body Keypoints

Table 5.4 reports the PCK for hips, knees, and ankles.

HRNet struggles with hip localization, achieving only a PCK of 0.670 under ControlNet, whereas knees and ankles consistently score above 0.88. Sapiens-0.3B fares better, reaching 0.766 for hips under ControlNet, yet still exhibits a substantial drop in hip accuracy compared to knees and ankles. These trends indicate persistent ambiguity in hip localization.

Model	Method	PCK@0.05		
		Hips	**Knees**	**Ankles**
HRNet	Baseline	0.653	0.872	0.894
	Fine-tuned	0.628	0.861	0.883
	Seg. Attn.	0.657	0.872	0.894
	ControlNet	0.670	0.883	**0.918**
Sapiens-0.3B	Baseline	0.708	**0.939**	0.914
	Fine-tuned	0.750	0.903	0.883
	Seg. Attn.	0.675	0.894	0.891
	ControlNet	**0.766**	0.930	0.914

Table 5.4: Percentage of Correct Keypoints (PCK) for lower-body landmarks: hips, knees, and ankles.

To further analyze this, Table 5.5 reports the AP and AR for lower-body keypoints individually. Unlike PCK, which measures the proportion of keypoints within a fixed threshold, AP and AR use OKS calculations that dynamically adjust the threshold based on keypoint uncertainty and scale. This distinction is critical, as whole-pose AP considers all keypoints collectively, making direct comparisons between whole-body and individual joint AP results impractical. However, AP and AR remain valuable tools for understanding keypoint-specific behaviors.

Model	Method	Hips		Knees		Ankles	
		AP	**AR**	**AP**	**AR**	**AP**	**AR**
HRNet	Baseline	0.906	0.930	0.942	0.965	0.968	0.984
	Fine-tuned	0.910	0.935	0.937	0.962	0.964	0.982
	Seg. Attn.	0.907	0.931	0.945	0.966	0.969	0.984
	ControlNet	0.914	0.935	0.941	0.964	0.972	**0.985**
Sapiens-0.3B	Baseline	0.886	0.938	**0.958**	**0.976**	**0.973**	**0.985**
	Fine-tuned	0.923	0.944	0.947	0.967	0.962	0.980
	Seg. Attn.	0.918	0.939	0.949	0.967	0.958	0.978
	ControlNet	**0.927**	**0.948**	0.953	0.972	0.968	0.984

Table 5.5: Average Precision (AP) and Average Recall (AR) for lower-body landmarks: hips, knees, and ankles.

Despite the lower PCK scores for hips, their AP remains relatively high across all models, exceeding 0.90 for HRNet and peaking at 0.927 under Sapiens-0.3B ControlNet. However, the discrepancy between AP and PCK indicates that these predictions often fail to meet the strict spatial accuracy requirement of PCK.

This pattern suggests that the model systematically predicts hips inside the frame, even in cases where truncation should result in their absence. Rather than omitting hip keypoints when insufficient visual cues exist, the model appears to extrapolate their locations based on partial torso visibility. This bias toward in-image hip predictions leads to an overestimating presence and introduces systematic misplacement, explaining the drop in PCK. This behavior is analyzed further in the next section.

In contrast, knees and ankles exhibit both high PCK and high AP/AR, indicating that their keypoint estimates are both consistently detected and precisely localized. Knees surpass AP 0.94 for HRNet and exceed 0.95 under Sapiens-0.3B, while ankles maintain the highest AP and AR across all models. These results confirm that knees and ankles benefit from stronger spatial cues, while hip keypoints remain a failure case due to truncation ambiguity.

These findings highlight a critical challenge in hip localization: partial truncation misleads the model into systematically placing hips within the visible frame, even when they should be missing. Segmentation-based attention and ControlNet yield only marginal improvements, with fine-tuning showing the most notable gains under Sapiens-0.3B. However, even in the best case (Sapiens-0.3B ControlNet), hip keypoint estimation remains the most error-prone among the lower-body landmarks.

5.3 Failure Cases

This section qualitatively analyzes failure cases related to truncation and segmentation accuracy to better understand the persistent challenges in hip localization. The goal is to examine how these factors contribute to systematic errors in keypoint estimation and whether proposed modifications, such as segmentation-based attention or ControlNet, provide meaningful improvements.

Hip Performance Despite generally strong pose detection performance, hip keypoints remain a notable failure case across all models. The results in Table 5.4 highlight a significant drop in hip localization accuracy compared to knees and ankles, with PCK scores consistently lower across all models. Even the best-performing configuration, Sapiens-0.3B ControlNet, achieves only 0.766 PCK for hips, whereas knees and ankles exceed 0.90. The discrepancy between high AP and low PCK further suggests that while the models frequently detect hip keypoints, they often place them with greater spatial error.

One key factor contributing to these errors is truncation ambiguity. The scanner's multi-camera setup often captures partial rather than complete lower-body truncations. Unlike cases where keypoints are entirely absent and the model can infer their occlusion, these partial

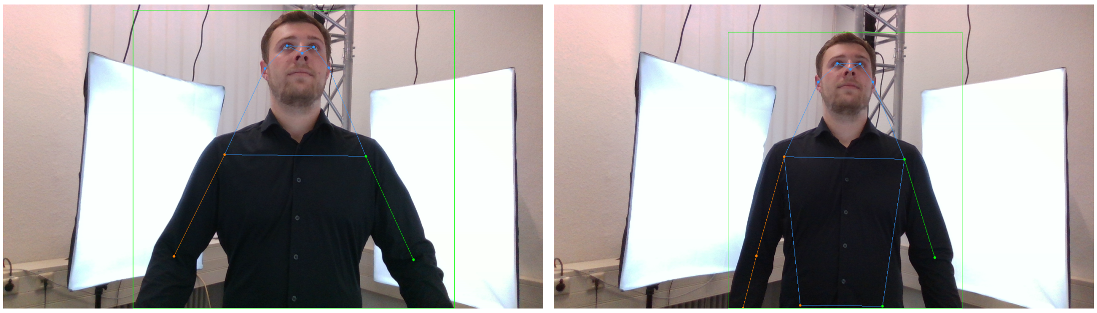


Figure 5.1: Scanner images illustrating hip keypoint errors. On the left, higher truncation prevents the network from estimating hip keypoints. On the right, slightly reduced truncation leads to incorrect in-image placement.

truncations introduce visual uncertainty. The visible torso may only marginally intersect with the hips, leaving the model without clear localization cues. As a result, the model systematically estimates hip keypoints within the image, even when they should be missing, leading to frequent misplacement.

Figure 5.1 illustrates this issue. In the left image, where truncation is more extreme, the model omits the hip keypoints entirely. In contrast, the right image, which has slightly less truncation, leads to incorrect in-image placement of the hips. This highlights how even minor changes in camera positioning or subject stance can shift the truncation boundary, leading to inconsistent keypoint behavior. Given that PCK measures correctness within a strict distance threshold, such systematic misplacement results in a disproportionate drop in PCK for hips compared to other lower-body keypoints. Given the 3D scanner camera setup, many images with such ambiguous truncations are captured. Figure 5.2 visualizes a select few more.

Another contributing factor is segmentation uncertainty. The Sapiens model employs segmentation-based attention to refine keypoint estimates. This strategy only provides marginal improvements for hips. One possible reason is how the segmentation model handles clothed body regions. Instead of segmenting individual body parts, the model often identifies entire clothing regions, such as *upper clothing* and *lower clothing*. This reduces the granularity of body part information, leading to ambiguous keypoint placement at transition areas like the hips.

Figure 5.3 illustrates this segmentation issue. While the bare arms were segmented into more fine-grained regions, the whole lower body is a single continuous region. Since the model relies on segmentation cues to refine keypoint placement, this lack of precise body part delineation further impacts hip and, in general, keypoint localization.

Compared to knees and ankles, the lower accuracy of hip localization stems from a combina-

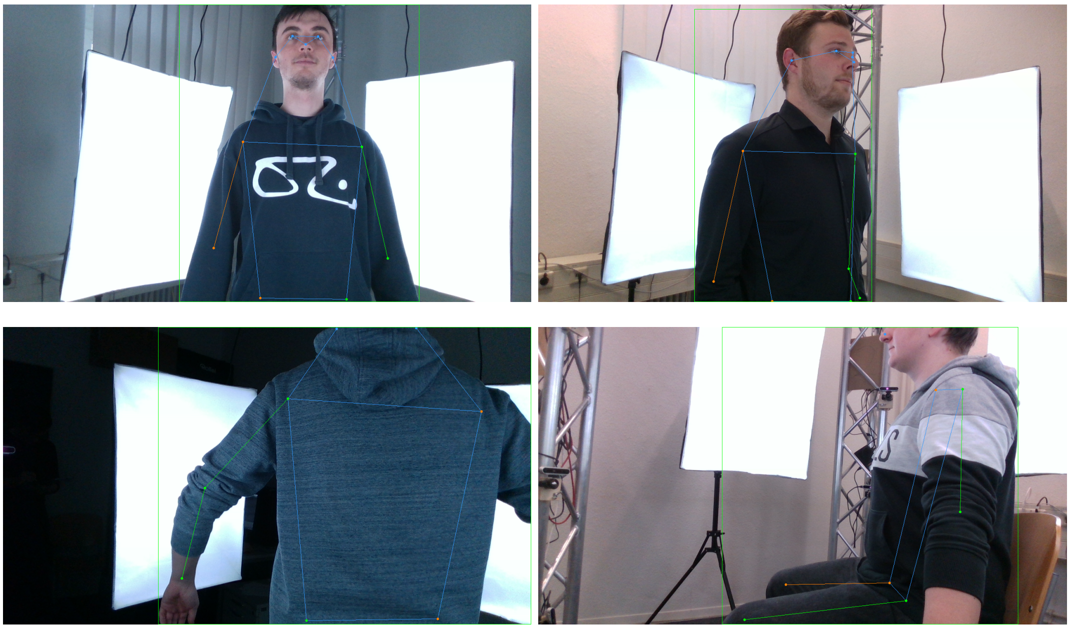


Figure 5.2: Scanner images with detection results showing hip keypoint errors. Partial truncation leads to incorrect predictions, as the model is uncertain whether the hips lie inside or outside the visible region.

tion of truncation ambiguity, systematic misplacement, and segmentation limitations. Unlike knees and ankles, which benefit from strong reference cues such as clear joint articulations, hips often lack well-defined anchors in partially visible cases. This results in a pattern where the model estimates hip positions with high confidence, as reflected in the high AP scores, but fails to place them accurately, leading to the observed drop in PCK.

Although fine-tuning and segmentation-based attention introduce refinements, neither strategy fundamentally resolves the issue of systematic hip misplacement in truncation scenarios. Future improvements may require more robust handling of truncations, such as explicit occlusion-aware training strategies or dynamic keypoint confidence modeling. These findings confirm that hip localization remains a major challenge, with both truncation-induced ambiguity and segmentation limitations contributing to persistent errors.

Segmentation Attention Performance The results indicate that segmentation-guided attention provides only marginal improvements in keypoint localization. This limited impact can be attributed to how the HRNet and Sapiens backbones process feature maps. As illustrated in Figure 5.4, both backbones inherently generate feature maps that resemble pseudo-heatmaps, where high-activation regions correspond to keypoint locations while the surrounding areas

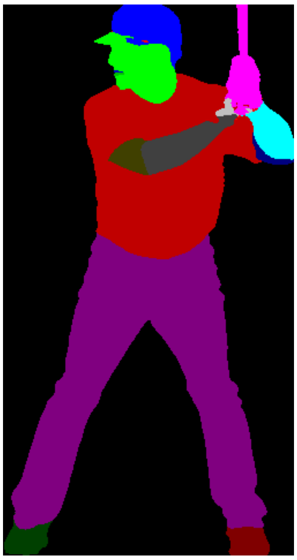


Figure 5.3: Example segmentation of a MSCOCO [Lin+15] image, where the lower body is segmented as a single continuous region instead of individual body parts as seen in the upper body.

exhibit low activation.

Since segmentation-guided attention is applied post-feature extraction, it does not fundamentally modify these internal representations. Instead, it acts as an additional constraint, reinforcing keypoint predictions within the segmented regions. This explains why segmentation-guided attention primarily improves AP, as it helps suppress irrelevant activations, but has a negligible effect on PCK, which depends on precise spatial localization. Given that the feature extraction process remains largely unchanged, segmentation attention is unable to resolve structural ambiguities, particularly in cases of partial truncation where segmentation lacks sufficient precision.

5.4 Discussion and Insights

These experiments confirm that truncated images, common in a multi-camera scanner, pose significant challenges for human pose estimation. Segmentation-based attention and ControlNet boost global spatial awareness and moderately improve AP metrics, yet they only slightly enhance fine-grained localization (PCK). HRNet, having strong feature extraction, sees limited gains from segmentation-based conditioning, whereas Sapiens-0.3B benefits more—particularly under ControlNet—due to its transformer architecture and self-attention mechanisms.

ControlNet stands out for Sapiens-0.3B in raising AR and mid-to-high threshold precision,

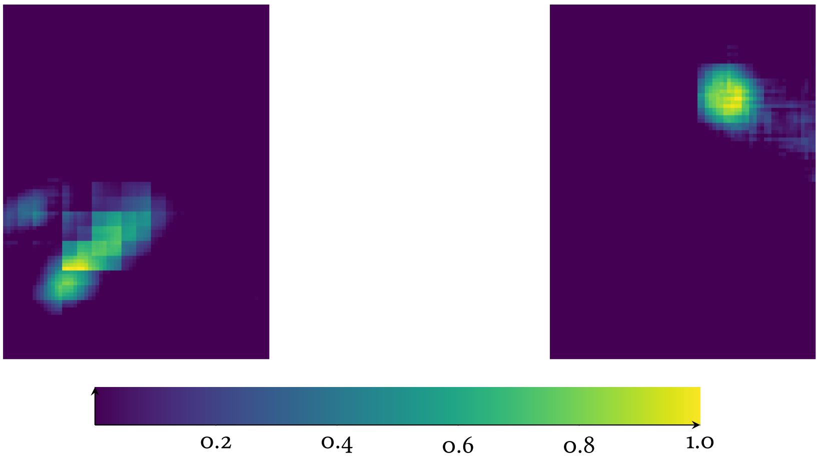


Figure 5.4: Feature maps from the HRNet backbone, highlighting strong activations around distinct joint locations and serving as pseudo-heatmaps.

reflecting a more robust interpretation of segmentation cues. Nonetheless, the most consistent performance gains across both models come from fine-tuning with truncation-augmented data. This training approach helps the network adapt to the partial views instead of relying on assumptions formed under full-body training conditions.

Inspection of lower-body joints highlights how coarse torso segmentation impedes reliable hip detection. By contrast, knees and ankles are easier to localize if they are visible at all, since their truncation boundaries do not introduce the same ambiguity as those that cut through the hip region.

6 Conclusion

This work examined how segmentation-guided attention and ControlNet-based multi-layer conditioning can mitigate human pose estimation errors under realistic truncation scenarios. A 3D scanner dataset that captures partial body views was used to reflect practical conditions for human pose estimation in multi-camera setups. Two architectures, a CNN (HRNet) and a transformer-based model (Sapiens-0.3B), were evaluated to explore how different feature extraction strategies respond to partial-view inputs and segmentation cues. Experiments revealed that segmentation-guided methods enhance high-level spatial awareness by enforcing structural constraints and improving average precision metrics. However, fine-grained keypoint localization did not consistently benefit, especially under severe truncation. ControlNet offered slight advantages over single-stage segmentation attention, but these gains were marginal for HRNet and low for Sapiens-0.3B. This indicates that both models already capture significant spatial context and that segmentation-guided attention does not add the necessary auxiliary information to improve human pose estimation under partial observations greatly.

Segmentation-based methods proved more helpful for visible keypoints since segment masks clarify ambiguous boundaries within the frame, and the global attention mechanism of Sapiens-0.3B integrated segmentation constraints more effectively than HRNet. Nonetheless, severe truncation near the hips or lower-body joints remained problematic, as broad segmentation masks supply insufficient detail for inferring occluded keypoints. This limitation was amplified when coarse segmentation labels (for instance, a single torso region) overshadowed anatomical subdivisions.

Augmenting training data with partial-view images consistently improved keypoint detection, indicating that architectural modifications alone do not substitute for an appropriate training distribution. Learning from truncated examples helped bridge the gap between full-body pretraining and real-world truncated poses. HRNet and Sapiens-0.3B benefited from this approach, showing that data-centric strategies effectively improve model robustness.

6.1 Future Directions

Both approaches in this study use an external segmentation model to create the body part segmentations before keypoint estimation, which makes the process resource-intensive and slow. A promising future direction involves integrating segmentation and keypoint estimation within a single model as has been done by [He+17], allowing a shared backbone to jointly predict both outputs and train it specifically for truncated poses. This approach could improve keypoint localization while reducing computational overhead by eliminating the need for an external segmentation model.

Future work could enhance segmentation granularity by dividing the torso into smaller anatomical regions, such as the upper torso, waist, and hips. This finer segmentation may introduce more localized constraints, helping to reduce ambiguity in partially visible areas. Additionally, an adaptive weighting mechanism that dynamically adjusts the influence of segmentation conditioning based on the visibility or confidence of each keypoint or segmentation mask could help mitigate errors caused by incomplete or inaccurate segmentation.

While finer segmentation maps may enhance the utility of segmentation attention, their overall effectiveness for improved pose estimation under partial views remains uncertain, as this study demonstrated. Segmentation maps provide valuable contextual information about visible body parts but do not fully resolve the challenge of missing pose information due to truncation. Future research could explore complementary strategies to address these limitations better.

Therefore, more extensive truncation augmentation, including varied cropping patterns, perspective distortions, and extreme camera angles, could enhance the model's ability under challenging conditions.

Although segmentation-based conditioning has shown modest benefits on truncated HPE in this study, its potential impact on occlusions warrants further investigation.

6.2 Concluding Remarks

Optimizing the training distribution with truncation-specific augmentation emerged as the most effective strategy for improving human pose estimation when key body regions are missing. Segmentation-based conditioning provides structural benefits but does not entirely compensate for extensive or highly uncertain truncations. The path forward includes more precise segmentation maps, adaptive weighting, and carefully designed data augmentations that address persistent boundary errors around areas such as the hips, driving greater reliability in human pose estimation for real-world contexts.

Bibliography

[And+14] Mykhaylo Andriluka et al. "2D Human Pose Estimation: New Benchmark and State of the Art Analysis." In: *2014 IEEE Conference on Computer Vision and Pattern Recognition.* June 2014, pp. 3686–3693.

[AP22] Taravat Anvari and Kyoungju Park. "3D Human Body Pose Estimation in Virtual Reality: A Survey." In: *2022 13th International Conference on Information and Communication Technology Convergence (ICTC).* Oct. 2022, pp. 624–628.

[Bau+15] Tobias Baur et al. "Context-Aware Automated Analysis and Annotation of Social Human–Agent Interactions." In: *ACM Trans. Interact. Intell. Syst.* 5.2 (June 2015), 11:1–11:33.

[BM21] Aritz Badiola-Bengoa and Amaia Mendez-Zorrilla. "A Systematic Review of the Application of Camera-Based Human Pose Estimation in the Field of Sport and Physical Exercise." In: *Sensors* 21.18 (Jan. 2021), p. 5996.

[Car+20] Nicolas Carion et al. "End-to-End Object Detection with Transformers." In: *Computer Vision – ECCV 2020.* Ed. by Andrea Vedaldi et al. Vol. 12346. Cham: Springer International Publishing, 2020, pp. 213–229.

[Che+18] Yilun Chen et al. "Cascaded Pyramid Network for Multi-person Pose Estimation." In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition.* June 2018, pp. 7103–7112.

[Con20] MMPose Contributors. *OpenMMLab Pose Estimation Toolbox and Benchmark.* `https://github.com/open-mmlab/mmpose`. 2020.

[Cub+19] Ekin D. Cubuk et al. "AutoAugment: Learning Augmentation Strategies From Data." In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* June 2019, pp. 113–123.

[Cub+20] Ekin Dogus Cubuk et al. "RandAugment: Practical Automated Data Augmentation with a Reduced Search Space." In: *Advances in Neural Information Processing Systems.* Vol. 33. Curran Associates, Inc., 2020, pp. 18613–18624.

[Dev+19] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* May 2019. arXiv: `1810.04805 [cs]`.

[DMS18] Nikita Dvornik, Julien Mairal, and Cordelia Schmid. "Modeling Visual Context Is Key to Augmenting Object Detection Datasets." In: *Proceedings of the European Conference on Computer Vision (ECCV).* 2018, pp. 364–380.

[Dos+21] Alexey Dosovitskiy et al. *An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale.* June 2021. arXiv: `2010.11929 [cs]`.

[DT17] Terrance DeVries and Graham W. Taylor. *Improved Regularization of Convolutional Neural Networks with Cutout.* Nov. 2017. arXiv: `1708.04552 [cs]`.

[Gao+25] Zheyan Gao et al. "A Systematic Survey on Human Pose Estimation: Upstream and Downstream Tasks, Approaches, Lightweight Models, and Prospects." In: *Artificial Intelligence Review* 58.3 (Jan. 2025), p. 68.

[GEB16] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. "Image Style Transfer Using Convolutional Neural Networks." In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* June 2016, pp. 2414–2423.

[GZF21] Kehong Gong, Jianfeng Zhang, and Jiashi Feng. *PoseAug: A Differentiable Pose Augmentation Framework for 3D Human Pose Estimation.* May 2021. arXiv: `2105.02465 [cs]`.

[Han+24] Gangtao Han et al. *Occluded Human Pose Estimation Based on Limb Joint Augmentation.* Oct. 2024. arXiv: `2410.09885 [cs]`.

[Han+25] Gangtao Han et al. "Occluded Human Pose Estimation Based on Limb Joint Augmentation." In: *Neural Computing and Applications* 37.3 (Jan. 2025), pp. 1241–1253.

[He+16] Kaiming He et al. "Deep Residual Learning for Image Recognition." In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* IEEE Computer Society, June 2016, pp. 770–778.

[He+17] Kaiming He et al. "Mask R-CNN." In: *2017 IEEE International Conference on Computer Vision (ICCV).* Oct. 2017, pp. 2980–2988.

[He+22] Kaiming He et al. "Masked Autoencoders Are Scalable Vision Learners." In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2022, pp. 16000–16009.

[Hua+17] Gao Huang et al. "Densely Connected Convolutional Networks." In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* IEEE Computer Society, July 2017, pp. 2261–2269.

[Jia+24] Wentao Jiang et al. “Data Augmentation in Human-Centric Vision.” In: *Vicinagearth* 1.1 (Oct. 2024), p. 8.

[KB17] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization.* Jan. 2017. arXiv: `1412.6980 [cs]`.

[Khi+24] Rawal Khirodkar et al. *Sapiens: Foundation for Human Vision Models.* Aug. 2024. arXiv: `2408.12569 [cs]`.

[Kir+19] Alexander Kirillov et al. *Panoptic Segmentation.* Apr. 2019. arXiv: `1801.00868 [cs]`.

[Kna24] Pawel Knap. *Human Modelling and Pose Estimation Overview.* June 2024. arXiv: `2406.19290 [cs]`.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. “ImageNet Classification with Deep Convolutional Neural Networks.” In: *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1.* Vol. 1. NIPS’12. Red Hook, NY, USA: Curran Associates Inc., Dec. 2012, pp. 1097–1105.

[Lan+23] Gongjin Lan et al. “Vision-Based Human Pose Estimation via Deep Learning: A Survey.” In: *IEEE Transactions on Human-Machine Systems* 53.1 (Feb. 2023), pp. 253–268.

[Lec+98] Y. Lecun et al. “Gradient-Based Learning Applied to Document Recognition.” In: *Proceedings of the IEEE* 86.11 (Nov. 1998), pp. 2278–2324.

[LH17] Ilya Loshchilov and Frank Hutter. *SGDR: Stochastic Gradient Descent with Warm Restarts.* May 2017. arXiv: `1608.03983 [cs]`.

[LH19] Ilya Loshchilov and Frank Hutter. *Decoupled Weight Decay Regularization.* Jan. 2019. arXiv: `1711.05101 [cs]`.

[Li+21] Yanjie Li et al. “TokenPose: Learning Keypoint Tokens for Human Pose Estimation.” In: *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 2021, pp. 11313–11322.

[Lin+15] Tsung-Yi Lin et al. *Microsoft COCO: Common Objects in Context.* Feb. 2015. arXiv: `1405.0312`.

[Lin+17] Tsung-Yi Lin et al. “Feature Pyramid Networks for Object Detection.” In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2017, pp. 2117–2125.

[Liu+21] Ze Liu et al. "Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows." In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV).* Oct. 2021, pp. 9992–10002.

[LSD15] Jonathan Long, Evan Shelhamer, and Trevor Darrell. *Fully Convolutional Networks for Semantic Segmentation.* Mar. 2015. arXiv: `1411.4038 [cs]`.

[LVX20] Yalin Liao, Aleksandar Vakanski, and Min Xian. "A Deep Learning Framework for Assessing Physical Rehabilitation Exercises." In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 28.2 (Feb. 2020), pp. 468–477.

[Ma+22] Haoyu Ma et al. "PPT: Token-Pruned Pose Transformer for Monocular and Multi-view Human Pose Estimation." In: *Computer Vision – ECCV 2022.* Ed. by Shai Avidan et al. Cham: Springer Nature Switzerland, 2022, pp. 424–442.

[NYD16] Alejandro Newell, Kaiyu Yang, and Jia Deng. *Stacked Hourglass Networks for Human Pose Estimation.* July 2016. arXiv: `1603.06937 [cs]`.

[Pas+17] Adam Paszke et al. "Automatic differentiation in PyTorch." In: *NIPS-W.* 2017.

[PLP20] Soonchan Park, Sang-baek Lee, and Jinah Park. "Data Augmentation Method for Improving the Accuracy of Human Pose Estimation with Cropped Images." In: *Pattern Recognition Letters* 136 (Aug. 2020), pp. 244–250.

[PP21] Soonchan Park and Jinah Park. "Localizing Human Keypoints beyond the Bounding Box." In: *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW).* Montreal, BC, Canada: IEEE, Oct. 2021, pp. 1602–1611.

[RP17] Matteo Ruggero Ronchi and Pietro Perona. "Benchmarking and Error Diagnosis in Multi-instance Pose Estimation." In: *2017 IEEE International Conference on Computer Vision (ICCV).* Venice: IEEE, Oct. 2017, pp. 369–378.

[Sár+21] István Sárándi et al. "MeTRAbs: Metric-Scale Truncation-Robust Heatmaps for Absolute 3D Human Pose Estimation." In: *IEEE Transactions on Biometrics, Behavior, and Identity Science* 3.1 (Jan. 2021), pp. 16–30.

[SK19] Connor Shorten and Taghi M. Khoshgoftaar. "A Survey on Image Data Augmentation for Deep Learning." In: *Journal of Big Data* 6.1 (July 2019), p. 60.

[SSP03] Patrice Y. Simard, Dave Steinkraus, and John C. Platt. "Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis." In: *Proceedings of the Seventh International Conference on Document Analysis and Recognition - Volume 2.* Vol. 2. ICDAR '03. USA: IEEE Computer Society, Aug. 2003, p. 958.

[Sun+19] Ke Sun et al. "Deep High-Resolution Representation Learning for Human Pose Estimation." In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* June 2019, pp. 5686–5696.

[SZ15] Karen Simonyan and Andrew Zisserman. *Very Deep Convolutional Networks for Large-Scale Image Recognition.* Apr. 2015. arXiv: `1409.1556 [cs]`.

[TN18] Luke Taylor and Geoff Nitschke. "Improving Deep Learning with Generic Data Augmentation." In: *2018 IEEE Symposium Series on Computational Intelligence (SSCI).* Nov. 2018, pp. 1542–1547.

[TW19] Wei Tang and Ying Wu. "Does Learning Specific Features for Related Parts Help Human Pose Estimation?" In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2019, pp. 1107–1116.

[Vas+23] Ashish Vaswani et al. *Attention Is All You Need.* Aug. 2023. arXiv: `1706.03762 [cs]`.

[Xu+22] Yufei Xu et al. *ViTPose: Simple Vision Transformer Baselines for Human Pose Estimation.* Oct. 2022. arXiv: `2204.12484 [cs]`.

[XWW18] Bin Xiao, Haiping Wu, and Yichen Wei. "Simple Baselines for Human Pose Estimation and Tracking." In: *Proceedings of the European Conference on Computer Vision (ECCV).* 2018, pp. 466–481.

[Yan+21] Sen Yang et al. "TransPose: Keypoint Localization via Transformer." In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV).* Oct. 2021, pp. 11782–11792.

[YR13] Yi Yang and Deva Ramanan. "Articulated Human Detection with Flexible Mixtures of Parts." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.12 (Dec. 2013), pp. 2878–2890.

[Yua+21] Yuhui Yuan et al. *HRFormer: High-Resolution Transformer for Dense Prediction.* Nov. 2021. arXiv: `2110.09408 [cs]`.

[Yun+19] Sangdoo Yun et al. "CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features." In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV).* Oct. 2019, pp. 6022–6031.

[Zan+23] Andrei Zanfir et al. "HUM3DIL: Semi-supervised Multi-modal 3D HumanPose Estimation for Autonomous Driving." In: *Proceedings of The 6th Conference on Robot Learning.* PMLR, Mar. 2023, pp. 1114–1124.

[Zha+19] Feng Zhang et al. *Distribution-Aware Coordinate Representation for Human Pose Estimation.* Oct. 2019. arXiv: `1910.06278`.

[Zha+24] Tong Zhang et al. "Enhancement and Optimisation of Human Pose Estimation with Multi-Scale Spatial Attention and Adversarial Data Augmentation." In: *Information Fusion* 111 (Nov. 2024), p. 102522.

[Zho+23] Lijuan Zhou et al. *Human Pose-based Estimation, Tracking and Action Recognition with Deep Learning: A Survey.* Oct. 2023. arXiv: `2310.13039 [cs]`.

[ZRA23] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. "Adding Conditional Control to Text-to-Image Diffusion Models." In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV).* Oct. 2023, pp. 3813–3824.