

Multi-Person Pose Tracking in Videos

DISSERTATION

zur Erlangung des Doktorgrades (*Dr. rer. nat.*)

der Mathematisch-Naturwissenschaftlichen Fakultät

der Rheinischen Friedrich-Wilhelms-Universität, Bonn

vorgelegt von

ANDREAS DÖRING

aus

Köln, Deutschland

Bonn, 2025

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der
Rheinischen Friedrich-Wilhelms-Universität Bonn

Gutachter/Betreuer / Examiner/Advisor: Prof. Dr. Juergen Gall
Gutachter / Examiner: Prof. Dr. Vasileios Belagiannis
Tag der Promotion / Day of Promotion: 07.02.2025
Erscheinungsjahr / Year of Publication: 2025

Abstract

by Andreas Döring

for the degree of

Doctor rerum naturalium

This thesis addresses the complex challenges of multi-person pose estimation and tracking in dynamic, real-world environments. Motivated by advancements in artificial intelligence and the growing need for natural human-machine interaction, this research focuses on developing robust computer vision systems that can accurately estimate and track human poses. These systems have broad applications, ranging from healthcare and surveillance to autonomous driving, sports analysis, and entertainment. However, real-world scenarios, especially those involving multiple interacting individuals, frequently result in keypoint inaccuracies due to occlusions and overlapping body parts, as seen in sports where highly articulated poses are common. Effective pose tracking requires robust occlusion handling and the ability to preserve identities, even across videos.

Previous approaches formulated multi-person pose tracking as spatiotemporal graph partitioning problems, which are computationally expensive and unsuitable for real-time applications. More recent methods using flow-, motion-, or appearance-based assignments struggle when individuals are occluded, and appearance-based methods face challenges with partial occlusions and visually similar individuals. Additionally, existing datasets for multi-person pose tracking are sparsely annotated and lack identity information.

This research tackles several key challenges, including occlusions, motion blur, crowded scenes, and varying lighting conditions, which heavily impact pose estimation and tracking accuracy. We propose a top-down multi-person pose estimation and tracking framework, with the core contribution being a self-supervised keypoint correspondence network that recovers missed detections, associates keypoints across video frames, and restores broken tracks due to occlusions. Moreover, recognizing the limitations of existing datasets, we introduce PoseTrack21 - a densely annotated dataset designed to improve training and evaluation in multi-person pose tracking, multi-object tracking, and person search. PoseTrack21 includes comprehensive annotations for keypoints, bounding boxes, and person identities, enabling joint evaluation across these tasks. Additionally, we present a Gated Attention Transformer model that integrates appearance and keypoint-based similarities through a novel gating mechanism, improving the robustness of pose tracking systems. This model includes a matching layer to eliminate redundant detections and a pose-conditioned re-identification network to handle occlusions, enhancing tracking consistency.

Our proposed methods are extensively evaluated on the PoseTrack 2017, 2018, and PoseTrack21 datasets, demonstrating their effectiveness. The contributions of this thesis establish a foundation for more efficient multi-person pose tracking systems, advancing the state-of-the-art in this crucial area of computer vision.

Keywords: Multi-Person Pose Tracking, Multi-Person Pose Estimation, Multi-Object Tracking, Person Re-Identification

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my advisor, Prof. Dr. Juergen Gall, for his unwavering guidance and support throughout my PhD journey. I was fortunate to have the freedom to explore my ideas, and his valuable feedback and insightful advice consistently helped me refine and improve them.

I would also like to sincerely thank the members of my dissertation committee, Prof. Dr. Vasileios Belagiannis, Prof. Dr. Dominik Bach, and Prof. Dr. Ingrid-Maria Gregor, for taking the time to review my thesis and provide their valuable insights and expertise.

I am immensely grateful to my colleagues and friends in the Computer Vision Group. Our stimulating discussions, shared motivation, and fun times together made this journey both rewarding and enjoyable.

Finally, I want to thank my family and friends for their endless support, encouragement, and patience, especially during the more challenging times of this PhD. Your belief in me has been my greatest source of strength.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Problem Formulation	3
1.3	Contributions	4
1.3.1	Self-Supervised Keypoint Correspondences for Multi-Person Pose Tracking .	4
1.3.2	PoseTrack21: A Dataset for Person Search, Multi-Object Tracking and Multi-Person Pose Tracking	4
1.3.3	A Gated Attention Transformer for Multi-Person Pose Tracking	5
1.4	Thesis Structure	5
2	Related Work	7
2.1	2D Pose Estimation	8
2.1.1	Single Person Pose Estimation	8
2.1.2	Multi-Person Pose Estimation	9
2.1.3	Multi-Frame Multi-Person Pose Estimation and Tracking	17
2.2	Object Detection and Multi-Object Tracking	24
2.2.1	Object Detection in MOT	25
2.2.2	Multi-Object Tracking	26
2.3	Person Search and Person Re-Identification	33
3	Preliminaries	39
3.1	Neural Networks	40
3.1.1	Neuron	41
3.1.2	Activation Functions	41
3.1.3	Objective Functions	43
3.1.4	Training	44
3.1.5	Neural Network Normalization	45
3.2	Convolutional Neural Networks	47
3.2.1	Transposed Convolution	48
3.3	Transformer Networks	49
3.3.1	Attention Mechanism	49
3.3.2	Transformer Architecture	50
3.4	Convolutional Neural Networks for Multi-Person Pose Tracking	53
3.4.1	Backbone Networks	53
3.4.2	Multi-Person Pose Estimation and Tracking	55
3.5	Person Re-Identification and Person Search	57
3.6	Evaluation Metrics	60
3.6.1	2D Multi-Person Pose Estimation	60
3.6.2	Person Re-Identification and Person Search	61
3.6.3	Multi-Target Tracking	62

4 Self-supervised Keypoint Correspondences for Multi-Person Pose Estimation and Tracking in Videos	67
4.1 Introduction	68
4.2 Method Overview	70
4.2.1 Pose Estimation Model	70
4.3 Keypoint Correspondence Network	71
4.3.1 Siamese Matching Module	72
4.3.2 Refinement Module	73
4.3.3 Training	73
4.4 Multi-Person Pose Tracking	73
4.4.1 Recover Missed Detections	74
4.4.2 Tracking	74
4.4.3 Track Merging	75
4.5 Experiments and Results	76
4.5.1 Implementation Details	76
4.5.2 Baselines	77
4.5.3 Effect of Joint Detection Threshold and Pose Recovery	78
4.5.4 Effect of Refinement Module and Duplicate Removal	79
4.5.5 Impact of τ_{corr}	79
4.5.6 Comparison with State-of-the-art Methods	79
4.5.7 Qualitative Results	80
4.5.8 Failure Cases	80
4.6 Conclusion	81
5 PoseTrack21: A Dataset for Person Search, Multi-Object Tracking and Multi-Person Pose Tracking	87
5.1 Introduction	88
5.2 Related Datasets	89
5.3 The PoseTrack21 Dataset	92
5.4 Multi-Person Re-ID Pose Tracking	94
5.4.1 Proposed CorrTrack Baselines	94
5.4.2 Proposed Tracktor++ Baselines	95
5.5 Analysis	96
5.6 Dataset Statistics	102
5.7 Ablation Studies	102
5.8 Discussion	103
6 A Gated Attention Transformer	107
6.1 Introduction	108
6.2 Gated Attention Transformer for Multi-Person Pose Tracking	109
6.2.1 Gated Attention Decoder	111
6.2.2 Confidence-Guided Track Update	113
6.2.3 Embeddings E_D^t and $E_{T \rightarrow D}$	114

6.2.4	Training Objective	115
6.3	Experiments	117
6.3.1	Datasets and Evaluation	117
6.3.2	Comparison with State of the Art	117
6.3.3	Implementation Details	119
6.3.4	Ablation Studies	120
6.3.5	Comparison to CORRTRACK + REID	124
6.4	Conclusion	124
7	Conclusion	127
7.1	Overview	127
7.2	Contributions and Discussion	128
7.3	Future Work	129

List of Figures

3.1	Illustration of a Multi-Layer Perceptron (MLP) with two hidden layers.	40
3.2	Illustration of a biological neuron. An incoming stimulus is received by the dendrites and processed by the cell body. Depending on the incoming signal, the neuron transmits information through the axon to subsequent neurons. Image adopted from Doering (2018)	41
3.3	Illustration of a technical neuron. The incoming input (stimulus) is weighted at the synapsis. All weighted stimuli are linearly combined within the cell body. The activation of a technical neuron is simulated by an activation function that operates on the received and linearly combined stimulus. Image adopted from Doering (2018)	42
3.4	Overview of the most commonly used activation functions in neural networks.	43
3.5	Illustration of batch normalization and instance normalization. While batch normalization normalizes across the batch dimension, instance normalization normalizes across the spatial dimensions of each sample.	45
3.6	Illustration of transposed convolution with kernel size 2 and stride 2.	48
3.7	Illustration of the transformer architecture. The input features are augmented with positional encodings to preserve spatiotemporal information and then processed by N encoder layers. Each layer is comprised of multi-head self-attention and feed forward layers. Similarly, output embeddings are processed by N decoder layers. Each decoder layer comprises an additional multi-headed cross attention layer that takes as input the encoded input embeddings from the encoder. Figure from Vaswani et al. (2017)	51
3.8	Illustration of a positional encoding as defined in (3.22) for an input sequence of length $N = 100$ and a feature dimension of $C=256$	52
3.9	Illustration of an inception block.	53
3.10	Illustration of a residual block.	54
3.11	Illustration of the two-branch architecture for multi-person pose estimation proposed in Cao et al. (2017) . Each branch processed image features $\mathbf{F}$ and estimates keypoint heatmaps $\mathbf{S}$ and part affinity fields $\mathbf{L}$. In addition to the image features, succeeding stages take as input keypoint heatmaps and part affinity fields of the previous stage.	56
3.12	Proposed architecture of JointFlow (Doering et al., 2018). a) illustrates the Siamese pose estimation architecture, adopted from Cao et al. (2017) and b) is the architecture of the temporal flow fields prediction branch. Figure from Doering et al. (2018)	58
3.13	Standard person re-identification pipeline extended by Luo et al. (2019b) with a batch normalization neck. Figure from Luo et al. (2019b)	59
3.14	Illustration of the person search architecture proposed in SeqNet (Li and Miao, 2021). a) illustrates the baseline architecture for person approaches based on Faster R-CNN (Ren et al., 2015), as used in Chen et al. (2020b) . b) SeqNet incorporates an additional Faster R-CNN head, which takes as input proposals from the RPN network (denoted in blue).	60

- 3.15 Toy example illustrating the limited expressivity of CMC. (a) and (b) depict ranked lists, each containing two true positives (denoted by green boxes). In (a), both true positives are ranked at the top, resulting in a CMC curve equal to 1 and an Average Precision (AP) of 1. In (b), the second true positive appears at the fifth position, lowering the precision and yielding an AP of 0.71. Despite this, the CMC curve remains 1. 62
- 4.1 Our contributions: (Left) We use keypoint correspondences to recover missed pose detections by using the temporal context of the previous frame. (Right) We use keypoint correspondences to associate detected and recovered pose detections for the task of multi-person pose tracking. 69
- 4.2 Given a sequence of frames, we detect a set of person bounding boxes and perform top-down pose estimation. Our proposed method uses keypoint correspondences to (i) recover missed detections and (ii) to associate detected and recovered poses to perform tracking. The entire framework does not require any video data for training since the network for estimating keypoint correspondences is trained on single images using self-supervision. 70
- 4.3 Our two-stage pose estimation framework. Each stage uses GoogleNet (Szegedy et al., 2015) as the backbone. The features extracted by the backbone in the first stage are fed into a deconvolution layer block to produce pose and joint offset maps. The backbone features, pose heatmaps, and joint offset maps from the first stage are fed into the second stage to produce refined pose and joint offset maps. 71
- 4.4 Keypoint correspondence network. The Siamese network takes images $\mathbf{I}_1$ and $\mathbf{I}_2$ and keypoints $\mathcal{P} = \{P^p\}_{1:N_p}$ for all persons p in image $\mathbf{I}_1$ as input and generates the feature maps E_1 and E_2 , respectively. The keypoints of the different persons are shown in green and yellow, respectively. For each keypoint, a descriptor e_j^p is extracted from E_1 and convolved with the feature map E_2 to generate an affinity map A_j^p . To improve the affinity maps for each person, the refinement network takes E_1 , E_2 and the affinity maps A_j^p for person p as input and generates refined affinity maps C_j^p 72
- 4.5 Recovering missed detections. (a) Person detected by the top-down pose estimation framework in frame $f-1$. (b) The person was missed by the top-down pose estimation framework in frame f due to occlusion. (c) Keypoint affinity maps of the missed person from frame $f-1$ to frame f . (d) Corresponding keypoints in frame f . (e) Estimated bounding box from the corresponding keypoints and the recovered pose. 74
- 4.6 Pose to track association. (a) A tracked human pose till frame $f-1$. (b) Keypoint affinity maps of the track to frame f . (c) A pose instance of frame f . The dashed lines indicate the position of each detected joint of the pose instance in the correspondence affinity maps of the tracked pose in frame $f-1$ 75

4.7	Tack merging: For the start frame f_s^q , the center frame f_c^q and the last frame f_e^q of track T^q , we estimate poses from keypoint correspondences in the start frame f_s^p of T^p , as illustrated by the colored dashed lines. We use an OKS-based similarity metric to measure the average pose similarity between the poses from correspondences and the pose in the starting frame f_s^p of track T^p	76
4.8	Qualitative comparison between optical flow and correspondences for the task of pose warping under occlusion, motion blur, and large motion. (a) Query pose in frame f . (b) Warped pose using optical flow. (c) Warped pose using correspondences. In contrast to optical flow, the correspondences warp the poses correctly despite occlusions, motion blur, or large motion.	80
4.9	Qualitative results for recovering missed detections. Best seen using the zoom function of the PDF viewer.	84
4.10	Failure cases. Duplicates by the person detector lead to multiple tracks of the same person and negatively impact the tracking performance.	85
5.1	Comparison between PoseTrack 2018 (Andriluka et al., 2018) and PoseTrack21. We densely annotated crowded scenes to increase the difficulty of multi-person pose tracking, multi-object tracking, and person search. Ignore regions are drawn in red color. Best viewed in color with a PDF reader.	89
5.2	From left to right: person search queries in PoseTrack21 with increasing difficulty.	92
5.3	While the blue and red person (ground-truth) cross their path at the last frame, both tracks (cross, ring) keep straight on. In the last frame, the identities of both persons are therefore wrongly estimated, which is penalized by keypoint HOTA. HOTA (Luiten et al., 2020) does not penalize this case if the cross overlaps with the blue circle and the ring with the red circle higher than a threshold α	93
5.4	(a) Pose tracking ablation studies. The plots show the recall (tracked objects) and the number of objects for different bounding box visibility (1-IoU), the size of the bounding box, and the number of keypoints. (b) MOT ablation studies. The plots show the recall (tracked objects) and the number of objects for different bounding box visibility (1-IoU), the size of the bounding box, and the number of keypoints.	97
5.5	Statistics for the validation set of PoseTrack21.	101
5.6	(a) Multi-person pose tracking : Ablative evaluation of false positives and MOTA with respect to bounding box size. (b) Multi-object tracking : Ablative evaluation of false positives and MOTA with respect to bounding box size. Best viewed with a PDF reader with zoom function.	104
5.7	Person search : Ablative experiments with respect to bounding box size and visibility. Best viewed with a PDF reader with zoom function.	105
6.1	Qualitative examples of our proposed method on the PoseTrack21 dataset. The first row contains visual tracking results of our method and the second row shows visualizations of CorrTrack with ReID (Chapter 5).	108

6.2	The proposed multi-person pose tracking architecture entails the following steps: a) Given a set of person crops and their respective keypoint heatmaps at time frame t , we compute pose-conditioned appearance features and feed them into N transformer encoder stages. This gives an embedding for each detection E_D^t , which will be used to measure the similarity to previous tracks. b) We also compute spatial similarities between tracks and detected persons by means of Intersection over Union (IoU) and Object Keypoint Similarity (OKS), which we then encode into a pose similarity embedding $E_{T \rightarrow D}^t$. This serves as a strong prior for matching. c) The gated attention decoder takes the embeddings of the previous tracks E_T^{t-1} , the detection embeddings E_D^t and the pose similarity embeddings $E_{T \rightarrow D}^t$ as input and updates the embeddings. It adaptively weights the spatial prior $E_{T \rightarrow D}^t$ and the appearance similarity between E_T^{t-1} and E_D^t . d) Finally, the matching stage assigns detections to tracks, removes redundant detections, and initializes new tracks.	110
6.3	Gated Attention Transformer Architecture	112
6.4	Illustration of the a) track embedding head and b) temporal pose similarity embedding head.	114
6.5	Impact of the parameters α , maximum track ages τ_{age} and the duplicate confidence threshold τ_{dup} on the tracking performance. a) visualizes the impact of α with $\tau_{age} = 60$ and $\tau_{dup} = 0.4$. b) shows the performance evaluation with respect to the maximum track age τ_{age} with $\alpha = 0.3$ and $\tau_{dup} = 0.4$. c) illustrates the impact of τ_{dup} with $\alpha = 0.3$ and $\tau_{age} = 60$	123
6.6	Additional qualitative examples of our proposed method on the PoseTrack21 dataset. In both (a) and (b), the first row contains visual tracking results of our method and the second row shows visualizations of CorrTrack with ReID (Doering et al., 2022). .	125

List of Tables

2.1	Overview of related works on multi-person pose tracking. The table is divided into top-down methods (top block) and bottom-up methods (bottom block).	21
4.1	Comparison with tracking baselines on the PoseTrack 2017 validation set. For the comparison, the detected poses based on ground-truth bounding boxes (GT Boxes) or detected bounding boxes are the same for each approach. Correspondence-based tracking consistently improves MOTA compared to the baselines and significantly reduces the number of identity switches (IDSW).	77
4.2	Effect of joint detection threshold and pose recovery on mAP and MOTA for the PoseTrack 2018 validation set. The results are shown for (left) detected poses only and (right) detected and recovered poses. As expected, recovering missed detections improves both MOTA and mAP. A good trade-off between mAP and MOTA is achieved by the joint detection threshold 0.3.	78
4.3	Comparison of mAP and MOTA for different design choices on the PoseTrack 2017 validation set.	78
4.4	Impact of τ_{corr} on mAP and MOTA on the PoseTrack 2017 validation set.	79
4.5	Comparison to the state-of-the-art on the PoseTrack 2017 and 2018 validation set for multi-frame pose estimation.	82
4.6	Comparison to the state-of-the-art on the PoseTrack 2017 and 2018 validation and test sets. Approaches marked with ⁺ use additional external training data. Approaches marked with * do not report results on the official test set.	83
5.1	Comparison with different datasets for multi-person pose tracking, MOT, and person search. Datasets marked with [†] : only training and validation set. *: no manual annotations. [◇] : dataset not available. [‡] : Only reports the total number of tracks. . . .	90
5.2	Multi-person pose tracking baselines evaluated with the keypoint MOTA metric on PoseTrack21. Approaches marked with [†] have a model trained on PoseTrack 2018 (Andriluka et al., 2018).	94
5.3	Multi-person pose tracking baselines evaluated with the keypoint HOTA metric on PoseTrack21	95
5.4	Multi-object tracking baselines evaluated with the MOTA metric on PoseTrack21MOT.	99
5.5	Multi-object tracking baselines evaluated with the HOTA metric on PoseTrack21MOT.	99
5.6	Person search evaluation on PoseTrack21. Approaches marked with [†] where re-implemented (Chen et al., 2020b).	100
6.1	Comparison to multi-person pose tracking methods on the PoseTrack21 dataset. . . .	116
6.2	Comparison to MOT methods on the PoseTrack21-MOT dataset.	117
6.3	Comparison to the state of the art on PoseTrack 2018 Andriluka et al. (2018). Two versions of the validation set have been released containing 74 (v1) and 170 (v2) sequences, respectively.	118

6.4	Comparison to state-of-the-art methods on Market1501 (Zheng et al., 2015). The best results are bold , and the second best results are <u>underlined</u>	119
6.5	Impact of several components in our pose tracking network on the tracking performance. CG-Update denotes the Confidence-Guided Track Update as discussed in Section 6.2.1.	120
6.6	Impact of using attention weights or attention logits in the α -Gate (6.5) of the Gated Attention Transformer on the overall performance on PoseTrack21.	120
6.7	(a) Impact of different temporal person similarities used to calculate the pose similarity embeddings. (b) Impact of the proposed SPAPDE network for re-identification on the overall performance on PoseTrack21.	121
6.8	Person re-identification performance (mAP) on the PoseTrack21 dataset for various model settings.	122
6.9	Impact of the <i>New Track Embedding Head</i> on the overall tracking performance on PoseTrack21.	122

Nomenclature

Acronyms

An alphabetically sorted list of abbreviations used in the thesis:

2d/3d	Two-dimensional/Three-dimensional
AI	Artificial Intelligence
ANN	Artificial Neural Network
AP	Average Precision
CMC	Cumulative Matching Characteristic
CNN	Convolutional Neural Network
DAG	Directed Acyclic Graph
FA-HOTA	Fragmentation-aware HOTA
FFN	Feed-Forward Network
FN	False-Negative
FP	False-Positive
FragA	Fragmentation Accuracy
HOG	Histogram of Oriented Gradients
HOTA	Higher Order Tracking Accuracy
IDP	Identity Precision
IDR	Identity Recall
IDSW	Identity Switch
ILP	Integer Linear Programming
IoU	Intersection over Union
LocA	Localization Accuracy
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
mAP	Mean Average Precision
MLP	Multi-Layer Perceptron
MOT	Multi-Object Tracking
MOTA	Multi-Object Tracking Accuracy
MOTP	Multi-Object Tracking Precision
MPPT	Multi-Person Pose Tracking
MSE	Mean Squared Error
NLP	Natural Language Processing
NMS	Non-Maxima Suppression
OKS	Object Keypoint Similarity
ReID	Re-Identification
ReLU	Rectified Linear Unit
ResNet	Residual Network
RNN	Recurrent Neural Network
RPN	Region Proposal Network
SIFT	Scale-Invariant Feature Transformation
SVM	Support Vector Machine
TP	True-positive
VIT	Vision Transformer

List of Publications

The thesis is based on the following publications:

- **Self-supervised Keypoint Correspondences for Multi-Person Pose Estimation and Tracking in Videos**
Umer Rafi*, Andreas Doering*, Bastian, and Juergen Gall
IEEE European Conference on Computer Vision (ECCV), 2020.
DOI: https://doi.org/10.1007/978-3-030-58565-5_3
- **PoseTrack21: A Dataset for Person Search, Multi-Object Tracking and Multi-Person Pose Tracking**
Andreas Doering, Di Chen, Shanshan Zhang, Bernt Schiele, and Juergen Gall
IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2022.
DOI: <https://doi.org/10.1109/CVPR52688.2022.02029>
- **A Gated Attention Transformer for Multi-Person Pose Tracking**
Andreas Doering and Juergen Gall
IEEE International Conference On Computer Vision Workshop (ICCVW), 2023.
DOI: <https://doi.org/10.1109/ICCVW60793.2023.00343>

(* denotes equal contribution)

Introduction

Contents

1.1 Motivation	1
1.2 Problem Formulation	3
1.3 Contributions	4
1.3.1 Self-Supervised Keypoint Correspondences for Multi-Person Pose Tracking	4
1.3.2 PoseTrack21: A Dataset for Person Search, Multi-Object Tracking and Multi-Person Pose Tracking	4
1.3.3 A Gated Attention Transformer for Multi-Person Pose Tracking	5
1.4 Thesis Structure	5

1.1 Motivation

The field of Artificial Intelligence (AI) has rapidly advanced within the last decade, and due to vast progress in computational performance and efficiency, AI-based technologies have become an indispensable part of our daily lives. In consequence, nowadays, even smartphones can run AI-based pipelines that required heavy computational resources a few years back. The next leap forward into the future is the development and integration of systems into everyday life that facilitate a natural interaction between humans and machines. Such systems require capabilities to fully reason about their environments, including the interpretation of non-verbal communication and the anticipation of future actions a human might take within the next seconds or even minutes. Johansson (1973) demonstrated in his experiments that the movement of body keypoints is sufficient for humans to infer the activity a test person is undergoing. For a machine, such capabilities require robust computer vision approaches to estimate and track human poses and re-identify previously seen persons. A typical use case for such a system might include a humanoid service robot in elderly care that interacts with disabled persons in their daily routines. Such a robot would be required to re-identify the person of interest, anticipate their future actions, and plan the following tasks accordingly.

Furthermore, the estimation and tracking of human keypoints provide an essential baseline for several downstream tasks. In medical applications, human poses can be used for gait analysis. The observation of gait can be helpful in chiropractic and osteopathic professions to detect misalignments in the human body. Also, gait analysis can be used to recognize and verify people's identities by their pace and style of walking, which can be helpful in surveillance systems. Moreover, an intelligent surveillance system would be able to predict future actions of individual persons based on their tracked poses. It could raise alarms during or even before a critical event. Estimating future actions or movements might also be an essential trait of autonomous vehicles, which can analyze the behavior

of the surrounding pedestrians and initiate a break maneuver if necessary. Conversely, estimating the driver's pose can aid a sleep detection system that takes action if the driver falls asleep. In sports, pose information can even be used to analyze and understand the entire dynamics of a football or basketball game, i.e., estimate each athlete's performance or identify weaknesses in the opponent's team. In the film and gaming industry, multi-person pose tracking systems can be used in markerless motion capture systems that do not require body suits.

However, systems operating in complex real-world scenarios must be robust to various influencing factors, such as varying lighting and weather conditions in densely populated environments with an unknown and dynamic number of people that might occlude each other, often leading to incomplete or inaccurate poses. Situations in which people interact with each other can often lead to inaccuracies or misplaced keypoints due to hidden or overlapping body parts. Moreover, people appear at various distances from the camera and thus in varying heights and scales. Handling such variations is crucial for a robust pose estimation system. The human body can adopt a wide range of articulated poses. Robustness against highly articulated and unusual poses, often occurring in the sports domain, is crucial for robust pose estimation systems. Besides accurately estimating the keypoints of human poses, precisely associating and tracking these poses throughout a video drastically increases the complexity. Fast-moving objects, motion blur, and moving cameras impact pose estimation accuracy and can lead to temporally inconsistent tracks. Robust pose tracking systems require robust occlusion handling, as individuals can disappear for several frames or leave the scene entirely. The latter requires a robust identity preservation mechanism that re-assigns a person to the correct track once it re-appears within a scene. The re-association is far from trivial: differentiating between ambiguous poses estimated in crowded scenarios or similar appearances, such as similar clothing or due to occlusion, can lead to false assignments. Complex interactions, such as group activities or team sports, require the algorithm to accurately understand and model the spatial and temporal relationships between people.

Early works on multi-person pose tracking (Iqbal et al., 2017; Andriluka et al., 2018; Insafutdinov et al., 2017) formulate a graph partitioning problem to solve a spatio-temporal graph with integer linear programming. However, optimizing spatio-temporal graphs is computationally heavy and unsuitable for real-time applications. Thus, research (Doering et al., 2018; Girdhar et al., 2018; Xiu et al., 2018; Xiao et al., 2018) moved towards methods that employ bipartite graphs optimized using greedy or Hungarian (Kuhn, 1955) optimization. These works employ flow- and motion-based methods for assigning detected poses to tracks. Thus, flow-based methods fail if a person becomes occluded for more extended periods of time. On the other hand, appearance-based methods (Zhang et al., 2019) are prone to errors in partial occlusion or similar appearances. Moreover, limited annotations in existing pose tracking datasets (Andriluka et al., 2018) require training additional models on external datasets for person re-identification.

Thus, this thesis aims to address the abovementioned challenges, and we strive to bridge the gap between flow-based and appearance-based tracking strategies. In the first step, we develop a top-down multi-person pose estimation model and propose a keypoint correspondence network for multi-person pose tracking that allows the recovery of tracks previously broken due to occlusion. The underlying pose tracking dataset (Andriluka et al., 2018) is sparsely annotated, and we thus trained our keypoint correspondence model in a self-supervised manner on a pose estimation dataset (Lin et al., 2014). Due to the sparsity of the PoseTrack dataset, a natural next step is the proposal of a

densely annotated dataset, which we denote as PoseTrack21.

PoseTrack21 is a meticulously annotated dataset to address this gap that enriches existing annotations by incorporating detailed poses alongside consistent person identities throughout the dataset, complemented by precise bounding boxes for each person. We introduce a new problem we refer to as multi-person re-id pose tracking aimed at simultaneously tracking and re-identifying persons throughout and across multiple video sequences. Moreover, our dataset allows us to bridge the gap between multi-person pose tracking, multi-object tracking, and person search, for which we propose various baselines. In addition, we propose an online tracking method that employs a gated attention mechanism that integrates appearance and keypoint-based similarities in a novel way. Notably, the method incorporates a matching layer that allows the removal of duplicate detections, i.e., multiple detections of the same person, thus refining the tracking process. Moreover, to enhance the robustness of the appearance-based features, we introduce a pose-conditioned re-identification model. This model normalizes appearance features based on heatmaps derived from detected keypoints, effectively mitigating noise and enhancing the reliability of appearance-based re-identification.

In the subsequent section of this chapter, we outline the problem statement upon which this thesis is founded, succeeded by an overview of the contributions made within this thesis.

1.2 Problem Formulation

In this thesis, we represent a human pose as a set of K locations that define anatomical keypoints on the human body and develop methods for 2D multi-person pose estimation and tracking. We thus, describe a 2D human pose as $P = \{\mathbf{x}_j\}_{j \in K}$ where $\mathbf{x}_j = (x_j, y_j) \in \mathbb{R}^2$ denote the pixel locations in an image $\mathbf{I}$. We additionally localize and represent individual persons by a bounding box $B = (x_{tl}, y_{tl}, x_{br}, y_{br}) \in \mathbb{R}^4$ that is composed of top-left $(x_{tl}, y_{tl}) \in \mathbb{R}^2$ and bottom-right $(x_{br}, y_{br}) \in \mathbb{R}^2$ pixel locations.

Sophisticated systems for understanding human behavior are required to process unconstrained video streams of real-world data. Thus, we define a video as a sequence of F frames $\mathcal{F} = \{\mathbf{I}_f\}_{f \in [1, F]}$, where $\mathbf{I}_f$ denotes video frame f ¹. In dependence on the given task, we define a track T as a sequence of person-specific localization information and assign unique identities to each person. In multi-person pose tracking, a track $T = \{(P_f^q)\}$ is a trajectory of poses for every person q in frame f . However, in multi-object tracking, we define a track as a sequence of bounding boxes $T = \{(B_f^q)\}$.

Johansson (1973) indicated that the human visual system only requires a sparse representation to recognize human actions. We thus follow literature (Andriluka et al., 2018; Iqbal et al., 2017; Andriluka et al., 2014) and represent a pose as a sparse set of landmarks, i.e., $K \in [15, 17]$. Nonetheless, the level of detail may vary for different applications, i.e., 3D pose and human shape estimation (Tripathi et al., 2023). Throughout this thesis, we further represent a video as a sequence of RGB images.

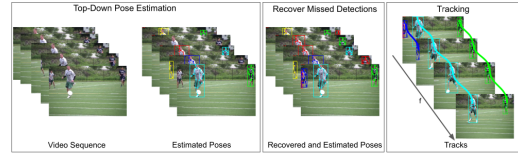
¹We interchangeably use f and t as time-dependent indices of chronological order of frames or tracks.

1.3 Contributions

In this thesis, we propose the following contributions towards multi-person pose tracking.

1.3.1 Self-Supervised Keypoint Correspondences for Multi-Person Pose Tracking

As our first contribution, we propose an approach for multi-person pose tracking that relies on keypoint correspondences for the association of persons in videos, which we trained in a self-supervised manner. Existing multi-person pose estimation and tracking datasets are less diverse and more sparsely annotated than large-scale image datasets for human pose estimation. Consequently, it is challenging to learn deep learning-based models for associating keypoints across frames that are robust to nuisance factors such as motion blur and occlusions for multi-person pose tracking. Thus, instead of training the network to estimate keypoint correspondences on video data, it is trained on a large-scale image dataset for human pose estimation using self-supervision. Combined with a top-down framework for human pose estimation, we use keypoint correspondences to (i) recover missed pose detections and to (ii) associate pose detections across video frames. Our approach achieves an accuracy similar to competing methods of multi-frame pose estimation and multi-person pose tracking on the PoseTrack 2017 and 2018 datasets. This chapter is based on the work published in [Rafi et al. \(2020\)](#).



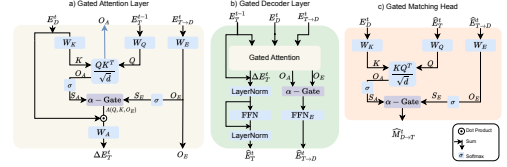
1.3.2 PoseTrack21: A Dataset for Person Search, Multi-Object Tracking and Multi-Person Pose Tracking

As a next step, we aim to address the sparsity of multi-person pose tracking datasets. We introduce PoseTrack21, a large-scale dataset with high diversity in pose that provides rich annotations like human pose annotations, including annotations of joint occlusions, bounding box annotations even for small persons, and person-ids within and across video sequences. Current research evaluates person search, multi-object tracking, and multi-person pose estimation as separate tasks on different datasets. However, these tasks are very akin to each other and comprise similar sub-tasks, e.g., person detection or appearance-based association of detected persons. Consequently, approaches to these respective tasks are eligible to complement each other. Therefore, PoseTrack21 allows us to evaluate multi-object and multi-person pose tracking jointly with person re-identification or exploit structural knowledge of human poses to improve person search and tracking, particularly in severe occlusions. With PoseTrack21, we want to encourage researchers to work on joint approaches that perform reasonably well on all three tasks. This chapter is based on the work published in [Doering et al. \(2022\)](#).



1.3.3 A Gated Attention Transformer for Multi-Person Pose Tracking

Despite the advancements in online tracking methods, the association of poses across frames remains a challenging research problem, primarily due to motion blur, crowded scenes, and occlusion. Thus, our last work focuses on the association challenge and proposes a Gated Attention Transformer. The core aspect of our model is the gating mechanism that automatically adapts the impact of appearance embeddings and embeddings based on temporal pose similarity in the attention layers. To re-identify occluded persons, we incorporate a pose-conditioned re-identification network that provides initial embeddings and allows the matching of persons even if the number of visible joints differs between frames. We propose a matching layer based on gated attention for pose-to-track association and duplicate removal. We evaluate our approach on PoseTrack 2018 and PoseTrack21. This chapter is based on the work published in [Doering and Gall \(2023\)](#).



1.4 Thesis Structure

The remainder of this thesis follows a structured organization, as detailed below. The upcoming chapter will present a comprehensive literature review on 2D (multi-person) human pose estimation, multi-person pose tracking, object detection, multi-object tracking, and person search. The third chapter will delve into the fundamental concepts on which this thesis is founded. Chapters 4-6 will detail the contributions outlined in Section 1.3.

Chapter 2 offers a comprehensive review of the literature on 2D human pose estimation, covering single and multi-person pose estimation, multi-frame multi-person pose estimation, and multi-person pose tracking. It then delves into a detailed overview of the literature on object detection and multi-object tracking. After that, it discusses the state-of-the-art in person search and person re-identification and concludes with an overview of the common datasets used in the literature for the abovementioned topics.

Chapter 3 presents a brief introduction to the fundamental concepts crucial for understanding the developed methods proposed in this thesis.

Chapter 4 introduces a pipeline that tracks the poses of multiple persons. The pipeline is built on a keypoint correspondence network trained in a self-supervised setting and combined with a top-down framework for human pose estimation. The main objective of this pipeline is to recover missed detections and associate estimated poses across video frames.

Chapter 5 proposes PoseTrack21, a new dataset for multi-person pose tracking. The dataset comprises comprehensive annotations, including human pose annotations, joint occlusions, bounding box annotations, and unique person identities. The dataset allows a joint evaluation of various tasks such as person search, multi-object tracking, and multi-person pose tracking. Chapter 5

proposes baseline methods for each task and includes extensive experimental evaluations.

Chapter 6 presents a Gated Attention Transformer model that addresses the association challenge in multi-person pose tracking. The model’s gating mechanism adapts appearance embeddings and temporal pose similarity. Additionally, it incorporates a pose-conditioned re-identification network and a matching layer based on gated attention for improved association and duplicate removal.

The final section of this thesis is **Chapter 7**, which provides a conclusion and suggests directions for future research.

Related Work

Contents

2.1	2D Pose Estimation	8
2.1.1	Single Person Pose Estimation	8
2.1.2	Multi-Person Pose Estimation	9
2.1.3	Multi-Frame Multi-Person Pose Estimation and Tracking	17
2.2	Object Detection and Multi-Object Tracking	24
2.2.1	Object Detection in MOT	25
2.2.2	Multi-Object Tracking	26
2.3	Person Search and Person Re-Identification	33

Over the past decades, researchers have devoted considerable attention to visual understanding of the human body. This field has given rise to various downstream applications, including human activity recognition, gait analysis, and the anticipation of human behavior. One of the earliest scholars to conceptualize the human body as a collection of keypoints was the psychologist Johansson (Johansson, 1973). Through his work, Johansson demonstrated that the motion of a subset of human body keypoints is enough to accurately identify human activities like walking and running. His work unleashed a considerable interest in the computer vision community (O’Rourke and Badler, 1980; Hogg, 1983; Chen and Lee, 1992; Qian and Huang, 1992) and led to the emergence of several research directions related to human body understanding, including human pose estimation (Niyogi and Adelson, 1994; Andriluka et al., 2009; Cao et al., 2017; Sun et al., 2019a; Xu et al., 2022), human pose tracking (Rohr, 1993; Bregler, 1997; Ramanan et al., 2005; Doering et al., 2018; Wang et al., 2020c) and activity recognition (Polana and Nelson, 1993; Iqbal et al., 2016; Hejazi and Abhayaratne, 2022).

The literature on multi-person pose estimation and tracking is enormous and diverse. This thesis addresses the problems of 2D human pose estimation and 2D multi-person pose tracking in RGB video frames from single-camera views. The domain of multi-person pose tracking encompasses several issues addressed separately in the literature. Despite their differences, these problems share common characteristics and can be interrelated. Many multi-person pose tracking approaches can easily transfer into the multi-object tracking domain and vice versa. Furthermore, common works on multi-person pose tracking incorporate re-identification components to aid the associative capabilities of the tracking methods. To provide a comprehensive overview of the current research landscape, this thesis includes a compact review of the state-of-the-art literature concerning multi-object tracking, person search, and person re-identification, particularly on multi-person pose tracking.

2.1 2D Pose Estimation

The task of estimating 2D human pose from images and videos presents a significant challenge due to a multitude of factors. These include occlusions, truncation, highly deformed poses, and other unusual poses, especially in unconstrained real-world scenarios. Additionally, varying appearances, such as clothing, illumination changes, and motion blur, can pose significant challenges to pose estimation pipelines. Furthermore, pose estimation models must be able to differentiate between human and non-human poses, such as mannequins, and distinguish between photographed individuals (e.g. advertisements in the background) and real people in a scene. Furthermore, highly cluttered or changing backgrounds can introduce significant noise into the system, complicating the estimation process. These challenges require careful consideration and sophisticated approaches in order to achieve accurate and reliable 2D human pose estimation. For these reasons, pose estimation has been studied in very constrained scenarios (e.g. on a single, fully visible person) for decades (Section 2.1.1). With the advancement of deep convolutional networks, research shifted towards pose estimation in unconstrained real-world scenarios with multiple-person instances.

This chapter reviews the literature on single-person pose estimation approaches in Section 2.1.1, followed by coverage of state-of-the-art works on multi-person pose estimation in Section 2.1.2. Finally, Section 2.1.3 concludes this section with a comprehensive overview of multi-person pose estimation and tracking in videos.

2.1.1 Single Person Pose Estimation

Early methods for single-person human pose estimation predominantly utilized part-based approaches. These traditional approaches often depended on hand-crafted low-level features (Tompson et al., 2014), such as Histogram of Oriented Gradients (HOG) (Dalal and Triggs, 2005) or Scale-Invariant Feature Transformation (SIFT) (Lowe, 1999), which were then fed into higher generative models or standard classifiers like AdaBoost (Schapire, 1999), Support Vector Machine (SVM) (Steinwart and Christmann, 2008), or random forests (Criminisi and Shotton, 2013). In a bottom-up manner, body part detectors independently estimated the probability of body joint presence at the pixel level, as seen in (Dantone et al., 2014). Beyond these unary joint potentials, traditional models extensively incorporated graphical structures (Chen and Yuille, 2014; Ouyang et al., 2014; Yang and Ramanan, 2011) or loopy models (Ferrari et al., 2009; Ren et al., 2005; Daphne Tsatsoulis and Forsyth, 2015) to represent human pose. These graphical models captured spatial dependencies between individual body parts. Among these methods, the pictorial structure model (Fischler and Elschlager, 1973), initially introduced by Fischler and Elschlager in 1978 and later refined for tree structures by Felzenszwalb and Huttenlocher (2005), gained significant popularity (Johnson and Everingham, 2010; Yang and Ramanan, 2011; Dantone et al., 2014; Ouyang et al., 2014; Johnson and Everingham, 2011). This approach achieved optimal inference by leveraging distance transforms (Johnson and Everingham, 2011).

A common limitation of part-based models is their restricted ability to represent only a limited subset of possible relations between body parts (Toshev and Szegedy, 2014). To enhance robustness, some methods (Pishchulin et al., 2013a,b) aimed to model higher-order part relations (Tompson et al., 2014), for example, by augmenting the pictorial structure model with poselet (Bourdev and Malik, 2009) priors. Despite these advancements, all traditional approaches relied on hand-crafted features

tailored to specific human body parts. This resulted in weak part templates or fragile body part representations (Ouyang et al., 2014).

To address the limitations of traditional models, research shifted towards deep convolutional neural networks for human pose estimation (Ouyang et al., 2014; Toshev and Szegedy, 2014), demonstrating superior performance and surpassing traditional methods such as those in (Dantone et al., 2014; Yang and Ramanan, 2011; Johnson and Everingham, 2011; Wang and Li, 2013). With advancements in deep learning, the research focus expanded to multi-person pose estimation, where top-down methods (Section 2.1.2.1) essentially deploy advanced single-person pose estimation networks. those

2.1.2 Multi-Person Pose Estimation

Multi-person pose estimation is a technique that aims to estimate the pose of multiple individuals in an image captured in unconstrained scenarios, where different levels of truncation and occlusion are common. Unlike single-person pose estimation, which focuses on estimating the pose of a single person in an image, multi-person pose estimation methods have to deal with unusual poses, such as those encountered in sports where the persons of interest may appear upside down or lying on the ground. Furthermore, varying lighting conditions, motion blur, and crowded scenes further complicate the task. Works on multi-person pose estimation can be classified into two main categories: bottom-up and top-down methods.

Bottom-up methods, such as those proposed by Cao et al. (2017) and Newell et al. (2017), directly predict and group the keypoints of all individuals into poses. One of the main advantages of these approaches is their fast runtime, which allows for real-time applications. However, they often suffer from high variations in scale and can exhibit degenerate performance in large and small instances. Furthermore, occlusions can lead to invisible keypoints, posing a challenge in assembling individual keypoints into poses. In recent years, a new category of bottom-up approaches emerged in the literature called single-stage methods (Duan et al., 2019; Zigang et al., 2021; Artacho and Savakis, 2023; Nie et al., 2019b). The primary objective of these methods is to directly detect instance-aware keypoints by utilizing estimated root locations.

In contrast, top-down approaches, like the one proposed by Sun et al. (2019a), are applied on fixed-size image crops and assume each crop contains a single person at its center. Generally, these approaches perform better than bottom-up methods because they operate on a fixed resolution. Nevertheless, top-down approaches require a person detector that first localizes all individuals in the image. Therefore, the performance of these approaches is highly limited by the underlying object detector, which can miss a person. Particularly in crowded scenes, it is hard to detect people bounding boxes accurately. Furthermore, the runtime of top-down approaches is highly impacted and scales linearly with the number of detected person instances in an image, depending on the inference speed of the underlying person detector.

2.1.2.1 Top-down methods

The first approaches towards multi-person pose estimation were top-down methods, which emerged in the early 2010s. In particular, Eichner and Ferrari (2010) was one of the pioneers, developing one of the first techniques for multi-person pose estimation that prioritized addressing person-to-person

occlusion. To achieve this, Eichner introduced occlusion priors and proposed a joint multi-person Pictorial Structure model (PSM) that was highly effective. Meanwhile, Pishchulin et al. (2012) put forward a pictorial structure deformable parts model (PS-DPM) that was trained in a similar manner to Andriluka et al. (2012) and evaluate the pose of every person independently.

Building on N-best inference algorithms (Nilsson, 1998), Yang and Ramanan (2013) extends the work of Park and Ramanan (2011) and proposes a N-best maximal algorithm that does not require a person detector. This approach generates N-best pose candidates that are different in scale or have root joints that are sufficiently far apart. Ladicky et al. (2013) further extends this line of work and incorporates pixel-wise labeling methods to perform instance and body-part segmentation, thereby eliminating implausible candidates, e.g. if estimated keypoints do not agree with estimated foreground pixels. Moreover, this method can handle partially occluding instances and self-occlusion. All these methods rely on hand-crafted features (e.g. HOG (Hogg, 1983)) and traditional concepts of machine learning, such as random forests (Breiman, 2001), boosting (Kearns, 1988), structured SVM (Xue et al., 2008) and perform inferior to CNN-based and transformer-based architectures.

Deep-Learning based methods With the emerging success of deep neural networks, the landscape of multi-person pose estimation approaches has changed drastically. One of the first works towards this direction was proposed by Iqbal et al. (2016). Iqbal et al. generate a set of keypoint candidates for every detected person bounding box using Convolutional Pose Machines (Wei et al., 2016) and assemble these keypoints into a pose by solving a densely connected graph following Pishchulin et al. (2016). Haoshu Fang and Lu (2016) propose a framework that extends single-person pose estimation methods (i.e. Newell et al. (2016) or Wei et al. (2016)) to multi-person pose estimation. Since single-person pose estimation methods are highly sensitive to localization errors, Fang et al. introduce a spatial transformer network (STN) that centers the person of interest before passing the input to the pose estimation model. Additionally, they propose a spatial de-transformer network (SDTN) to reverse the transformation after pose estimation.

Many top-down methods (Xia et al., 2017; Papandreou et al., 2017; Xiao et al., 2018; Sun et al., 2019a) follow a standard pipeline and utilize Faster-RCNN (Ren et al., 2015) as a person detector and a ResNet (He et al., 2016) or HRNet (Sun et al., 2019a) as pose estimation backbone. Pose estimation is applied in a fully-convolutional manner similar to (Pishchulin et al., 2016; Insafutdinov et al., 2016) to estimate keypoint heatmaps directly. Other methods (Sun et al., 2018b; Li et al., 2021a; Gu et al., 2021; Mao et al., 2022) propose to regress the keypoint locations instead directly.

Xia et al. (2017) propose to jointly estimate poses and semantic body parts and propose an architecture based on (He et al., 2016; Insafutdinov et al., 2016) for human pose estimation and follow Chen et al. (2015) for the semantic part segmentation task. Unlike Chen et al. (2015), their fully-connected conditional random field (FCRF) is applied to the bounding box crops instead of the full image to reduce complexity. They combined initial semantic part score maps, joint heatmaps, and joint neighbor offset maps within an FCRF to estimate the final pose. Papandreou et al. (2017) further simplify the pose estimation pipeline, predicting keypoint offset predictions and keypoint heatmaps to compensate for scale-based inaccuracies due to the network stride. They follow Chen et al. (2015) and propose a modification to the ResNet101 backbone and incorporate atrous convolution to reduce the output stride from 32 pixels to 8 pixels. The proposed offsets compensate for the remaining stride and produce more accurate keypoint predictions. Xiao et al. (2018) follow a similar pipeline

and propose a decoding layer composed of transposed convolutions that reduce the output stride to 4 pixels. In contrast to Papandreou et al. (2017), their method achieves state-of-the-art results without additional keypoint-offset regression. Zhang et al. (2019) combine both ideas of (Papandreou et al., 2017; Xiao et al., 2018) and propose a pose estimation model based on Xiao et al. (2018) with an additional keypoint-offset regression head similar to Papandreou et al. (2017).

A different path of works (Chen et al., 2018b; Su et al., 2019; Li et al., 2019b) explore multi-level features, coarse-to-fine supervision, and cross-stage feature aggregation for increased robustness against crowded backgrounds. Chen et al. (2018b) address the localization of "hard" keypoints which can not be localized due to occlusion or crowded backgrounds. Chen et al. (2018b) argue that these keypoints can only be localized from the context of estimated keypoints, but not from appearance features only. For that reason, they propose two sub-networks: GlobalNet and RefineNet. GlobalNet is based on FPN (Lin et al., 2017a), extracts robust context features, and provides initial keypoint estimates. The RefineNet integrates information from different feature levels similar to Kong et al. (2016), focussing on detecting hard keypoints by employing an online hard keypoints mining strategy. Building upon the work of Chen et al. (2018b), Su et al. (2019) aim to address the inherent trade-offs between low-level and high-level feature maps in pose estimation by recalibrating their interdependencies. To achieve this, they upsample and concatenate features derived from the FPN, followed by a shuffling operation as proposed by Zhang et al. (2017). Additionally, Su et al. (2019) extend the residual bottlenecks in the RefineNet with channel-wise attention. This adjustment allows the network to suppress irrelevant regions in the feature maps while emphasizing key areas critical for accurate keypoint localization.

Li et al. (2019b) adopt GlobalNet from Chen et al. (2018b). They propose aggregating features across different stages to increase information flow. They also adopt the coarse-to-fine supervision strategy, as pose localization accuracy is gradually refined during multi-stage training. A multi-stage network is vulnerable to information loss during repeated up-and-down sampling. Therefore, they use a cross-stage feature aggregation strategy to efficiently propagate multi-scale features from early to current stages. Sun et al. (2019a) follow a similar idea as Li et al. (2019b) but propose a parallel multi-stage network with repeated multi-scale fusion across stages, such as high-to-low and low-to-high feature fusion. This leads to rich, high-resolution representations and spatially precise keypoint heatmaps.

Cai et al. (2020) argue that there is a big gap between the receptive fields during multi-scale feature fusion, which leads to coarse representations and lower localization accuracies. The authors propose a multi-stage architecture similar to (Li et al., 2019b; Su et al., 2019) emphasizing intra-level feature fusion. Motivated by Huang et al. (2017), their proposed Residual Steps Block (RSB) aims to fuse features with the same spatial dimensions but different receptive fields within the same network stage. Similar to Su et al. (2019), this paper proposes an attention mechanism for further refinement and a better trade-off between local and global representations.

The current state-of-the-art methods (Sun et al., 2019a; Xiao et al., 2018) for human pose estimation suffer from inaccuracies due to quantization effects caused by network output strides of up to 32 pixels. To improve localization accuracies, sub-pixel shifting (Newell et al., 2016; Sun et al., 2019a) is a commonly used technique. Alternatively, some works (Papandreou et al., 2017; Zhang et al., 2019) aim to regress keypoint offsets within their proposed models directly, and other works (Sun et al., 2018b; Li et al., 2021a; Gu et al., 2021) aim to regress keypoint locations directly. For instance,

[Sun et al. \(2018b\)](#) propose integral regression that combines the benefits of heatmap prediction and pose regression methods to circumvent quantization errors and non-differentiable post-processing. Nonetheless, this method still comprises an underlying bias that affects the pose estimation performance compared to methods based on direct regression, as shown by [Gu et al. \(2021\)](#). [Li et al. \(2021a\)](#) is the first regression-based method that achieves performance comparable with heatmap-based approaches. In their method, referred to as residual log-likelihood estimation (RLE), the authors propose to leverage normalizing flows ([Dinh et al., 2017](#)) to re-parameterize the predicted output distribution of human keypoints.

While most methods focus on enhancing keypoint localization techniques, recent studies ([Huang et al., 2020](#); [Zhang et al., 2020a](#)) have highlighted that state-of-the-art data processing techniques significantly degrade localization performance due to biased data augmentation ([Huang et al., 2020](#)) and suboptimal heatmap decoding ([Zhang et al., 2020a](#); [Huang et al., 2020](#)). To address these issues, [Huang et al. \(2020\)](#) and [Zhang et al. \(2020a\)](#) propose techniques that can be integrated into existing pose estimation pipelines.

The standard data processing pipeline for human pose estimation involves two problems that significantly degrade pose estimation performance. As demonstrated by [Huang et al. \(2020\)](#), augmentation techniques such as cropping, resizing, rotation, or flipping can lead to unaligned results during inference. To compensate for this degradation in performance caused by induced biases, [Huang et al. \(2020\)](#) propose an unbiased data processing (UDP) system that includes an unbiased coordinate system transformation and an unbiased keypoint format transformation. While the former targets data augmentation, it specifies transformations for keypoint heatmap encoding and decoding. The keypoint decoding follows [Zhang et al. \(2020a\)](#), who propose a distribution-aware coordinate presentation for keypoints to increase localization accuracy. This method involves finding the location of a keypoint by analyzing the first derivative of a Gaussian distribution, i.e. where the first derivative equates to zero.

Multi-Person Pose Estimation in Crowds The estimation of poses in densely crowded scenarios is still a challenging task for current state-of-the-art methods. For that reason, multiple works ([Li et al., 2019a](#); [Qiu et al., 2020a](#); [Khironkar et al., 2021](#); [Ding et al., 2022](#)) dedicate their effort to improving multi-person pose estimation in crowded scenarios. Multiple datasets ([Li et al., 2019a](#); [Qiu et al., 2020a](#)) have been released to evaluate those methods better.

One of the major problems of densely crowded scenarios is the erroneous performance of person detectors, which often miss a bounding box. Further, keypoints of different person instances are often closely located, which increases the difficulty and often leads to misplacements. A common way ([Li et al., 2019a](#); [Khironkar et al., 2021](#)) is to extend existing pose estimation networks to directly estimate the keypoints of multiple instances within a single bounding box, as this formulation allows to compensate for missing detections. [Li et al. \(2019a\)](#) build on top of [Fang et al. \(2022\)](#) and propose a global association scheme that generates a Person-Joint Graph and removes redundant keypoint detections accordingly. [Khironkar et al. \(2021\)](#), on the other hand, propose a Multi-Instance Modulation Block that can be integrated into existing pose estimation architectures that condition feature responses by a scalar instance selector. In contrast to [Li et al. \(2019a\)](#), this method directly estimates instance-related keypoints.

A different approach ([Qiu et al., 2020a](#); [Ding et al., 2022](#)) is to refine the detected poses based on

their inter-pose relations. The approach of Qiu et al. (2020a) is inspired by the human capability of inferring occluded poses due to abundant prior knowledge. Therefore, the authors propose an image-guided progressive GCN to correct an initially estimated pose. In the second stage, the underlying pose graph is extended into a couple graph that modulates relationships between poses. Ding et al. (2022), on the other hand, propose a transformer network (Vaswani et al., 2017) to model intra-human and inter-human relations based on pooled features from a pose estimation network (e.g. Sun et al. (2019a)).

Transformer-based methods With the success of the (vision-) transformer (Vaswani et al., 2017; Dosovitskiy et al., 2021; Liu et al., 2021a), researchers shifted their attention to transformer-based method for human pose estimation (Yuan et al., 2021; Li et al., 2021c; Yang et al., 2021a; Xu et al., 2022; Mao et al., 2022) and achieve superior performance on common benchmarks (i.e. Lin et al. (2014)). Due to the high demand in memory and computational cost of transformer-based models, most works rely on CNN backbones or carefully designed attention mechanisms. In the literature, there are multiple categories of approaches to the task of transformer-based pose estimation. A common method follows the standard encoder-decoder transformer architecture (Vaswani et al., 2017) as proposed in (Mao et al., 2022; Li et al., 2021b). These works decode keypoint tokens (i.e. one token per keypoint class) from which the keypoint locations are directly regressed. For instance, Li et al. (2021b) re-purpose DETR (Carion et al., 2020), an end-to-end transformer for object detection, and estimate and refine the keypoints in a cascade manner. Mao et al. (2022) extend the work of Li et al. (2021a) and propose an encoder-decoder architecture to directly regress the keypoint locations based on residual log-likelihood estimation (Li et al., 2021a).

Yang et al. (2021a) argue that the estimation of keypoint heatmaps is essentially an encoding task and employs only the classic transformer encoder from Vaswani et al. (2017) to model long-range dependencies of predicted keypoint heatmaps. Compared to (Yuan et al., 2021; Mao et al., 2022), dropping the transformer-decoder reduces the computational cost remarkably. Most recent works (Xu et al., 2022; Li et al., 2021c; Yang et al., 2021b) follow this transformer-encoder-only architecture and further aim to optimize the computational cost by patchifying the image (Xu et al., 2022) or the image features (Li et al., 2021c) into visual tokens or apply local attention on non-overlapping regions in the feature space (Yang et al., 2021b). The proposed work of Yang et al. (2021b) follows the architecture of HRNet (Sun et al., 2019a) and utilizes transformer encoders to extract high-resolution features. Moreover, the authors propose a parallel multi-resolution transformer with local window attention that gradually fuses high- and low-level features. Li et al. (2021c) prepend learnable keypoint tokens to patchified feature tokens extracted from a CNN backbone (Xiao et al., 2018; Sun et al., 2019a). An MLP then predicts a heatmap for every keypoint token individually. In contrast to these works, Xu et al. (2022) propose a plain vision transformer with a simple yet effective architecture. Their model computes visual tokens by patchifying the input image. After several transformer-encoder stages, the image tokens are re-assembled into a 2D image, and similar to Xiao et al. (2018), a simple fully-convolutional decoder consisting of transposed convolution layers is utilized to estimate keypoint heatmaps.

2.1.2.2 Bottom-up Methods

The objective of bottom-up approaches is to estimate the positions of keypoints for all person instances in an image and subsequently construct individual poses. A supplementary set of relational characteristics between the keypoints is necessary, such as affinity fields (Cao et al., 2017) or associative embeddings (Newell et al., 2017). These methods are classified as two-stage approaches due to their methodology in recent literature. In contrast, single-stage approaches directly estimate the pose representations with instance awareness, e.g. these methods estimate instance-aware center locations and regress the keypoints as offsets to the person’s center location (Zhou et al., 2020b; Zigang et al., 2021; Artacho and Savakis, 2023; Nie et al., 2019b).

Two-Stage Methods The earliest studies on bottom-up pose estimation (Pishchulin et al., 2016; Insafutdinov et al., 2016) propose a fully connected graph of detected keypoint locations to be solved using iterative linear programming (ILP). Pishchulin et al. (2016) employ body part detectors (Pishchulin et al., 2013b) in combination with (Girshick, 2015) to create pair-wise terms for every pair of distinct body part candidates. On the other hand, Insafutdinov et al. (2016) suggested a more robust body-part detector based on He et al. (2016), which further regresses pair-wise relationships between body parts. Despite being up to 5 times faster than Pishchulin et al. (2016), neither of the methods is practical for real-time scenarios.

To overcome the limitations of previous pose estimation methods, recent works have utilized convolutional neural networks (CNNs) to predict stronger pair-wise potentials and employ greedy optimization techniques to achieve more accurate and efficient results (Cao et al., 2017; Newell et al., 2017; Papandreou et al., 2018; Li et al., 2020a; Kreiss et al., 2019). These methods differ primarily in their choice of the backbone network and conceptual design of pair-wise potentials. Among these works, two major categories have emerged. The first category involves predicting various representations of body parts, such as vector fields (Cao et al., 2017; Papandreou et al., 2018; Kreiss et al., 2019; Ning and He, 2017) or elliptical heatmaps (Li et al., 2020a), and accumulating pair-wise potentials along the edge between pairs of detected keypoints. For instance, Cao et al. (2017) and Ning and He (2017) extend Wei et al. (2016) by adding an extra branch in each stage of their network to predict part-affinity-fields (PAFs), which are vector fields consisting of unit vectors between two keypoints directed towards the target keypoint. Ning and He (2017) further incorporate dual-path blocks that are related to keypoints and PAFs. Similarly, Papandreou et al. (2018) use mid-range offsets as pair-wise potentials and regress short-range offsets in combination with keypoint heatmaps to refine keypoint locations based on Hough voting and generate instance segmentation masks based on the predicted poses. Kreiss et al. (2019) propose fusing keypoint heatmaps with a keypoint regression, denoted as part intensity fields (PIF). In contrast to Papandreou et al. (2018), PIF additionally comprise keypoint-specific scale and spread parameters that lead to more accurate keypoint localizations. Kreiss et al. additionally generate part association fields that predict a confidence score and two vectors pointing to the respective keypoints connected along the predicted limb, as well as two width parameters that control the localization variance of both connected keypoints. Qiu et al. (2020b) build upon Kreiss et al. and propose an iterative dynamic graph convolutional network that models contextual relations among keypoints. To compensate for occlusions and high variation in poses, the graph’s structure is dynamically changed in each iteration during training. Li et al. (2020a) extend

the work of Newell et al. (2017) and employ a channel-attention mechanism (Hu et al., 2018) across different stages of their network. Their work further follows Cao et al. (2017) for the association of keypoints into poses, but instead of part-affinity fields, their network predicts elliptic heatmaps. A slightly different approach is proposed by Nie et al. (2018), who include a dense regression of keypoint offsets to link predicted keypoints to human centers. Based on the predicted human centers, keypoints are partitioned into local clusters and partitioned into poses based on greedy optimization. The second category of works (Newell et al., 2017; Cheng et al., 2020; Luo et al., 2021; Jin et al., 2020) emerged from Newell et al. (2017). In their work, Newell et al. propose associative embeddings that assign similar feature descriptors to estimated keypoints of the same person instance. This allows clustering all keypoints into respective poses easily. Due to its simplicity, many works incorporated associative embeddings into their pipeline and mostly differ in the design choice of their neural network architecture. For instance, Newell et al. (2017) build upon a stacked hourglass network (Newell et al., 2016), whereas Cheng et al. (2020) extend an HRNet (Sun et al., 2019a) to generate multi-resolution and high-resolution heatmaps, which they refer to as HigherHRNet.

Bottom-up approaches are known to be sensitive to scale variations (Cheng et al., 2020; Wei et al., 2020; Kocabas et al., 2018). Related works employ multi-level features to compensate for performance degradation related to varying scales among person instances. During inference, a straightforward post-processing method (Cheng et al., 2020; Wei et al., 2020) is to evaluate the input image at multiple resolutions and estimate poses based on the averaged keypoint heatmaps. On the network side, multiple methods employ feature pyramid networks (FPN) (Kocabas et al., 2018; Wei et al., 2020), multi-level feature fusion (Cheng et al., 2020; Luo et al., 2021) or intermediate supervision (Cao et al., 2017; Luo et al., 2021; Cheng et al., 2020; Newell et al., 2017) to incorporate a certain degree of scale invariance into the neural network. For instance, Kocabas et al. (2018) builds upon RetinaNet (Lin et al., 2017b) that was originally proposed for object detection tasks. RetinaNet is comprised of a feature pyramid network and employ a multi-task architecture. Kocabas et al. (2018) extends the multi-task architecture and combine detection, pose estimation, and segmentation of persons. Their method consists of a two-branch network with a shared backbone. Each branch is a feature pyramid network that predicts human bounding boxes or keypoint heatmaps, respectively. Based on the detected bounding boxes, the predicted keypoint heatmaps are cropped, resized, and passed to an additional pose residual network (PRN) that refines and disambiguates the final pose for every person instance. In all the previous methods, the standard derivation of the Gaussian kernels used in keypoint heatmaps is fixed and can not adapt to different spatial sizes. Luo et al. (2021) argue that such heatmaps can cause semantic confusion and further do not incorporate labeling ambiguities and propose scale-adaptive and weight-adaptive heatmap regression as guidance during the training process. In particular, an additional network head predicts scale maps used to re-scale the ground-truth heatmaps during training. In contrast, the weight-adaptive heatmap regression loss suppresses the impact of negative samples.

Single-Stage Methods The post-processing step of grouping keypoints into poses is not differentiable and requires several hyperparameters. To circumvent the tuning of hyperparameters and post-processing routines, recent works propose single-stage methods that incorporate the keypoint clustering step into their network architecture in an end-to-end fashion. A common baseline in literature incorporates the detection of center locations for each person instance, which is then associ-

ated with all detected keypoints corresponding to the same instance. Towards this end, many works (Zhou et al., 2020b; Zigang et al., 2021; Artacho and Savakis, 2023; Nie et al., 2019b) aim to associate center locations and keypoints by regressing offsets from the center location to all associated keypoints, rely upon attention-based associations (Brasó et al., 2021) or propose instance-aware operations (Mao et al., 2021). For instance, Nie et al. (2019b) regresse dense keypoint displacements from the center locations. In contrast, other works (Zigang et al., 2021; Zhou et al., 2020b; Artacho and Savakis, 2023) additionally employ keypoint heatmaps to refine the regressed keypoint locations. For instance, Zhou et al. (2020b) propose two prediction heads that predict keypoint heatmaps, including the center locations and keypoint offsets, respectively. Zigang et al. (2021) argue that directly regressed keypoints are spatially inaccurate, particularly in direct comparison with two-stage methods. Therefore, the authors propose a disentangled keypoint regression (DEKR) module that employs adaptive convolutions within a multi-branch structure. In particular, the DEKR module consists of multiple regression heads that predict offsets for each keypoint type individually based on disentangled backbone features, e.g. based on non-overlapping feature groups. Artacho and Savakis (2023) closely follow the idea of Zigang et al. (2021) and additionally incorporate multi-scale features into their disentangled adaptive regression heads, but in contrast, their work does not employ multiple offset heads for each keypoint class. Mao et al. (2021) directly incorporate instance-aware convolutions into their network that dynamically predict and refine the keypoint for every person instance individually. Due to dynamic convolutions, the underlying architecture requires fewer parameters and achieves real-time performance in a fully convolutional manner. Brasó et al. (2021) and Shi et al. (2021) employ implicit instance awareness and propose transformer architectures that associate detected center locations with predicted keypoints. In particular, Brasó et al. (2021) process CNN features from a backbone network (Cheng et al., 2020) in a transformer encoder layer for every detected center and keypoint location and propose a keypoint grouping module. The grouping module contains a classification head that classifies the validity of each detected center location and a multi-head attention (MHA) layer that associates centers and keypoints in a fully-differentiable manner. In contrast, Shi et al. (2022) do not rely on a pre-trained pose estimator and utilize a transformer architecture with two decoders that estimate and refine poses in a cascaded manner. First, the pose decoder employs deformable attention with reference points that serve as initial pose locations concerning each pose query. Multiple pose-decoder stages further refine the respective pose locations, and the additional joint-decoder stages utilize joint queries to decode refined poses.

Single-stage methods that regress keypoint offsets from the center of a person often suffer from localization inaccuracies due to extensive structural variations in human poses. However, Xiao et al. (2022), Xue et al. (2022) and Wang et al. (2021a) directly address this issue through their proposed adaptive body-part regression (Xiao et al., 2022), keypoint-offset refinement (Xue et al., 2022), and individual body part keypoint estimation with further refinement techniques (Wang et al., 2021a). In (Xiao et al., 2022), an adaptive body-part regression approach is proposed, which includes a two-hop regression scheme to address the issue of high structural variations in human pose. The proposed model first regresses offsets from the person center to implicitly learned body part locations, such as the head and left arm. In the second stage, keypoint offsets are calculated from the respective body part locations. The authors argue that their model is less affected by large displacements from the center and keypoint locations, as they are avoided. Xue et al. (2022) estimate initial poses following the standard approach by predicting keypoint and center heatmaps and regressing keypoint offsets

from the center locations. Local features around each keypoint are extracted within an 11x11 grid to generate keypoint attraction maps (KAMs), which the authors call convolutional message passing. In a final step, these KAMs are utilized to refine the global keypoint heatmaps. Wang et al. (2021a) propose a method that divides each person into a pre-defined number of body parts. Unlike previous works, this approach estimates offsets to a person's keypoints with respect to each body part individually, allowing it to handle severe occlusion cases where a person's center location can be hidden. A dynamic graph convolution network is used to refine the estimated poses by incorporating additional keypoint features that encode global person information. The final pose fusion fuses duplicate poses generated from multiple body parts.

Real-Time Pose Estimation on Edge Devices In their recent work, Wang et al. (2022c) propose an optimized and efficient pose estimation framework suitable for edge-devices and achieves real-time performance. This proposed framework builds upon Cheng et al. (2020) and incorporates MobileNetV2 (Sandler et al., 2018). In their experiment, the authors gradually shrank the architecture by removing several blocks from the base network, resulting in a single-stage architecture. The high-resolution fusion block from Cheng et al. (2020) incorporates deconvolution layers and refinement blocks to fuse features from multiple resolutions. However, this high-resolution block contains redundant refinement blocks, which Wang et al. (2022c) proposed to remove to boost performance.

2.1.3 Multi-Frame Multi-Person Pose Estimation and Tracking

Multi-person pose tracking not only aims to detect the poses of multiple persons in a sequence of video frames but also tracks the pose of each individual person throughout the entire sequence and potentially throughout multiple sequences. This task increases the complexity and comes with additional challenges that must be considered. The association of estimated poses throughout time is complicated when individuals are occluded, partially visible, or not even present for an unknown amount of frames. Furthermore, interactions between individuals cause persons to merge (occlusion) or separate temporarily. These interactions can create ambiguity in the tracking process, causing different tracks to break or merge and require sophisticated algorithms to resolve. Moreover, approaches have to cope with high variations in scale and severe changes in pose between frames, and similar to multi-person pose estimation, the quality and consistency are highly impacted by factors such as lightning conditions, camera motion, and motion blur.

To tackle these challenges, especially those induced by motion blur, occlusions, or missed detections, many works incorporate temporal information from neighboring frames into their methods for more robust pose estimation results (Pfister et al., 2015; Xiao et al., 2018; Xiu et al., 2018; Snower et al., 2019; Song et al., 2017; Jain et al., 2015; Zhou et al., 2020a; Yang et al., 2021b). In literature, these approaches are often denoted as multi-frame pose estimation (Liu et al., 2021b) or pose estimation in videos (Andriluka et al., 2018). For that reason, this section reviews related works under two aspects, multi-frame pose estimation, and multi-person pose tracking, as some works towards multi-person pose tracking do not incorporate multi-frame pose estimation into their pipelines.

2.1.3.1 Multi-Frame Pose Estimation

A widely used technique in multi-frame pose estimation as explored by several works (Pfister et al., 2015; Xiao et al., 2018; Xiu et al., 2018; Snower et al., 2019; Song et al., 2017; Jain et al., 2015), involves integrating motion information to transform estimated poses from previous frames to the current frame, i.e., employing optical flow. Subsequently, an enlarged bounding box is computed for each pose, and a person detector reevaluates the pose for each transformed bounding box. In a subsequent stage, redundant poses are eliminated through a pose non-maximum suppression process. Pfister et al. (2015) and Song et al. (2017), on the other hand, employ optical flow to warp and align heatmaps of neighboring frames. Pfister et al. (2015) utilize a parametric cross-channel pooling layer to estimate the final heatmaps. In contrast, Song et al. (2017) propose a message-passing layer that operates on a spatiotemporal graph constructed by keypoint candidates and their relationships after warping. Both methods are designed under the assumption of a constrained scenario where only a single person is present in the video. In contrast, Jain et al. (2015) suggest directly merging motion features (Weinzaepfel et al., 2013) into a CNN to estimate poses in videos efficiently.

Temporal feature-alignment Contrary to explicit motion-based warping techniques, other works emphasize feature alignment methods (Bertasius et al., 2019; Liu et al., 2021b, 2022; Feng et al., 2023) based on implicit motion. For example, the work of Bertasius et al. (2019) focuses on estimating the poses of a frame A by utilizing the features extracted from a temporally distant frame B. As a result, their proposed approach effectively learns motion offsets implicitly to facilitate feature warping efficiently. This technique enables the authors to propose an effective method for temporal pose aggregation. Liu et al. (2021b) extend the work of Bertasius et al. (2019) and proposes a temporal refinement network that exploits temporal information by aggregating pose heatmaps of consecutive frames. Their method incorporates an additional pose residual fusion module that estimates inter-frame keypoint offsets. A pose residual fusion module refines the initial pose estimate based on predicted keypoint residuals and aggregated pose heatmaps. Feng et al. (2023) follow a similar idea but propose a multi-stage temporal difference encoding module that leverages deformable convolutions (Dai et al., 2017; Zhu et al., 2019) to encode motion features. Their novel representation disentanglement module aims to remove the impact of background noise and employs mutual information minimization between relevant motion features and noisy motion cues. Liu et al. (2022) propose a different strategy and follow a coarse-to-fine feature alignment strategy incorporating a global transformation module and a local calibration module. The global transformation module learns affine transformations to align temporally distant frames. In contrast, the local calibration module processes the aligned features. It performs pixel-wise deformation by incorporating deformable convolutions (Dai et al., 2017) to obtain a fine alignment between the keyframe and its neighboring. The resulting finely aligned features are aggregated with the key frame feature to enhance the feature representation. Additionally, the authors propose a mutual information objective to encode complementary task-relevant information.

RNN-based method Other methods leverage recurrent neural networks (RNN) (Gkioxari et al., 2016; Yue et al., 2018; Artacho and Savakis, 2020; Fan et al., 2021) to aggregate temporal information and implicitly encode motion information for pose estimation in the current frame. As an

example, [Gkioxari et al. \(2016\)](#) and [Yue et al. \(2018\)](#) employ an Encoder-RNN-Decoder framework, in which a frame is initially processed by an encoder network to acquire high-level image representations. Subsequently, RNNs propagate temporal information among adjacent frames, generating hidden states. Finally, a decoder network inputs these hidden states and produces pose estimation results for the current frame. However, these approaches typically necessitate large networks to effectively learn intermediate representations or preliminary poses to achieve satisfactory performance, resulting in limited efficiency. While [Gkioxari et al. \(2016\)](#) can capture longer temporal dependencies, it relies on assumptions regarding regular motion patterns. [Artacho and Savakis \(2020\)](#) employ an LSTM module to refine predicted keypoint heatmaps of the current frame by jointly processing the predicted keypoint heatmaps of the previous frame. [Fan et al. \(2021\)](#) argue that video compression techniques heavily depend on temporal coherence to minimize video size. To address this aspect, the authors propose a motion adaptive pose estimation network that effectively utilizes the temporal coherence in compressed video streams, thus avoiding the computationally expensive feature extraction task. An adaptive gate is employed to determine whether performing a complete extraction of image features is necessary to compensate for spatially aligned features transmitted through a motion-compensated convolutional LSTM.

Efficient methods Incorporating motion features explicitly or implicitly or utilizing recurrent neural networks introduces an additional computational overhead, restricting inference time. Consequently, some works ([Zhang et al., 2020d](#); [Nie et al., 2019a](#)) have explored various directions to develop robust and efficient methods for estimating the pose across multiple frames. For example, [Zhang et al. \(2020d\)](#) introduce a key-frame proposal network that has been trained unsupervised. This network enables the recovery of a person’s complete pose throughout a video by interpolating based on the estimated key frame poses. On the other hand, [Nie et al. \(2019a\)](#) propose a dynamic kernel distillation model that leverages temporal cues from previous frames. Unlike related works that perform feature warping using temporal cues ([Bertasius et al., 2019](#); [Xiao et al., 2018](#); [Xiu et al., 2018](#)), or aggregate temporal information using recurrent schemes ([Yue et al., 2018](#); [Artacho and Savakis, 2020](#); [Fan et al., 2021](#)), the authors transfer pose localization knowledge from previous frames. In Chapter 4, we demonstrate that pose-based kernels effectively localize keypoints in temporally distant frames.

Action-based pose estimation As it has been shown in literature ([Johansson, 1973](#)), the motion of specific keypoints on the human body keypoints is sufficient for accurate identification of human activities such as walking and running. Conversely, human activity cues can serve as reliable priors for estimating human poses, as each activity generates unique motion patterns. Several works ([Yao and Fei-Fei, 2012](#); [Ukita, 2013](#); [Nie et al., 2015](#); [Iqbal et al., 2016](#)) aim to leverage these motion patterns and incorporate human activity cues into their pose estimation modules to enhance the accuracy of estimated poses in video sequences. [Ukita \(2013\)](#) and [Yao and Fei-Fei \(2012\)](#) employ a mixture of pictorial structure models with a separate model for each action class. However, this approach does not scale well with the number of action classes. [Nie et al. \(2015\)](#) propose jointly estimating action classes and refining poses. In particular, estimated poses are divided into sub-parts and then tracked across frames based on action-specific constraints. The proposed approach estimates keypoint locations and action labels based on the part tracks. While the action labels are inferred with additional

activity recognition approaches, the proposed work of [Iqbal et al. \(2016\)](#) eliminates the need for additional action recognition approaches. Moreover, activities are directly predicted from a sequence of poses. The authors further employ an action-conditioned pose estimation model to re-estimate the pose for each frame.

2.1.3.2 Multi-Person Pose Tracking

The body of work towards multi-person pose tracking is diverse and approached from multiple directions. For instance, multi-person pose tracking can be approached from a bottom-up or top-down direction similar to multi-person pose estimation in Section 2.1.2. Tracking can also be conducted online or offline. Online approaches follow the tracking-by-detection paradigm and associate newly estimated poses with existing tracks. In contrast, offline approaches first estimate the poses for the entire video or a fixed subset and perform the association of poses afterward.

Due to its complexity and missing datasets, multi-person pose tracking has not been tackled for a long time. However, early works ([Andriluka et al., 2008](#); [Izadinia et al., 2012](#)) follow an off-line approach and first estimate body part locations separately for each person and, in a second step, employ body-part tracklets that are successively merged into person tracks. [Insafutdinov et al. \(2017\)](#) and [Iqbal et al. \(2017\)](#) were the first to release datasets for multi-person pose estimation and tracking in the wild. Both datasets are on a very small scale and not suitable for real-world applications. Both works ([Insafutdinov et al., 2017](#); [Iqbal et al., 2017](#)) formulate a graph partitioning problem to solve a spatio-temporal graph of detected keypoints with integer linear programming (ILP). The unary potentials of the graph are obtained by a joint detector ([Insafutdinov et al., 2016](#)), and temporal potentials are obtained by DeepMatching ([Weinzaepfel et al., 2013](#)). With joint effort, the authors of both datasets propose PoseTrack ([Andriluka et al., 2018](#)), the first large-scale dataset for multi-person pose tracking, and propose extensions to ([Insafutdinov et al., 2017](#); [Iqbal et al., 2017](#)) as baseline methods. Essentially, the authors convert the work of [Insafutdinov et al. \(2017\)](#) into a top-down approach and employ Faster R-CNN ([Ren et al., 2015](#)) as an object detector to estimate the poses of each detected person individually. Additionally, a stronger pose estimator ([Cao et al., 2017](#)) is incorporated into the approach of [Iqbal et al. \(2017\)](#). [Jin et al. \(2017\)](#) closely follows the improved baseline of [Iqbal et al. \(2017\)](#) and propose further modifications to the pose estimator ([Cao et al., 2017](#)) by employing inception-residual blocks. PoseTrack provides three challenges: single-person pose estimation, multi-frame pose estimation, and multi-person pose tracking. As we will show in Chapter 5, we extend PoseTrack and propose PoseTrack21 that contains additional and more challenging annotations and further allows the evaluation of person search, multi-object tracking, and a new challenge we refer to as multi-person re-identification tracking.

The optimization of spatio-temporal graphs is a computationally challenging task and not suitable for real-time applications. To address this issue, researchers ([Girdhar et al., 2018](#); [Doering et al., 2018](#); [Xiao et al., 2018](#); [Jin et al., 2019b](#); [Snower et al., 2019](#); [Raaj et al., 2019](#)) have shifted their focus from complex optimization techniques to efficient heuristics and approximation algorithms that can quickly find near-optimal solutions. Several works ([Girdhar et al., 2018](#); [Doering et al., 2018](#); [Xiao et al., 2018](#)) utilize bipartite graphs to represent the pairwise relationships between frames and employ greedy or Hungarian matching ([Kuhn, 1955](#)) algorithms to associate tracks with poses. Traditionally, detected poses in the first frame initiate tracks that are matched with detected poses

Method	Detection Improvement	Tracking
Gated Attention (Chapter 6)	-	Gated Attention
Dynamic-GNN (Yang et al., 2021b)	GNN Prediction	GNN Prediction
TKMRN(Zhou et al., 2020a)	Temporal Context	Temporal Keypoint Matching
CombDetTrack (Wang et al., 2020c)	3D Video HRNet	Spatio-temporal Tracklet Merging
CorrTrack (Chapter 4)	Correspondences	Keypoint Correspondences
KeypointTrans(Snowe et al., 2019)	Temporal OKS	Pose Entailment
HRNet (Sun et al., 2019a)	Temporal OKS	Optical Flow
POINet (Ruan et al., 2019)	-	Ovonic Insight Net
MDPN (Guo et al., 2018)	Ensemble	Optical FLOW
LightTrack (Ning and Huang, 2019)	Ensemble / BBox Prop.	GCN
SimpleBaseline (Xiao et al., 2018)	Temporal OKS	Optical Flow
ProTracker (Girdhar et al., 2018)	-	IoU
SSVC (Fu et al., 2021)	-	Video Instance Correspondences
STAF (Raaj et al., 2019)	-	ST Fields
ST Embeddings (Jin et al., 2019b)	-	ST Embeddings
MIPAL (Hwang et al., 2019)	-	Flow Maps
JointFlow (Doering et al., 2018)	-	Flow Fields
ArtTrack (Insafutdinov et al., 2017)	-	Integer Linear Programming
PoseTrack17 (Iqbal et al., 2017)	-	Integer Linear Programming

Table 2.1: Overview of related works on multi-person pose tracking. The table is divided into top-down methods (top block) and bottom-up methods (bottom block).

in subsequent frames. A new track is initiated whenever a pose cannot be associated with a track. Conversely, a track is deactivated if it cannot be matched with any detected pose for a predefined number of frames. One example of this approach (bottom-up) is presented in (Girdhar et al., 2018), where the authors extend MaskRCNN (He et al., 2017) and propose to employ 3D convolutions to process a sequence of frames. Given a sequence of frames, the network predicts person tubelets for every person present in the sequence, where a person tubelet is a sequence of detected bounding boxes and keypoints for each person instance. Poses are then tracked based on bounding box intersection over union (IoU) between tracks and detections under the assumption of minimal motion between frames. While achieving a higher tracking performance, this method is two magnitudes faster than the work of Iqbal et al. (2017).

We follow Section 2.1.2 and categorize the remaining works towards multi-person pose tracking into bottom-up and top-down approaches. Both categories have the same advantages and disadvantages as discussed for multi-person pose estimation approaches: Bottom-up approaches first detect individual body parts, cluster them into poses, and predict spatio-temporal pose relationships across frames. This allows for a holistic view of the entire scene, which can be advantageous in complex interactions between persons. Nonetheless, bottom-up approaches suffer from variations in scale and may struggle to distinguish between body parts in close vicinity. On the other hand, top-down approaches estimate the pose for each detected person individually and tend to be more accurate than bottom-up approaches. However, top-down approaches are limited to the context of the respective person crops and may struggle in complex interactions. Furthermore, top-down tracking methods have a higher runtime, which increases with the number of people and lags behind bottom-up ap-

proaches. An overview of the most relevant works is summarized in Table 2.1.

Bottom-Up Methods Most bottom-up methods predict the spatiotemporal relationships among keypoints and poses (Doering et al., 2018; Raaj et al., 2019; Hwang et al., 2019; Jin et al., 2019b; Fu et al., 2021). These methods typically follow a two-stage pipeline approach, where the first step involves clustering keypoints into poses, and the second step involves associating these poses temporally across different frames. One such approach is proposed by Doering et al. (2018), Hwang et al. (2019), and Raaj et al. (2019) that extend the work of Cao et al. (2017) to model relations between keypoints temporally. Originally, Cao et al. (2017) employed a VGG backbone (Simonyan and Zisserman, 2015) and predicted keypoint heatmaps and part affinity fields (PAF) to cluster pairs of keypoints. While Doering et al. (2018) propose temporal flow fields to link keypoints across frames, this method suffers from ambiguities during slow movements or stationary poses. On the other hand, Hwang et al. (2019) propose temporal flow maps that predict the movement of entire limbs instead of individual keypoints. Raaj et al. (2019) follow the idea of temporal flow fields by Doering et al. (2018) but propose an improved topology for spatio-temporal affinity fields that are predicted by a recurrent convolutional neural network (CNN). The recurrent network utilizes the output from both the previous and the current frames to generate refined predictions, overcoming the problem of stationary poses. Jin et al. (2019b) follow a different body of work and highlights several limitations of the keypoint embeddings presented in Newell et al. (2017), such as their lack of interpretability and trainability. To overcome these limitations, the authors build upon the work of Newell et al. (2017) and introduce several auxiliary tasks to incorporate geometric ordinal relations into the keypoint embeddings. Additionally, the authors propose using spatial instance embeddings (SIE) to encode relative displacements of keypoints from the human center, facilitating reliable keypoint grouping. To handle the temporal association of poses during tracking, the authors introduce an additional temporal network that leverages pooled bounding box features and temporal instance embeddings (TIEs) to predict instance-aware human embeddings. TIEs are an extension of SIEs that predict the relative temporal displacement between keypoints and human centers. In Fu et al. (2021), the authors propose a semi-supervised method for tracking instance segmentation masks in videos. They utilize instance-contrastive learning with maximum entropy regularization and incorporate self-supervised video correspondence learning, allowing their method to be trained on unlabeled videos, improving tracking performance. The proposed method can be extended to multi-person pose tracking and achieves comparable performance.

Top-Down Methods In contrast to bottom-up approaches, top-down approaches do not consider the overall context of the scene, but instead rely on motion information (Guo et al., 2018; Xiao et al., 2018; Snower et al., 2019; Ruan et al., 2019; Wang et al., 2020c; Yang et al., 2021b; Ning and Huang, 2019). These methods incorporate spatio-temporal context based on features and poses into their frameworks. Most of the state-of-the-art works employ pose estimation models based on either ResNet (Xiao et al., 2018; Guo et al., 2018; Zhang et al., 2019; Ning and Huang, 2019; Ruan et al., 2019; Zhou et al., 2020a) or HRNet (Snower et al., 2019; Wang et al., 2020c; Yang et al., 2021b). However, the effectiveness of top-down methods is heavily reliant on the performance of the underlying object-detector and pose estimation models, which are known to be susceptible to errors caused by motion blur and occlusion. Consequently, many approaches employ strategies involving multi-

frame context and multi-frame pose estimation in order to address missing detections and ensure reliable tracking. For example, in the work by [Xiu et al. \(2018\)](#), a motion-guided box propagation strategy based on DeepMatching ([Weinzaepfel et al., 2013](#)) is employed to recover missed bounding boxes, while [Xiao et al. \(2018\)](#) employ optical flow to propagate poses from previous frames into the current frame. On the other hand, [Snower et al. \(2019\)](#) make the assumption of negligible inter-frame movement and copies poses from the previous frame into the current frame. Both methods ([Xiao et al., 2018](#); [Snower et al., 2019](#)) leverage magnified bounding boxes from the warped poses for pose estimation in the present frame. They further employ Object Keypoint Similarity (OKS) to eliminate redundant poses. Contrary, [Zhou et al. \(2020a\)](#) exploit temporal context around the current frame in forward and backward directions, iteratively selects the best keypoint detections from previous and subsequent frames, and employs a keypoint refinement module to improve pose estimation accuracy. It aggregates features from detections along the temporal path and updates the predicted pose by considering potential correct keypoint locations.

After the spatiotemporal refinement of previously estimated poses, most approaches follow an online approach to track poses throughout a video. For example, [Xiu et al. \(2018\)](#) introduce a tracking algorithm that employs dynamic programming and exploits spatio-temporal context to optimal associations between tracks and detections. Compared to bipartite methods, this approach exhibits resilience to occlusion and motion blur, compared to bipartite methods ([Xiao et al., 2018](#); [Ning and Huang, 2019](#); [Yang et al., 2021b](#)). The proposed Pose Flow NMS eliminates redundant poses by utilizing temporal context, encompassing pose distances within and across frames for overlapping tracks. In subsequent work, [Ning et al. \(2019\)](#) builds upon [Xiu et al. \(2018\)](#) by incorporating a more powerful pose estimation model ([Chen et al., 2018b](#)), known as Cascaded Pyramid Network. Multiple works ([Xiao et al., 2018](#); [Snower et al., 2019](#); [Ning and Huang, 2019](#); [Zhou et al., 2020a](#)) employ the temporal context of estimated poses to reason about the similarity of temporal distant poses. While [Xiao et al. \(2018\)](#), [Snower et al. \(2019\)](#) and [Ning and Huang \(2019\)](#) focus on keypoint locations, [Zhou et al. \(2020a\)](#) and [Yang et al. \(2021b\)](#) additionally incorporate image features into their methods. [Xiao et al. \(2018\)](#) and [Guo et al. \(2018\)](#), for example, propose a flow-based tracking algorithm that warps the poses of tracks from the previous frame into the present frame based on optical flow and employs greedy matching based on OKS to assign tracks and detections, whereas [Guo et al. \(2018\)](#) follow a multi-domain learning strategy. Their network comprises dataset-specific prediction heads and leverages the estimates from all heads to form a final estimation. [Snower et al. \(2019\)](#) propose a pose tracking approach based on a pose entailment method. This method classifies whether two poses from different time steps belong to the same person instance. The concept of pose entailment draws inspiration from natural language processing, where textual entailment is used to classify whether a hypothesis sentence can be inferred from a premise sentence. Their method employs a transformer network to process tokenized pose pairs to learn the spatio-temporal relationships between keypoints. Each pose token consists of position tokens, keypoint type tokens, and tokens representing the temporal offset between poses. In a similar vein, [Ning and Huang \(2019\)](#) propose a pose-similarity module to associate tracks and detection, but in contrast to [Snower et al. \(2019\)](#), they employ a siamese graph convolutional network that is trained in a contrastive fashion. Additionally, their proposed tracking framework integrates a pose-recovery mechanism, which allows for the re-estimation of targets that may have been missed by the pose estimator. On the other hand, [Zhou et al. \(2020a\)](#) suggest determining the similarity on a keypoint level and propose dedicated classifiers

for each keypoint class to infer the similarity between temporally distant keypoints. Unlike [Snower et al. \(2019\)](#) and [Ning and Huang \(2019\)](#), the authors extract feature descriptors for each keypoint, and the overall pose similarity, used for the association of tracks and detections, is defined as the sum of keypoint similarities. [Yang et al. \(2021b\)](#) also incorporate a graph neural network, but in contrast to [Ning and Huang \(2019\)](#), the GNN utilizes the spatiotemporal context of each tracklet and incorporates keypoint image features and keypoint locations to predict a corresponding pose in the current frame. The predicted poses are then matched with estimated poses from the pose estimator ([Sun et al., 2019a](#)) based on OKS. This way, missed poses by the pose estimation network can be recovered. [Ruan et al. \(2019\)](#) explore a pose-guided network designed for efficient track association. The proposed network is trained end-to-end to extract appearance features conditioned on pose, learn a similarity metric, and perform track association. These tasks are seamlessly integrated within a unified framework.

As indicated by [Zhou et al. \(2020a\)](#) and [Yang et al. \(2021b\)](#) and [Ruan et al. \(2019\)](#), appearance features play a crucial role in densely populated environments, allowing for the association of tracks with occluded pose detections and ambiguous pose configurations. Consequently, certain works ([Zhang et al., 2019](#); [Xu and Zhou, 2018](#)) aim to integrate person re-identification models into their tracking framework directly. For example, [Zhang et al. \(2019\)](#) follow the concepts presented in ([Haoshu Fang and Lu, 2016](#); [Papandreou et al., 2017](#); [Xiao et al., 2018](#)) and propose an offset-guided pose estimation network (cf. Section 2.1.2.1), while adopting the pose tracking approach from [Girdhar et al. \(2018\)](#). Furthermore, this work combines pose similarity (OKS) with appearance similarity by extracting features from an unspecified re-identification model within their matching algorithm. [Xu and Zhou \(2018\)](#) introduce a hierarchical tracking method that leverages basic similarity metrics such as Pose Intersection over Union (IoU) for positional similarity and an LSTM pose model for pose similarity based on angular configuration. Additionally, they employ a region-based deep network trained on person re-identification datasets ([Zheng et al., 2015](#); [Hirzer et al., 2011](#); [Li and Wang, 2013](#)) to extract appearance-based similarities, enabling reliable association in challenging situations involving occlusion or motion blur. The proposed method integrates spatial, temporal, and appearance cues to establish associations between tracks and detected poses.

Unlike previous works focusing mainly on online tracking methods, [Wang et al. \(2020c\)](#) adopt an offline approach and introduce a clip tracking network that simultaneously estimates and tracks poses in small video clips. Specifically, the clip tracking network extends a HRNet ([Sun et al., 2019a](#)) and replaces all 2D convolution layers with 3D convolutions. The network is trained to estimate the pose instances of the person at the center of the frame, positioned in the middle of the video clip. Consequently, their model implicitly learns to re-identify pose instances. Using a fixed step size, the authors propose to estimate the poses of overlapping video clips, forming arbitrary length tracks. The poses of tracklets with overlapping time frames are associated based on OKS, resulting in multiple pose hypotheses. To merge these pose hypotheses, the proposed pipeline employs Dijkstra’s algorithm ([Dijkstra, 1959](#)) to estimate the optimal keypoint locations.

2.2 Object Detection and Multi-Object Tracking

Multi-Object Tracking (MOT), also known as Multi-Target Tracking (MTT) ([Ciaparrone et al., 2020](#)), is a significant computer vision task that has received extensive attention in recent decades. MOT

aims to detect and track objects belonging to various categories in diverse scenarios. The objective is to estimate a bounding box for each object within a given frame, along with an associated target identity, and track all instances of objects throughout a video or a collection of videos. MOT finds applications in various domains, such as autonomous driving, action recognition, behavior analysis, traffic analysis, surveillance, and video querying in single and multi-camera scenarios. Notably, multi-object and multi-person pose tracking are closely related tasks, sharing similar foundational components such as object detection (for top-down multi-person pose estimation methods) and association blocks. As we show in Chapter 5, these tasks complement each other, and their evaluation can be consolidated using the same dataset. Furthermore, as illustrated in Chapter 6, pose information is vital in robustly associating individuals, particularly in crowded scenarios with substantial occlusions.

The existing body of literature on multi-object tracking predominantly follows the tracking-by-detection paradigm, which entails the separation of object detection and data association into distinct stages. An off-the-shelf object detector is commonly employed in the initial stage, while the subsequent stage heavily relies on motion and appearance information to associate tracks and detections. Recent approaches have adopted a detection-and-tracking paradigm, which involves the simultaneous detection and tracking of objects using an end-to-end trainable model. While both tracking paradigms are prevalent in contemporary literature, certain studies have pursued a detection-free tracking approach that necessitates manual initialization of a fixed number of objects. However, due to the deviation of detection-free tracking from the central theme of this thesis and its limited practical utility for modern real-world applications, we exclude works from this category from the literature we review.

Since object detection is a fundamental building block of MOT approaches, this section begins with a literature review of the most prominent state-of-the-art methods in Section 2.2.1. Finally, we conclude this section by conducting a comprehensive literature review of MOT in Section 2.2.2.

2.2.1 Object Detection in MOT

With the emergence of deep learning-based methods, the multi-object tracking community has witnessed significant advancements in tracking performance, primarily driven by improvements in object detection methods. The performance of the object detector critically influences the overall tracking performance (Ciaparrone et al., 2020). The literature on object detection can be broadly classified into two categories: one-stage and two-stage detectors. Two-stage detectors, which adhere to a coarse-to-fine detection paradigm, are epitomized by region-proposal-based methods (RCNN) (Girshick et al., 2014; Girshick, 2015; Ren et al., 2015; He et al., 2017; Cai and Vasconcelos, 2017; Lin et al., 2017a). In these methods, a Region Proposal Network (RPN) in the first stage generates coarse object locations, subsequently refined in terms of localization during the second stage. While capable of achieving high precision, their runtime often limits these methods.

Conversely, one-stage detectors (Redmon et al., 2016; Ge et al., 2021; Liu et al., 2016; Wang et al., 2023b; Lin et al., 2017b; Duan et al., 2019; Law and Deng, 2018; Tian et al., 2019; Zand et al., 2022) achieve higher frame rates and are frequently employed on mobile devices. However, their performance tends to degrade when detecting dense and small objects. One-stage detectors partition an image into multiple regions and simultaneously predict all object bounding boxes and probabilities

for each region. An alternative family of one-stage detectors (Tian et al., 2019; Zand et al., 2022; Duan et al., 2019; Law and Deng, 2018) reframes the bounding box regression problem as a keypoint detection task. For instance, CornerNet (Law and Deng, 2018) proposes detecting corner bounding box keypoints, which are later decoupled and regrouped using additional embedding information. More recent approaches (Tian et al., 2019; Zand et al., 2022; Duan et al., 2019) predict the center location of each object and regress all its attributes (such as size, orientation, and location) based on the reference center point.

Recently, transformer networks have profoundly influenced the field of computer vision. A pioneering work in this area is the DETection TRansformer (DETR) (Carion et al., 2020; Zhu et al., 2021), which combines a CNN backbone for feature extraction with a transformer network. This network models the relationships between objects and the global image context to produce the final set of object predictions in parallel directly.

2.2.2 Multi-Object Tracking

Many multi-object tracking (MOT) methods follow the **tracking-by-detection** paradigm and utilize state-of-the-art object detectors. Consequently, MOT reduces the assignment problem to one aimed at enhancing association performance beyond detection performance. It is common practice (Bergmann et al., 2019; Zhang et al., 2022) to incorporate appearance cues and motion to associate detections and track information. Appearance cues encompass re-identification features extracted from a pre-trained model or are jointly learned alongside the object detection model. The extraction of motion information often involves well-known techniques such as the Kalman filter (Kalman and Others, 1960) and camera motion calibration (Evangelidis and Psarakis, 2008). For the association of detections and tracks, most methods exploit location-based similarities (e.g., similarities between bounding boxes) and appearance similarities (e.g., cosine similarity), ultimately resulting in the computation of an association matrix. Prominent approaches to matching detections and tracks include the Hungarian algorithm (Kuhn, 1955) and greedy matching applied to the association matrix. Yu et al. (2016) highlights the limitations of the Hungarian algorithm that strives to achieve optimal performance but fails in cases of missing detections. Consequently, various association strategies have been proposed in the literature (Yu et al., 2016; Zhang et al., 2022; Wojke et al., 2017) to address this issue. These approaches involve dividing the association process into multiple stages. For instance, some studies (Yu et al., 2016; Zhang et al., 2022; Bae and Yoon, 2018) partition tracks into high and low-quality sets based on their detection scores, which are subsequently matched with detections consecutively. Other methods, such as Wojke et al. (2017), employ an iterative matching scheme where tracks are matched based on age, starting with the youngest tracks and progressing to the oldest. Here, the age of a track indicates the number of frames for which it has not been associated with any detection. Apart from different association strategies, works such as (Bergmann et al., 2019) employ motion models in combination with non-maxima suppression to recover missed detections.

Location and Motion Cues based Methods The Simple Online Realtime Tracking (SORT) method (Bewley et al., 2016) stands as one of the pioneering approaches that employ a Convolutional Neural Network (CNN), specifically Faster R-CNN (Ren et al., 2015), as the object detector. This method suggests a straightforward integration of motion cues. The authors adopt a Kalman filter

to predict the locations of each track in a new frame and leverage the Intersection over Union (IoU) metric between detected boxes and the predicted track boxes. Multiple methods (Han et al., 2022; Zhang et al., 2022; Stadler and Beyerer, 2022; Cao et al., 2023) follow SORT and propose various strategies to improve on the weaknesses of SORT. For instance, Cao et al. (2023) have identified that the assumption of linear motion amplifies the sensitivity of the Kalman filter to state estimation noise. This amplification leads to a significant variance in the estimated object velocity and subsequently results in the accumulation of prediction errors, particularly when a track is lost for multiple frames. The authors propose an additional update stage known as Observation-Centric Re-Update (ORU) to address this issue. The re-update stage is activated when a lost track is re-assigned. To prevent the accumulation of errors, the re-update incorporates virtual observations from previous time steps. These virtual observations are derived from a trajectory generated using the last observed position before the track was lost and the most recent observation that reactivates the track, acting as anchor points. Additionally, the authors employ Observation-Centric Momentum (OCM) to enforce direction consistency in the estimated tracks. Stadler and Beyerer (2022) follow a similar strategy and propose a camera motion-aware interpolation technique and an improved motion model based on the Kalman filter. The presented method also addresses the issue of ambiguous assignments and assesses various strategies for resolving duplications, specifically by searching for detection-track matches exhibiting similar affinities. On the other hand, Han et al. (2022) combine non-linear object motion with rigid camera motion to enhance robustness against drifting predictions. They propose an additional network to learn long-range tracklet features for the robust assignment of fragmented tracks. Zhang et al. (2022) propose the inclusion of low-confident detections that were previously discarded and presents a two-stage association strategy. In the first stage, high-confident detections, determined by a selected threshold, are matched to existing tracks. In the second stage, low-confident detections are matched to the remaining tracks.

Bochinski et al. (2017) and Yang et al. (2023) adhere to the constant velocity assumption and do not integrate any motion model into their tracking methods. While Bochinski et al. (2017) directly calculate the similarity between detections and tracks from the previous frame, yielding a rapid inference speed, this approach falls short in challenging scenarios involving fast motion and crowded environments. In contrast, Yang et al. (2023) adopt a more sophisticated approach by employing a cascaded matching strategy. Instead of utilizing a motion model, the authors propose constantly enlarging the bounding boxes for each track. This enables the initial stage to match slowly moving objects. In contrast, the subsequent stage matches the remaining detections with faster movement to the remaining tracks by incorporating a higher enlargement factor. Similarly, Wang et al. (2021b) follow a comparable approach and introduces a two-stage model that captures local and global motion consistencies. Unlike Yang et al. (2023), both stages in this model employ graph convolutional networks (GCN) for detecting associations between detections and tracks. The first stage encodes local motion and associates detections into tracklets. In contrast, the second stage utilizes an attention-based GCN, incorporating a temporal gated convolution mechanism to embed global motion consistencies.

Appearance-based Cues The association based on bounding box similarity is prone to errors, especially in crowded scenes with a high degree of occlusions, consequently leading to many identity switches. As shown in Wojke et al. (2017), the integration of appearance information helps to reduce the number of identity switches during tracking significantly and allows to re-initialize lost tracks. In

particular, the authors extend SORT (Bewley et al., 2016) with an additional ResNet (He et al., 2016) and augment the association matrix with cosine similarity. Moreover, the authors propose an iterative matching procedure based on the track age. Consequently, multiple methods (Yu et al., 2016; Bae and Yoon, 2018; Ullah and Cheikh, 2018; Fu et al., 2018; Zhou et al., 2018; Mahmoudi et al., 2019; Wen et al., 2019; Chen et al., 2019b,b) incorporate appearance features into their methods, e.g. based on pre-trained re-identification networks (Zheng et al., 2017b; Luo et al., 2019b). While these methods propose different techniques to leverage appearance features, they follow similar association strategies based on greedy or Hungarian matching as proposed in (Bewley et al., 2016; Wojke et al., 2017). For instance, Yu et al. (2016) and Ullah and Cheikh (2018) incorporate appearance features from a modified GoogLeNet (Szegedy et al., 2015). Yu et al. (2016) further employ a two-stage association scheme and divide the tracks into high-quality and low-quality sets associated with detections in two separate stages, similar to Zhang et al. (2022).

The current body of work can be divided into two main categories. The first line of works (Bae and Yoon, 2014; Kim et al., 2018; Seidenschwarz et al., 2023) emphasizes improving appearance features. Methods in the second category (Mahmoudi et al., 2019; Long et al., 2018; Zhou et al., 2018; Fang et al., 2017; Sadeghian et al., 2017; Xu et al., 2019; Chen et al., 2019b; Girbau et al., 2022; Zhang et al., 2020c) focus on the development of more sophisticated association strategies (Sun et al., 2021), including group models (Mahmoudi et al., 2019; Zhou et al., 2018; Girbau et al., 2022; Zhang et al., 2020c) or recurrent neural networks (Fang et al., 2017; Sadeghian et al., 2017; Chen et al., 2019b). One approach that falls into the first category is the work of Bae and Yoon (2014). Bae and Yoon (2014) follow a traditional approach and propose to learn appearance features using an incremental linear discriminant analysis in an online fashion to handle appearance variations. More recent works (Tang et al., 2017; Kim et al., 2018) employ convolutional neural networks to improve the extraction of appearance features. For instance, Tang et al. (2017) leverage body pose features to extract more discriminative appearance features. Kim et al. (2018) propose a bilinear LSTM that incorporates motion and appearance information to model strong long-term appearance features. Seidenschwarz et al. (2023) identify multiple cues that allow state-of-the-art performance while employing a standard re-identification network. In particular, the authors incorporate on-the-fly domain adaptation techniques and employ different similarity measures between detections and tracks based on the track state (e.g. active vs. inactive).

To improve the association process between detections and tracks, several works (Bae and Yoon, 2018; Mahmoudi et al., 2019; Aharon et al., 2022; Du et al., 2023; Xu et al., 2019) incorporate spatiotemporal features in addition to appearance features into the association process. Bae and Yoon (2018) leverage shape and motion features and Mahmoudi et al. (2019) employ target dynamic and position features. Aharon et al. (2022) and Du et al. (2023) follow the SORT tracking paradigm and extend ByteTrack (Zhang et al., 2022) and DeepSort (Wojke et al., 2017), respectively. Aharon et al. (2022) focus on improving the bounding box localization by a camera motion compensation-based tracker and an improved Kalman filter state vector. Moreover, a simple method is proposed to enhance the tracking performance, incorporating box and appearance similarities. Du et al. (2023), on the other hand, incorporate stronger object detection and appearance embedding modules and propose novel and lightweight tracking modules referred to as AFLink and GSI that can be incorporated in various third-party trackers. The appearance-free link module (AFLink) aims to predict the connectivity between tracklets based on spatio-temporal similarities. The Gaussian Smoothed Inter-

polation (GSI) module predicts non-linear motions based on Gaussian process regression (Williams and Rasmussen, 1995) to predict missing detections. Long et al. (2018) and Girbau et al. (2022) follow a different direction and propose a hierarchical data association strategy based solely on appearance features. Moreover, Long et al. (2018) employ IoU if appearance features are noisy, e.g. due to occlusion. Sun et al. (2021) propose a Deep Affinity Network based on a Siamese VGG-like network (Simonyan and Zisserman, 2015) that directly computes the similarity between detections and tracks based on their appearance. Xu et al. (2019) additionally incorporate topology cues into their method and propose Spatial-Temporal Relation Networks (STRN) that incorporate appearance and motion features and predict attention weights across frames.

Multiple works (Fang et al., 2017; Sadeghian et al., 2017; Kim et al., 2018; Zhang et al., 2020c) employ recurrent neural networks to model long-term appearance and motion cues. For instance, Fang et al. (2017) propose a recurrent autoregressive network to predict future tracklet candidates, which are then associated with detected objects based on appearance similarity and IoU. Contrary, Sadeghian et al. (2017) propose multiple RNNs (Hochreiter and Schmidhuber, 1997) to directly model appearance, motion, and interaction cues to encode long-term dependencies for robust tracking results. The works of Chen et al. (2019b) and Zhang et al. (2020c) employ recurrent neural networks to learn spatiotemporal association metrics directly. For instance, Chen et al. (2019b) propose a recurrent metric network (RMNet) that is robust to long-term object variations and acts as a similarity metric between tracks and detections. Based on RMNet as a similarity metric, the authors further propose a batch multiple hypothesis strategy that optimizes a hypothesis tree to find the best tracking results. Zhang et al. (2020c), on the other hand, incorporate a motion evaluation network (MEN) and an appearance evaluation network (AEN) that are based on LSTMs (Hochreiter and Schmidhuber, 1997). Both are used to encode long-term tracklet features. Similar to (Chen et al., 2019b), the authors follow a batch multiple hypothesis strategy.

A different line of work (Xiang et al., 2015; Chu and Ling, 2019; Chu et al., 2019) proposes to employ single object tracking methods to estimate object locations accurately and to reduce false negatives. Nonetheless, compared to related works on multi-object tracking, these methods are very slow, and their runtime is linearly scaled with the number of persons in a scene.

Joint detection-and-tracking Two-stage tracking methods that utilize separate models for detection and tracking can leverage the most suitable models for each task, which helps with scale variations, e.g. bounding boxes can be cropped and resized before being passed to the re-identification network. As a result, these approaches have demonstrated superior performance on common benchmarks such as the MOT17 Challenge (Milan et al., 2016). Given the inherent characteristics of these two-stage approaches, achieving real-time performance and high inference frame rates becomes challenging. To address this limitation, various works (Voigtlaender et al., 2019; Wang et al., 2020d; Liang et al., 2022; Pang et al., 2021; Lu et al., 2020) emphasize joint detection and tracking approaches. These methods propose object detection and extracting re-identification features within a unified model.

One prevalent strategy towards joint detection and tracking networks involves extending object detectors to incorporate motion features (Feichtenhofer et al., 2017; Zhou et al., 2020b; Pang et al., 2020; Peng et al., 2020) and re-identification features (Voigtlaender et al., 2019; Wang et al., 2020d; Zhang et al., 2021) into a unified neural network. Feichtenhofer et al. (2017) and Peng et al. (2020)

propose a Siamese network architecture that regresses and associates bounding boxes in adjacent frames. Specifically, [Feichtenhofer et al. \(2017\)](#) predict bounding boxes for adjacent frames and calculate inter-frame displacements between the boxes based on correlated image features. In contrast, [Peng et al. \(2020\)](#) employ a chained-anchor mechanism that simultaneously regresses two bounding boxes for each object instance in both frames. This method also utilizes a joint attention module that incorporates information from the classification and re-identification branches to guide the bounding box regression head. [Bergmann et al. \(2019\)](#), on the other hand, exploit the bounding box regression head to align propagated track bounding boxes and propose a non-maxima suppression-based tracker. CenterTrack ([Zhou et al., 2020b](#)) builds on top of CenterNet ([Duan et al., 2019](#)) and detects objects as points. The method takes the previous and the current frame as input, together with the estimated track centers in the previous frame. It predicts new object locations and offset vectors to guide the association of tracks and detections.

Although these methods achieve strong tracking results, these methods only operate on adjacent frames and consequently fail to recover lost tracks ([Zhang et al., 2021](#)). Therefore, methods like [Bergmann et al. \(2019\)](#) employ additional re-identification networks. Moreover, ([Voigtlaender et al., 2019](#); [Wang et al., 2020d](#); [Zhang et al., 2021](#)) directly incorporate re-identification branches into the object detection networks. For instance, [Voigtlaender et al. \(2019\)](#), [Redmon and Farhadi \(2018\)](#) and [Zhang et al. \(2021\)](#) propose to augment Mask-RCNN ([He et al., 2019b](#)), YoloV3 ([Redmon and Farhadi, 2018](#)) and CenterNet ([Duan et al., 2019](#)) with an additional re-identification branch, respectively. This way, these methods regress a bounding box and a re-ID embedding for every object proposal. [Voigtlaender et al. \(2019\)](#) implement a cascaded pipeline in which object proposals are detected first. The extracted proposal features are then processed by the re-identification branch to obtain re-identification features. In contrast, [Wang et al. \(2020d\)](#) simultaneously estimate re-identification for each object proposal. For both methods, the quality of the re-identification embeddings strongly correlates with the quality of the object proposals. [Zhang et al. \(2021\)](#) observed the weak re-identification performance in both methods, leading to increased identity switches. As the quality of the re-identification features strongly relies on the quality of the estimated object proposals, the authors argue that proposal-based methods are strongly biased towards bounding box localization accuracy. Consequently, both networks are trained in favor of bounding box localization accuracy over the quality of re-identification features. For that reason, [Zhang et al. \(2021\)](#) employ a heatmap-based object detector ([Duan et al., 2019](#)) without localization bias and incorporate the tracking method of [Long et al. \(2018\)](#). [Tokmakov et al. \(2021\)](#) follow CenterNet ([Duan et al., 2019](#)) as well and propose a recurrent memory module based on a ConvGRU ([Ballas et al., 2015](#)) that employs the tracklet history to reason about object locations, even during long periods of object occlusion. In contrast, [Wang et al. \(2021c\)](#) extend FairMOT ([Zhang et al., 2021](#)) and incorporate a pixel-wise local correlation module into the network that models topological relationships between target locations and their local environment. The proposed tracker follows an optical flow-based tracking mechanism similar to [Zhang et al. \(2020b\)](#) and predicts the motion of target center locations based on correspondences.

[Pang et al. \(2020\)](#) utilize a spatio-temporal object detector with a 3D ResNet ([Fang et al., 2019](#)) as its backbone. This detector processes a sequence of frames to detect bounding tubes, representing the spatiotemporal location of the detected objects. A bounding tube is a collection of three bounding boxes that pertain to the same object and are distributed across multiple frames. The proposed

approach enables the generation of bounding tubes that span non-adjacent frames, facilitating the recovery of missing frames through object location interpolation. All tubes are connected into tracks based on the Intersection over Union (IoU) metric during the tracking process. [Yan et al. \(2022\)](#) propose a unified tracking framework that addresses the complex tasks of Multiple Object Tracking (MOT), including multi-object segmentation and visual object tracking, simultaneously. This framework achieves concurrent tracking by integrating pixel-wise correspondences between a reference embedding and the embedding of the current frame, as well as by incorporating feature fusion in the prediction head. To effectively capture long-range dependencies, the authors integrate deformable attention ([Zhu et al., 2021](#)) into their network.

[Wang et al. \(2021d\)](#) and [Dendorfer et al. \(2022\)](#) propose methods that can be integrated into existing approaches. For instance, [Wang et al. \(2021d\)](#) employ a Multiple Nodes Tracking (MNT) framework. This framework employs a Recurrent Tracking Unit (RTU) based on LSTMs ([Hochreiter and Schmidhuber, 1997](#)) to score tracks by accumulating long-term information. The authors also propose an augmentation method to generate synthetic track data to overcome the problem of limited data in MOT datasets. On the other hand, [Dendorfer et al. \(2022\)](#) explore the impact of trajectory forecasting on the performance of long-term multi-object tracking. In their work, the authors propose to forecast the path of each trajectory from a bird’s eye view (BEV) space, thus reducing the search space for possible associations. The authors employ image depth estimation and segmentation methods to compute a holography transformation to map detected bounding boxes into the BEV space.

[Wang et al. \(2021f\)](#) employ a graph-based method to achieve two main objectives: 1) incorporating spatio-temporal object-to-object or object-track relations, and 2) improving data association. Specifically, [Wang et al. \(2021f\)](#) extend CenterNet ([Duan et al., 2019](#)) by incorporating a graph neural network (GNN) to extract descriptive object features that encode relational information within a given spatial context. This is achieved by constructing a graph representation, where nodes correspond to the features of both detected objects and tracklets, and edges are generated between detection nodes within a predefined spatial distance of a tracklet node. The features of each node are then refined iteratively by incorporating information from neighboring nodes. Ultimately, the learned features are utilized for data association.

Due to the tremendous success of the transformer ([Vaswani et al., 2017](#)) architecture, especially due to its capabilities of modeling long-range dependencies, many works ([Zeng et al., 2022](#); [Meinhardt et al., 2022](#); [Zhang et al., 2023b](#); [Xu et al., 2021](#); [Cai et al., 2022](#); [Zhao et al., 2022](#)) have proposed transformer-based method for MOT. While these methods have been shown to be accurate, they demand a high amount of computation resources. DETR ([Carion et al., 2020](#)) and Deformable DETR ([Zhu et al., 2021](#)) represent pioneering efforts in utilizing a transformer-based architecture for object detection. Notably, both approaches incorporate object queries within the decoder to encode detected objects and to estimate their respective bounding boxes. In a similar vein, [Meinhardt et al. \(2022\)](#), [Sun et al. \(2020\)](#); [Zeng et al. \(2022\)](#) and [Cai et al. \(2022\)](#) propose an extension to the work of [Zhu et al. \(2021\)](#) for multi-object tracking. This extension introduces track queries to the transformer decoder. During the tracking process, newly detected objects are initialized as track queries and are subsequently passed to the transformer decoder in subsequent frames. This enables the prediction of bounding boxes for the same object instance while updating the track queries. In contrast to TrackFormer ([Meinhardt et al., 2022](#)) and MOTR ([Zeng et al., 2022](#)), TransMOT ([Sun et al., 2020](#))

employs two transformer decoders for decoding detection and track queries, respectively. Furthermore, TrackFormer (Meinhardt et al., 2022) utilizes track NMS and track query re-identification techniques to enhance the tracking performance. In contrast, MOTR (Zeng et al., 2022) employs a strategy for tracklet-aware label assignment and a temporal aggregation network that updates the features of the track queries. Zhang et al. (2023b) further extend MOTR and propose MOTRv2, where object queries are replaced with proposal queries obtained from a state-of-the-art object detector (Ge et al., 2021). Cai et al. (2022) incorporate an additional memory encoding module that stores short-term and long-term information for each tracked object to encode long-term information fully.

Xu et al. (2021) and Zhao et al. (2022) propose to estimate objects from center point locations similar to Zhou et al. (2020b), Wang et al. (2021f) and Duan et al. (2019). TransCenter (Xu et al., 2021) utilizes the DETR transformer encoder (Carion et al., 2020) as the backbone network. A query learning framework (QLN) is also introduced, which learns dense multi-scale queries. These queries are processed by a transformer decoder for generating dense detection and track embeddings. On the other hand, P3AFormer (Zhao et al., 2022) uses either a ResNet (He et al., 2016) or a Swin-Transformer (Liu et al., 2021a) as the backbone network. This approach uses pixel-wise flow-guided feature propagation (Gong et al., 2021) between consecutive frames. The transformer decoder exploits these features to generate pixel-wise object distributions.

Off-line Methods Off-line methods (Zhang et al., 2008; Wen et al., 2014; Berclaz et al., 2011; Zamir et al., 2012; Milan et al., 2014; Choi, 2015; Tang et al., 2017; Brasó and Leal-Taixé, 2020; Hornáková et al., 2020; Cetintas et al., 2023; Zhou et al., 2022; Brasó et al., 2022) build a spatio-temporal graph and perform global optimization on the entire video sequence, thus often achieving better results compared to on-line methods, particularly due to their robustness to occlusions. These methods leverage various optimization techniques to enhance their performance. For instance, Zhang et al. (2008) construct a graphical model with nodes representing detections across frames, and optimize the assignment using a min-cost flow algorithm tailored to the graph’s structure for efficiency. Similarly, Berclaz et al. (2011) formulate data association as a flow optimization problem, solving it with the K-shortest paths algorithm to accelerate computation and minimize parameter tuning. Continuous energy minimization is another approach, as seen in Milan et al. (2014), which designs an energy function dependent on the targets’ locations and motions across all frames and physical constraints.

Recent developments have introduced more sophisticated approaches. For instance, Cetintas et al. (2023) present a unified hierarchical tracker capable of generalizing across multiple timespans and scaling to long videos. This model learns to utilize the most suitable cues for each timescale in a data-driven manner, enhancing identity preservation. Moreover, Zhou et al. (2022) follows the concept of DETR (Carion et al., 2020) and proposes a global tracking transformer that processes short frame sequences to generate global trajectories. This method can integrate any object detector and can be trained end-to-end. Graph neural networks are also prominent in off-line methods. For example, Brasó and Leal-Taixé (2020) propose trainable graph neural networks to globally associate detections, making MOT fully differentiable and simplifying graph construction with a neural solver for optimization. An extension, Brasó et al. (2022), integrates object segmentation.

In general, off-line methods also encompass various optimization techniques, including disjoint paths (Tang et al., 2015; Hornáková et al., 2020), efficient solvers (Berclaz et al., 2011; Butt and

Collins, 2013), minimum cliques (Zamir et al., 2012), network flows (Zhang et al., 2008; Berclaz et al., 2011; Brasó and Leal-Taixé, 2020; Brasó et al., 2022), and neural networks (Leal-Taixé et al., 2016; Son et al., 2017; Li et al., 2020b; Brasó and Leal-Taixé, 2020; Dai et al., 2021; Brasó et al., 2022; Zhou et al., 2022; Cetintas et al., 2023).

2.3 Person Search and Person Re-Identification

While person-identification methods are not directly related to the task of multi-person pose tracking and multi-object tracking, re-identification provides a strong prior for the association of tracks and is a key component for the re-detection of lost tracks. Person search, on the other hand, aims to detect and assign all instances of a given person provided in form of a bounding box crop. In this regard, person search can be regarded as a special case of multi-object tracking in which all detections in the first frame are treated as query. Person re-identification is then used to associate and track the remaining detections. Both tasks are challenging due to various impact factors such as changing camera views, varying image resolution, illumination changes, occlusions or background clutter.

Person Re-Identification Methods for person re-identification aim to generate robust feature representations of a query person, enabling the re-identification of each instance of that individual. Given the objective of person re-identification, most existing works (Sun et al., 2018c; Zheng et al., 2017a; Luo et al., 2019b; Chen et al., 2022; Hermans et al., 2017; Miao et al., 2019) adopt network architectures designed for image classification, such as ResNet50 (He et al., 2016), with essential modifications. These modifications include changing the stride of the last ResNet layer to 1 (Sun et al., 2018c; Hermans et al., 2017; Luo et al., 2019b), employing adaptive pooling in the last pooling layer (Sun et al., 2018c), or adding bottleneck layers with batch normalization after the last pooling layer (Luo et al., 2019b). Additionally, person re-identification systems employ various feature representation learning strategies, encompassing global feature representations (Zheng et al., 2017a; Hermans et al., 2017; Luo et al., 2019b; Wieczorek et al., 2021; Chen et al., 2023), local part-based features (Zhao et al., 2017; Yao et al., 2019; Sun et al., 2018c; Li et al., 2022a; Sun et al., 2019b), and prior-based (Miao et al., 2019; Chen et al., 2021b; Gao et al., 2020; Somers et al., 2023; Su et al., 2016; Nguyen et al., 2017; Wang et al., 2020b; Gao et al., 2020; Sarfraz et al., 2018; Zheng et al., 2019; Wei et al., 2017; Qi et al., 2018), such as estimated poses. Temporal information, such as video data (Subramaniam et al., 2019; Chen et al., 2022; Dai et al., 2018), and combinations of these strategies are also utilized.

For real-world applicability, research must address various noise factors, particularly occlusions, common in most scenes. To better manage occlusions, part-based models (Sun et al., 2018c; Li et al., 2022a; Sun et al., 2019b) divide each image into several parts and extract distinct part-based features. These features can be trained individually (Sun et al., 2018c) or recombined to obtain visibility-aware features (Gao et al., 2020; Wang et al., 2020b; Sarfraz et al., 2018; Zheng et al., 2019; Wei et al., 2017; Miao et al., 2019; Chen et al., 2021b) or mask-guided (Qi et al., 2018) approaches aim to suppress background noise by learning occlusion- and instance-aware features that are more robust in crowded scenarios with occlusions. For instance, Miao et al. (2019) leverage human pose landmarks to mitigate occlusion noise. This method generates attention maps to identify occluded body parts, guiding the model to focus on non-occluded regions. Partitioning global features and using pose

landmarks ensures that only visible regions contribute to the retrieval process, enhancing robustness against occlusion. [Chen et al. \(2021b\)](#) address the limitations of relying on holistic pedestrian regions. This method introduces an attention-guided mask module to locate visible body parts, supported by an occlusion augmentation scheme that diversifies and labels occlusions more precisely. [Wang et al. \(2020b\)](#) combine CNNs with a graph-based approach. After extracting semantic local features, this method uses an adaptive directional graph convolutional layer to manage occlusions and outliers. By viewing local features as graph nodes and aligning them through a cross-graph embedded alignment layer, their method enhances feature discriminability and alignment robustness.

To further improve robustness against occlusions, [Wang et al. \(2022d\)](#) target occlusions from non-pedestrian objects (NPO) and non-target pedestrians (NTP). This method employs an Occlusion Erasing Module to remove NPO features and a Feature Diffusion Module to integrate NTP characteristics into the feature space, guided by occlusion scores, thus improving the re-identification performance under occlusion. [Yu et al. \(2021\)](#) reconstruct occluded part features using information from neighboring samples. By leveraging visible part-based features and an outlier-removable graph neural network, NGFR enhances the representation of occluded body parts, improving the robustness and accuracy of ReID systems. [Hou et al. \(2020b\)](#) introduce a novel occlusion-robust block that recovers the semantics of occluded regions in the feature space. The Spatial RFC module uses non-occluded regions to predict features of occluded regions, while the Temporal RFC module refines these predictions using long-term temporal contexts. This approach allows seamless integration into existing CNNs, enhancing their occlusion robustness.

Recent works ([He et al., 2021](#); [Wang et al., 2022a,b](#); [Li et al., 2023](#); [Ye et al., 2022](#)) leverage the power of transformer architectures to discover and utilize diverse and discriminative features, pushing the boundaries of re-identification accuracy and robustness in the presence of occlusions. [Li et al. \(2021d\)](#) is one of the first works that employs a transformer encoder-decoder architecture for occluded person ReID. Discovering diverse parts through a transformer framework enhances part diversity and discriminability without requiring explicit part annotations. [Wang et al. \(2022b\)](#) utilize transformer encoders to capture contextual relationships in fixed-size image blocks. The method then uses pose-guided feature aggregation and matching mechanisms to highlight visible body parts, leveraging pose heatmaps and decoder keys to enhance body part discriminability. [Ye et al. \(2022\)](#) avoid explicit human structure parsing by employing a sparse encoder to drop less important image tokens, primarily background noise and occluders. This method includes a multi-view feature matching module and a feature consolidation decoder to improve ReID accuracy under occlusion. [Li et al. \(2023\)](#) introduce multiple embedding spaces within a vision transformer, each representing diverse and compact subspaces. A self-diverse constraint ensures these subspaces remain distinct, while a dynamic weight controller balances their importance during training, leading to more robust and discriminative embeddings. [Li et al. \(2022a\)](#) propose to learn discriminative implicit parts based on a vision transformer (ViT) ([Dosovitskiy et al., 2021](#)) architecture, which tokenizes each image into equally sized patches. The authors extract discriminative features for robust re-identification, designed to capture distinctive elements (e.g. a backpack) that aid in identifying individuals.

Image-based Person Search. Person search has gained significant attention due to its real-world applications, integrating object detection and person re-identification. The existing body of work on image-based person search can roughly be categorized into two-stage ([Chen et al., 2018a](#); [Lan](#)

et al., 2018; Han et al., 2019; Wang et al., 2020a; Dong et al., 2020b; Wang et al., 2020a) and single-stage methods (Xiao et al., 2017b; Chen et al., 2020a; Li and Miao, 2021; Chen et al., 2020b; Yan et al., 2021). In two-stage approaches, the person detection and person re-identification networks are trained separately to achieve the best performance for both detection and identification. Thus, disjoint training can lead to suboptimal person search performance due to detection-identification inconsistencies, such as inaccurate bounding box proposals (Lin et al., 2021). At the same time, one-stage methods share the backbone features and train sub-networks for person re-identification and detection. Both categories of person-search approaches have to cope with similar challenges (Lin et al., 2021). Firstly, the model must learn discriminative features from scene images that are robust to occlusions, changes in pose, and background clutter. Secondly, bridging the gap between the query and the detected person is crucial, as query features can differ significantly, e.g. due to different clothing. Thirdly, person detection and identification have conflicting objectives: detection focuses on common features, while identification emphasizes unique characteristics. Thus, a detected partial human body might not align with a full-body query image. Particularly, background clutter poses a challenge as different persons with similar backgrounds can be close in latent space, which leads to false associations. To reduce the impact of background clutter, Zheng et al. (2020) incorporate segmentation masks to extract person features invariant to background clutter. Chen et al. (2020b), on the other hand, propose a norm-aware embedding that separates persons from background based on feature norm and discriminates person identities by angles. Later on, the strategy of norm-aware embeddings was utilized in various works (Li and Miao, 2021; Yu et al., 2022; Han et al., 2021a). Dong et al. (2020a), on the other hand, propose an instance-aware branch that relates features extracted from cropped person images with features extracted from the entire image.

One of the first single-stage approaches was proposed by Xiao et al. (2017b), who introduced a re-identification branch into Faster R-CNN (Ren et al., 2015) that is trained with an Online Instance Matching (OIM) loss. OIM leverages a lookup table for known and unknown identities. Thus, the proposed method is scalable to datasets with numerous identities. The OIM approach is employed by many succeeding works (Xiao et al., 2017b; Gao et al., 2019; Chen et al., 2020a; Lee et al., 2022; Han et al., 2019). For instance, Gao et al. (2019) further extend the OIM loss by a center loss, while Chen et al. (2020a); Lee et al. (2022) extend the OIM architecture. In particular, (Chen et al., 2020a) propose a hierarchical online instance matching loss, while Lee et al. (2022) propose a novel normalization layer that alleviates the class imbalance problem and incorporate a new LOIM loss that employs adaptive momentum based on bounding box IoU to update the features in the lookup table. While a shared backbone significantly decreases the runtime, shared visual features reduce detection performance, as shown in Wang et al. (2020a). To address the localization problem in single-stage methods, earlier works (Liu et al., 2017; Munjal et al., 2019; Han et al., 2019) utilize query information to refine person proposals. For instance, the query-guided network proposed by Munjal et al. (2019) leverages global context from both query and gallery images to generate person proposals relevant to the query, guided by re-identification scores. (Yan et al., 2021) introduce the first anchor-free method for person search, addressing misalignment issues across multiple levels, including scale, region, and task.

Part-based models (Yan et al., 2019; Zhong et al., 2020; Tian et al., 2020) aim to mitigate the adverse effects of background clutter, occlusion, and similar appearances. Zhong et al. (2020), following a methodology akin to Sun et al. (2018c), segment the body into six parts and utilize a

self-attention mechanism to assign weights to the contributions of each body part. In contrast, [Yan et al. \(2019\)](#) use a person graph to extract region-based contextual information. [Tian et al. \(2020\)](#) jointly optimize instance segmentation and keypoint detection tasks to derive robust person features. Additionally, [Lan et al. \(2018\)](#) and [Lu et al. \(2019\)](#) introduce various multi-level losses applied to multi-level pyramid features extracted from the backbone network (e.g., ResNet50 ([He et al., 2016](#))). [Han et al. \(2021a\)](#) present a trident architecture comprising three branches dedicated to person detection, re-identification, and part classification. They propose an adaptive gradient propagation scheme that modulates the gradient flow through the re-identification and part classification branches based on detection confidence.

Recent research has increasingly focused on cascaded network architectures ([Li and Miao, 2021](#); [Jaffe and Zakhori, 2023](#); [Yu et al., 2022](#); [Fiaz et al., 2023](#)), aiming to address the often conflicting objectives of person detection and re-identification. The primary goal of these multi-stage methods is to decouple the tasks of detection and re-identification. These methods typically follow a coarse-to-fine scheme, wherein the initial stages are dedicated solely to the detection of proposals without incorporating any re-identification loss. Subsequent stages progressively refine these proposals and integrate re-identification heads. For instance, SeqNet ([Li and Miao, 2021](#)) extends the Faster R-CNN ([Ren et al., 2015](#)) into a two-stage network. [Jaffe and Zakhori \(2023\)](#) further enhance the SeqNet architecture by introducing SeqNeXt and a Gallery Filter Network (GFN) module, which discards irrelevant gallery scenes. Similarly, [Yu et al. \(2022\)](#) build upon the Cascaded R-CNN ([Cai and Vasconcelos, 2017](#)) and introduce a Cascade Occluded Attention Transformer (COAT) featuring a three-stage design. Each stage employs a multi-scale convolutional transformer with increasingly tighter IoU thresholds to progressively refine the quality of person proposals, like [Li and Miao \(2021\)](#). Later stages in COAT incorporate and refine re-identification features. Furthermore, [Fiaz et al. \(2023\)](#) propose a cascaded architecture akin to [Cai and Vasconcelos \(2017\)](#), tackling person search in a coarse-to-fine manner. Unlike [Chen et al. \(2020b\)](#), which attempt to decouple features using norm-aware embeddings but still shares parameters leading to sub-optimal performance, [Fiaz et al. \(2023\)](#) propose fully decoupled norm-aware embeddings for both detection and re-identification tasks. They also employ convolutional transformer blocks, similar to [Yu et al. \(2022\)](#), and introduce a context aggregator block to enhance the aggregation of scale information.

Weakly-supervised Person Search. Obtaining identity annotations for person search tasks is both costly and time-consuming. As a result, weakly supervised methods ([Wang et al., 2023a](#); [Li et al., 2022b](#); [Shuai et al., 2022](#); [Han et al., 2021b](#)) have gained traction by leveraging the abundance of person detection datasets that do not require exhaustive labeling.

[Han et al. \(2021b\)](#) propose a Region Siamese Network that employs a query-guided scheme for weakly supervised person search. This network utilizes a self-instance consistency loss alongside an inter-instance similarity consistency loss to supervise the learning of context-invariant feature representations. By balancing the separation and clustering of features, this approach ensures robust and discriminative feature learning, thereby enhancing the model’s performance in varying contexts.

[Li et al. \(2022b\)](#) take inspiration from unsupervised domain adaptation to develop a domain-adaptive person search framework that does not rely on identity annotations. Their method involves transferring person search models trained on labeled data to an unlabeled target domain. By eschewing the need for bounding box and identity annotations, they enhance domain-invariant feature learn-

ing through dynamic clustering and a hybrid hard case mining strategy. This approach significantly improves the adaptability and robustness of person search models in new, unlabeled environments. [Shuai et al. \(2022\)](#) introduce an identity-free training strategy based on contrastive learning. The authors utilize density-based clustering to leverage the temporal continuity of person appearances in video sequences, generating pseudo identities. This method capitalizes on the inherent structure in video data to create reliable pseudo labels, facilitating effective model training without explicit identity annotations. [Wang et al. \(2023a\)](#) propose an end-to-end self-similarity driven scale-invariant learning framework specifically designed for weakly supervised person search. Their approach integrates a multi-scale exemplar branch, which provides guidance by addressing the scale variation problem through a scale-invariant loss. Additionally, they introduce a dynamic multi-label learning mechanism that adapts effectively to different datasets, enhancing the model's versatility across various scenarios.

Preliminaries

In this chapter, we formally introduce the fundamental concepts underpinning the task of 2D multi-person pose estimation and tracking. Given that 2D multi-person pose tracking involves processing video data, we begin by covering the necessary background in Convolutional Neural Networks (CNNs) and transformer-based networks. Following this, we delve into the most relevant neural network architectures designed explicitly for multi-person pose estimation and tracking. Additionally, we explore the tasks of person re-identification and person search. Person re-identification can be seen as an integral part of the overall tracking process, enabling the continuous and consistent identification of individuals across different video frames. Lastly, we introduce and discuss the evaluation metrics employed to measure the performance and accuracy of multi-person pose estimation and tracking systems, multi-object tracking methods, and person re-identification and person search methods. These metrics are crucial for assessing the effectiveness and reliability of the proposed models and methodologies.

Contents

3.1	Neural Networks	40
3.1.1	Neuron	41
3.1.2	Activation Functions	41
3.1.3	Objective Functions	43
3.1.4	Training	44
3.1.5	Neural Network Normalization	45
3.2	Convolutional Neural Networks	47
3.2.1	Transposed Convolution	48
3.3	Transformer Networks	49
3.3.1	Attention Mechanism	49
3.3.2	Transformer Architecture	50
3.4	Convolutional Neural Networks for Multi-Person Pose Tracking	53
3.4.1	Backbone Networks	53
3.4.2	Multi-Person Pose Estimation and Tracking	55
3.5	Person Re-Identification and Person Search	57
3.6	Evaluation Metrics	60
3.6.1	2D Multi-Person Pose Estimation	60
3.6.2	Person Re-Identification and Person Search	61
3.6.3	Multi-Target Tracking	62

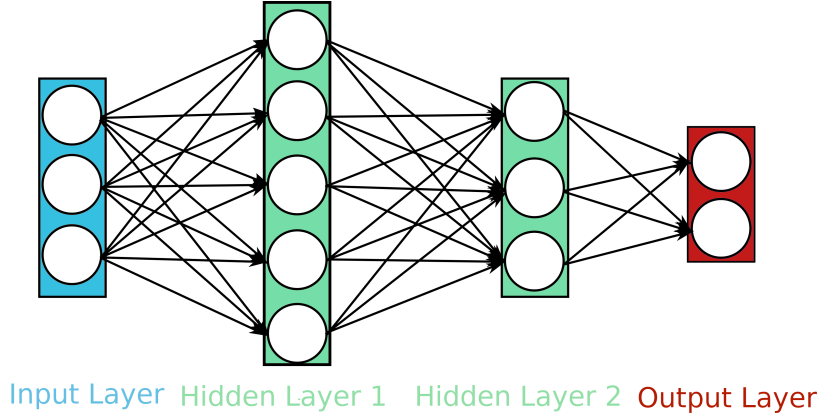


Figure 3.1: Illustration of a MLP with two hidden layers.

3.1 Neural Networks

The human brain is a large, interconnected network of neurons that process information by transmitting signals to each other through synapses (Lecun et al., 1998). Each neuron can form thousands of synaptic connections to other neurons, facilitating a wide range of cognitive functions and behaviors. Artificial Neural Network (ANN) draw inspiration from biological neural networks (Kriesel, 2007) and have been instrumental in the fast advancement of artificial intelligence in recent years.

Like biological neural networks, an ANN is a machine learning model consisting of interconnected layers, where each layer processes information received from the previous layer, which is then passed to a subsequent layer. The first and last layers of a neural network are referred to as input and output layers, respectively, and the intermediate layers are typically denoted as hidden layers. Each layer consists of multiple computational units referred to as neurons, and each neuron is connected to all the neurons in the previous and subsequent layers. Specifically, this type of neural networks forms a Directed Acyclic Graph (DAG) in which the neurons are the nodes and the edges are the weighted connections between nodes in different layers. These weights are adjusted during training to minimize the error of the objective function. Figure 3.1 illustrates a MLP with two hidden layers. In literature, these type of neural networks is often denoted as Feed-Forward Network (FFN) or MLP. As it has been proven by Cybenko (1989), FFNs with at least one hidden layer are universal function approximators that are capable of approximating any continuous function $f^* : \mathbb{R}^M \rightarrow \mathbb{R}^N$ with $|f^*(\mathbf{x}) - f(\mathbf{x})| < \varepsilon$ within a certain bound ε .

Over the recent years, neural networks have exhibited remarkable success across diverse fields such as Computer Vision, Autonomous Systems, Natural Language Processing (NLP), and Healthcare. This robust development has catalyzed a monumental transformation, establishing neural networks as indispensable components in everyday applications. Through these advancements, neural networks have significantly impacted technological progress, opening new avenues for research and application.

A summary of several fundamental building blocks of ANNs is provided in the following.

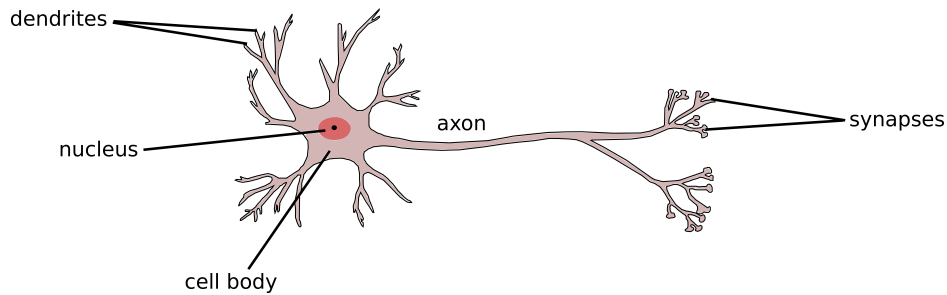


Figure 3.2: Illustration of a biological neuron. An incoming stimulus is received by the dendrites and processed by the cell body. Depending on the incoming signal, the neuron transmits information through the axon to subsequent neurons. Image adopted from [Doering \(2018\)](#).

3.1.1 Neuron

Neurons are the core building blocks of the human brain and, consequently, of ANNs. As depicted in Figure 3.2, a neuron acquires signals from its dendrites, which are subsequently processed by the cell body. Intricate chemical operations within the synaptic cleft enable the neuron to retain information ([Kriesel, 2007](#)). Depending on the incoming stimuli, the neuron gets activated and conveys a signal via the axon to other neurons of the neural network. Inspired by its biological counterpart, an artificial neuron (denoted as a neuron) can be viewed as a system that applies a transformation to its inputs and produces an output, which is referred to as activation. Formally, an individual neuron follows a linear projection $g : \mathbb{R}^d \rightarrow \mathbb{R}$, for a d -dimensional input $\mathbf{x}$,

$$o = \mathbf{w}^\top \cdot \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix}, \quad (3.1)$$

$$a = f(o), \quad (3.2)$$

where $\mathbf{w} \in \mathbb{R}^{d+1}$ denotes the learned projection weights of the neuron and $o \in \mathbb{R}$ denotes the aggregated stimulus. The weights $\mathbf{w}$ represent the inherent knowledge of the neuron. The bias term in (3.1) simulates the activation threshold of a biological neuron to convey a signal to other neurons of the network. The activation function f in (3.2) is a non-linear function that computes a new stimulus, which is then transferred to subsequent neurons. Thus, an individual neuron possesses the capacity to represent foundational boolean operations, such as AND, OR, and NOT ([Kriesel, 2007](#)).

3.1.2 Activation Functions

Activation functions are a fundamental neural network component that introduces non-linearity. Without non-linearity, the entire network would act like a single-layer linear model, independent of its depth. While the choice of activation functions for hidden layers is primarily driven by factors like introducing non-linearity, ensuring computational efficiency, and mitigating the vanishing or exploding gradient issues, it plays a pivotal role in determining the performance of a neural network. Moreover, the selection of activation functions for the output layer is task-dependent, being closely aligned with the specific objectives of the network, such as classification or regression, and the nature

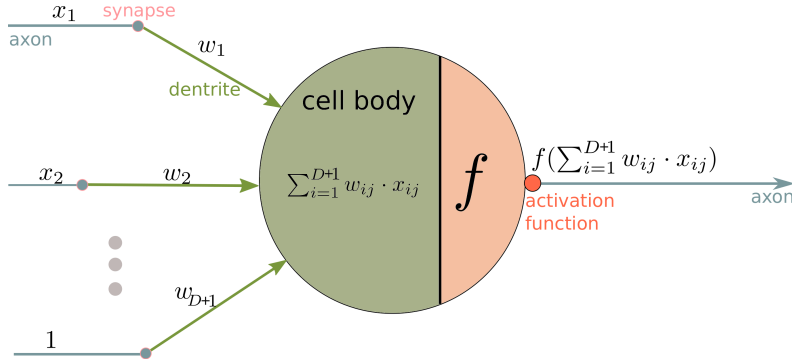


Figure 3.3: Illustration of a technical neuron. The incoming input (stimulus) is weighted at the synapse. All weighted stimuli are linearly combined within the cell body. The activation of a technical neuron is simulated by an activation function that operates on the received and linearly combined stimulus. Image adopted from [Doering \(2018\)](#).

of the data at hand.

Figure 3.4 visualizes the most common activation functions in neural networks. The Rectified Linear Unit (ReLU) ([Fukushima, 1969](#)) and its variants such as Leaky ReLU ([Maas, 2013](#)) are some of the most commonly used activation functions

$$f_{\text{ReLU}}(o) = \max(0, o), \quad (3.3)$$

$$f_{\text{LeakyReLU}}(o) = \begin{cases} o & \text{if } o \geq 0 \\ \alpha \cdot o & \text{if } x < o \end{cases}, \quad (3.4)$$

where α defines the slope of the function for negative inputs. As elaborated in [Glorot et al. \(2011\)](#), ReLUs incorporate sparsity effects within the network, potentially resulting in more robust and effective models. Moreover, [Glorot et al. \(2011\)](#) demonstrates that a hard saturation at 0 doesn't adversely affect the optimization process, provided that not all neurons are saturated. The hyperbolic tangent function *tanh* as depicted in Figure 3.4b,

$$f_{\text{tanh}}(o) = \frac{e^o - e^{-o}}{e^o + e^{-o}}, \quad (3.5)$$

is another frequently utilized activation function. It is symmetric around zero and is often employed in Recurrent Neural Networks (RNNs), (e.g. in Long Short-Term Memory (LSTM) ([Hochreiter and Schmidhuber, 1997](#))), due to its bounded nature and smooth differentiability.

The linear activation functions (Figure 3.4 c))

$$f_{\text{lin}}(o) = o, \quad (3.6)$$

is suitable for regression tasks where the output is real-valued. As for classification tasks, the choice of activation functions is bound to the number of classes. The *sigmoid* function

$$f_{\text{sigmoid}}(o) = \frac{1}{1 + e^{-o}} \quad (3.7)$$

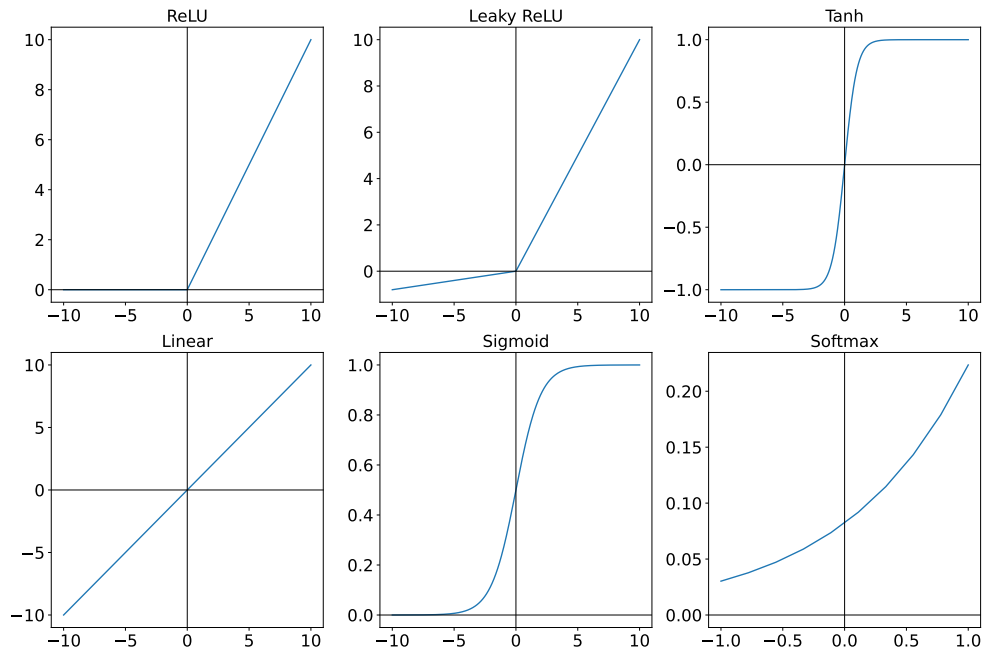


Figure 3.4: Overview of the most commonly used activation functions in neural networks.

maps the real-valued input within a range between 0 and 1. While this function is unsuitable for hidden layers due to issues like vanishing gradients, it is commonly used in the output layer for binary classification problems. For multi-class classification problems, the most common activation function is *softmax*. This function is defined as

$$f_{\text{softmax}}(o_i) = \frac{e^{-o_i}}{\sum_j^N e^{-o_j}}, \quad (3.8)$$

for any $o_j \in \mathbb{R}$.

3.1.3 Objective Functions

Training a neural network aims to optimize its parameters, specifically the weights and biases, to achieve the most accurate output possible. This is achieved by comparing the network's outputs against the given target values, commonly called the ground truth. To align the network's output with the ground truth, an optimization process minimizes a pre-defined objective function, also denoted as loss or cost function. The objective function quantifies the discrepancy between the network's output and the target data. The optimization process follows the negative gradient of the loss function to adapt the network parameters iteratively. Thus, an objective function must be differentiable.

Various common loss functions exist, but the choice of objective function is task-specific. One of the most common loss functions for regression tasks include the Mean Squared Error (MSE) and the Mean Absolute Error (MAE)

$$\mathcal{L}_{\text{MSE}}(\mathbf{X}; \theta) = \frac{1}{N} \sum_n^N \|f(\mathbf{x}_n; \theta) - \mathbf{y}_n\|_2^2, \quad (3.9)$$

$$\mathcal{L}_{\text{MAE}}(\mathbf{X}; \theta) = \frac{1}{N} \sum_n^N \|f(\mathbf{x}_n; \theta) - \mathbf{y}_n\|_1, \quad (3.10)$$

where $\mathbf{X} \in \mathbb{R}^{N \times D}$ is the set of N training examples of dimension D , $f(\mathbf{x}_n; \theta)$ is the prediction of the neural network for the n -th sample, θ denotes the weights and biases of the neural network and $\mathbf{y}_n$ is the corresponding ground truth. Compared to MAE, the MSE is more sensitive to outliers and thus less robust to errors in the training data. This can further lead to large magnitudes of the gradients during the training process, destabilizing the training process. Further, the MSE tends to inherently bias predictions towards the mean, which can be an undesirable behavior, i.e. in probabilistic contexts. Similarly, the MAE is biased towards the median.

The cross-entropy loss is commonly selected for classification tasks as the objective function of choice. For multi-label classification tasks, the cross-entropy loss is defined as

$$\mathcal{L}_{\text{CE}}(\mathbf{X}; \theta) = -\frac{1}{N} \sum_n^N \sum_c^C y_{n,c} \cdot \log(f_c(\mathbf{x}_n; \theta)), \quad (3.11)$$

where $y_n \in \{0, 1\}^C$ denotes a one-hot vector representing the target class for C classes, $y_{n,c}$ is the ground truth value for class c of training sample n and $f_c(\mathbf{x}_n; \theta)$ the respective network prediction. In the case of binary classification, the cross-entropy loss simplifies to

$$\mathcal{L}_{\text{CE}}(\mathbf{X}; \theta) = -\frac{1}{N} \sum_n^N y_n \cdot \log(f(\mathbf{x}_n; \theta)) + (1 - y_n) \cdot \log(1 - f(\mathbf{x}_n; \theta)), \quad (3.12)$$

with $y_n \in \{0, 1\}$.

3.1.4 Training

The training process is data-driven and has the goal to adjust the network parameters θ_t (i.e. the weights and biases as defined in Section 3.1.1) that minimize the objective function $\mathcal{L}$. In particular, the objective function defines an error surface with respect to the network parameters.

Starting from an initial set of network parameters, the optimization process follows the error surface in the direction of the steepest gradient leading to a local, or in the optimal case, a global minimum. Gradient decent-based optimization is most commonly used in training ANNs, and the network parameters are iteratively adjusted following

$$\theta^{t+1} = \theta^t - \eta \frac{\partial \mathcal{L}(\mathbf{X}; \theta^t)}{\partial \theta^t}, \quad (3.13)$$

where θ^t and θ^{t+1} denote the network weights at iteration t and $t + 1$, respectively. $\mathcal{L}(\mathbf{X}, \theta^t)$ is the

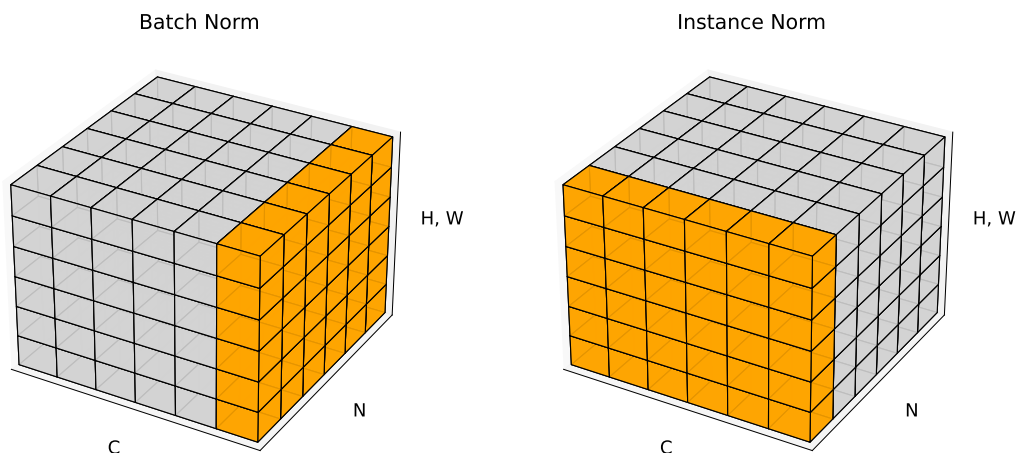


Figure 3.5: Illustration of batch normalization and instance normalization. While batch normalization normalizes across the batch dimension, instance normalization normalizes across the spatial dimensions of each sample.

error measured by the objective function for training samples $\mathbf{X}$ and $\eta \in \mathbb{R}$ is the learning rate. The choice of the learning rate is a crucial hyperparameter: A small learning rate results in more precise parameter adjustments but requires more steps till convergence, and the optimization process can get stuck in a local minimum. On the contrary, a high learning rate can overshoot the global optimum and can lead to an unstable optimization.

The backpropagation algorithm (Rumelhart et al., 1986; Lecun et al., 1998) used in ANN training leverages gradient descent to propagate the errors from the output layer back to the input layer, thus allowing the optimization of the weights of the neural network. The backpropagation algorithm leverages the chain rule that allows an efficient calculation of gradients based on gradients of subsequent layers.

In practice, gradient descent is applied on mini-batches (Goodfellow et al., 2016) which are randomly selected during training. This can lead to improved generalization capabilities and an improved convergence speed. By default, the learning rate η (3.13) is constant for all network parameters. More recent optimization algorithms such as Adam (Kingma and Ba, 2014) and AdamW (Loshchilov and Hutter, 2019) compute adaptive learning rates for each parameter using estimates of the first and second moments of the gradients, leading to faster and more efficient convergence. For more details, the interested reader is referred to (Goodfellow et al., 2016).

3.1.5 Neural Network Normalization

While training deep neural networks, the hidden layers can produce values with highly varying magnitudes, which can negatively impact the speed of convergence and can lead to overfitting. While adaptive optimizers such as Adam (Kingma and Ba, 2014) can adjust the learning rate for individual layers and neurons, deep neural networks are sensitive to parameter initialization and require low learning rates. Thus, normalization techniques inside the network, i.e. between hidden layers, are essential to mitigating the internal covariate shift, ensuring more stable distributions among layers,

which leads to improved gradient flows and consequently leads to faster convergence and a reduced sensitivity to initialization. In recent years, multiple normalization layers have been proposed (Ba et al., 2016; Ioffe and Szegedy, 2015; Ulyanov et al., 2016; Wu and He, 2018) that mainly differ in the scope of their application within a network, the dimensions along which they normalize and their impact on the training dynamics and final performance. Figure 3.5 illustrates two of the most common normalization layers and the respective normalized dimensions, where C denotes the number of feature channels, N is the number of samples within a mini-batch, and H, W denotes the height, and width of the input features.

Batch Normalization The batch normalization layer processes input features by deducting the mean and dividing by the standard deviation of the current minibatch. This step ensures that the features have zero mean and unit variance. After normalization, the batch normalization layer employs two parameters, a scaling factor γ and a shifting factor β , that allow to move the distribution of data in a direction favorable for optimal learning. Both parameters are learned during the network training process. Formally, let $\mathbf{X}_B$ be a minibatch and let $\mathbf{x} \in \mathbf{X}_B$ be the input to a batch normalization layer. The batch normalization layer applies the following function

$$BN(\mathbf{x}) = \gamma \odot \frac{\mathbf{x} - \mu_B}{\sigma_B} + \beta \quad (3.14)$$

where γ and β represent elementwise scaling and shifting, respectively. μ_B and σ_B are the sample mean and standard derivation of the minibatch $\mathbf{X}_B$, respectively, defined as

$$\mu_B = \frac{1}{|\mathbf{X}_B|} \sum_{\mathbf{x} \in \mathbf{X}_B} x \quad \text{and} \quad \sigma_B^2 = \frac{1}{|\mathbf{X}_B|} \sum_{\mathbf{x} \in \mathbf{X}_B} (x - \mu_B)^2 + \varepsilon, \quad (3.15)$$

where $\varepsilon > 0$ is a small constant to prevent a division by zero in (3.14). Practically, batch normalization layers act as regularizers during training: Small batch sizes inject high variance, whereas large batch sizes have a small regularization effect due to stable estimates of the data statistics. Consequently, smaller batch sizes can lead to better generalization and help to overcome local minima. Larger batch sizes, on the other hand, can lead to faster convergence in terms of the number of epochs required. During inference, the batch normalization layer is not normalized by batch statistics but by dataset statistics.

Layer Normalization While batch normalization can effectively be applied to FFNs and convolutional neural networks (cf. Section 3.2), it can not be applied to RNNs as this requires running averages of the summed input statistics for each timestep, especially for varying sequence length. Similarly, batch normalization is not suitable for transformer architectures (cf. Section 3.3), architectures with variable or small batch sizes, or online learning tasks (Ba et al., 2016). To overcome the aforementioned issues, layer normalization (Ba et al., 2016) directly determines the normalization statistics from the summed input overall feature channels (or, in other words, over all neurons within a hidden layer), as illustrated in Figure 3.5.

Following the same notation as for batch normalization, layer normalization for a vector $\mathbf{x} \in \mathbf{X}_B$

of dimension C is defined as

$$LN(\mathbf{x}) = \frac{\mathbf{x} - \mu_L}{\sigma_L}, \quad (3.16)$$

where mean and variance are scalar values applied to each dimension individually and are defined as

$$\mu_L = \frac{1}{C} \sum_c x^{(c)} \quad \text{and} \quad \sigma_L^2 = \frac{1}{C} \sum_c (x^{(c)} - \mu_L)^2 + \varepsilon. \quad (3.17)$$

As for batch normalization, ε is a small constant that prevents divergence, and $x^{(c)}$ denotes the value of $\mathbf{x}$ at dimension c . In their work, [Ba et al. \(2016\)](#) further shows that layer normalization remains unaffected by feature shifting and scaling per training-case basis. The interested reader is referred to [Ba et al. \(2016\)](#).

3.2 Convolutional Neural Networks

FFNs can only process one-dimensional data and thus require a vectorized input, which results in loss of spatial information. Moreover, each neuron in a layer of an FFN is connected to all neurons of the previous layer. Ultimately, this can result in a large memory footprint. Convolutional neural networks (CNNs) ([Le Cun et al., 1989](#)) are a class of neural networks with at least one convolutional layer. One of the earliest references for CNNs is already mentioned in [Fukushima \(1980\)](#). Overall, CNNs draw inspiration from biology and aim to leverage the structure of multi-dimensional data (e.g. in 2-dimensional images) and exploit the idea of locality and spatial invariance. In particular, each neuron in a convolutional layer is only connected to a local region of the previous layer. Thus, each convolutional layer implements a sparse, linear operation comparable to a sliding window applied to the activations of the previous layer. Moreover, sparsity is introduced by sharing the weights among neurons. The weights $\mathbf{K} \in \mathbb{R}^{h \times w}$ are denoted as kernel, with kernel size $\sigma_{kernel} = (h, w)$, where h, w denote the height and width, respectively. Every convolutional layer implements a discrete convolution, defined as

$$(\mathbf{I} * \mathbf{K})(i, j) = \sum_m \sum_n \mathbf{I}(i \cdot s - m, j \cdot s - n) \cdot \mathbf{K}(m, n), \quad (3.18)$$

that applies the kernel to all spatial locations i, j of the input image $\mathbf{I}$, while $m, n \in \mathbb{Z}$ and s is the stride parameter. The stride controls the spatial dimension of the output and thus balances computational efficiency. The stride choice is task-specific and typically set to $s = 1$. Larger strides can increase the robustness of small translations in the input data.

Convolutional neural networks typically consist of multiple layers that aggregate local information to capture longer-range features and a holistic representation of the entire input image. Research has shown that early layers represent low-level concepts such as edges, whereas deeper layers capture high-level concepts ([Goodfellow et al., 2016](#); [Zhang et al., 2023a](#)). CNNs have shown tremendous success in various fields of computer vision. Even though LeNet ([Lecun et al., 1998](#)) is one of the first CNNs for handwritten digit recognition, AlexNet proposed by [Krizhevsky et al. \(2012\)](#) was the first work to show superior performance over traditional approaches and marked the start of the deep

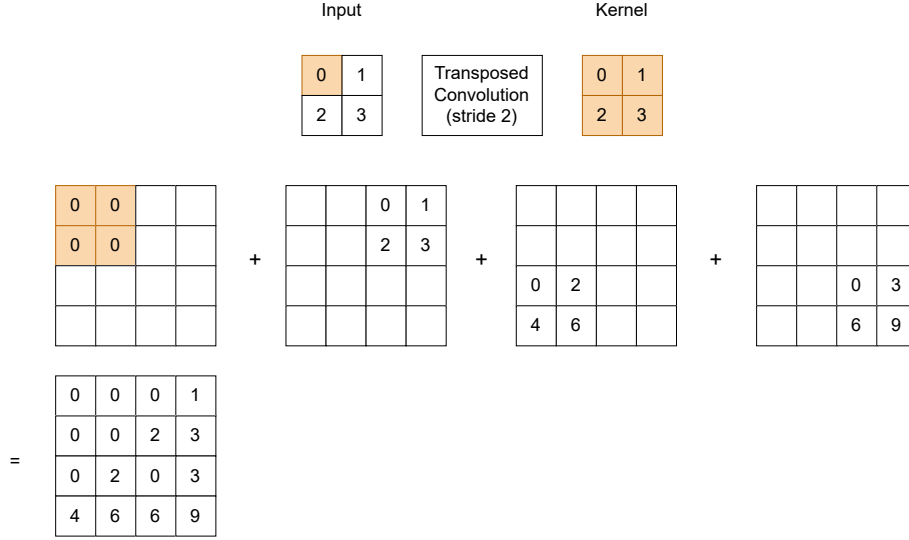


Figure 3.6: Illustration of transposed convolution with kernel size 2 and stride 2.

learning area. CNNs such as ResNet (He et al., 2016), Inception Szegedy et al. (2015) or HRNet (Sun et al., 2019a) are commonly used backbones for many state-of-the-art methods.

Pooling Pooling layers can be viewed as a parameter-free (i.e. no trainable weights) subclass of convolutional layers that incorporate similar hyperparameters such as stride s and kernel size σ_{kernel} . Pooling layers replace the input with local summary statistics of nearby input values and thus approximate spatial invariance to small translations (Goodfellow et al., 2016). The most common pooling operations are *average* pooling and *max* pooling (Zhou and Chellappa, 1988). Additionally, pooling allows the handling of variable-size inputs, which can be particularly useful for classification tasks and increase computational efficiency for strides $s > 1$.

3.2.1 Transposed Convolution

CNNs typically employ several convolutional and pooling layers to reduce the spatial dimension of the input data, i.e. a ResNet50 (He et al., 2016) has a downscaling factor of 32×32 . Even though this is favorable regarding complexity and memory footprint, some applications, such as semantic segmentation or multi-person human pose estimation, predict at a pixel level. The transposed convolution layer, also known as fractionally-strided convolution (Dumoulin and Visin, 2016) implements an upsampling operation with learnable weights, similar to a convolutional layer, except that the convolution is performed in the opposite direction. Instead of applying the kernel in a sliding-window fashion, the transposed convolution can be viewed as sliding the input over the kernel. Figure 3.6 illustrates an example for a transposed convolution with kernel size $\sigma_{kernel} = 2$ and stride $s = 2$.

3.3 Transformer Networks

Transformer networks mark a significant leap forward in deep learning, particularly in Natural Language Processing (NLP) and, more recently, in computer vision. The pioneering efforts of Vaswani et al. (2017) on the transformer model have dramatically propelled progress in NLP and computer vision, reshaping research in these areas. This has enabled new applications and breakthroughs previously unattainable with earlier methods and architectures. Before transformer networks, the field of NLP was dominated by RNNs and CNNs, which suffered from limited capabilities in handling long-range dependencies in the input data.

The architecture of the transformer network is centered around an attention mechanism, which allows the model to attend to different parts of the input sequence and thus enables capturing context and long-range dependencies more effectively. In contrast to RNNs, transformers parallelly process all input data elements, allowing efficient training and inference on modern hardware. Apart from their remarkable performance, transformer networks have a high computational demand and require large quantities of data. The success of transformer networks can, therefore, be attributed to the rapid increase in computing power that allowed to scale transformer networks up to *trillions* of parameters (i.e. 1.7 trillion parameters for GPT-4 (OpenAI, 2023)).

3.3.1 Attention Mechanism

While various attention mechanisms have been introduced in previous research (Kim et al., 2017; Parikh et al., 2016), this thesis primarily examines the approach presented by (Vaswani et al., 2017), which is widely recognized as the prevailing attention mechanism in contemporary state-of-the-art methodologies. Echoing the concepts of earlier studies, this attention mechanism is designed to replicate aspects of human cognitive processes, particularly the ability to concentrate on specific elements within an environment while disregarding others. Within computer vision, this translates to an emphasis on particular regions of pixel data.

From a technical standpoint, the attention mechanism can be conceptualized as a specialized function that allocates variable weights to different parts of the input data. This allocation is based on the relevance or importance of these parts in the context of a specific query. In other words, the output of the attention function is a weighted sum of a sequence of input vectors $\mathbf{V}$ denoted as *values*. The weights, often referred to as attention weights, are obtained by a scaled dot-product between a sequence of query $\mathbf{Q}$ and key $\mathbf{K}$ vectors normalized by a softmax. Formally, the scaled dot-product attention is defined as

$$\text{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax} \left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{C_k}} \right) \mathbf{V}, \quad (3.19)$$

where $\mathbf{Q} \in \mathbb{R}^{M \times C_K}$, $\mathbf{K} \in \mathbb{R}^{N \times C_K}$ and $\mathbf{V} \in \mathbb{R}^{N \times C_V}$ are matrices. M and N represent the sequence lengths, while C_K and C_V correspond to the feature dimensions, respectively.

When the value of C_K is large, it can result in significantly large values in the dot product. This, in turn, might cause training instabilities, such as vanishing gradients or extremely sharp distributions from the softmax function. To mitigate this, Vaswani et al. (2017) suggest scaling the dot product by the factor $\frac{1}{\sqrt{C_K}}$.

3.3.1.1 Multi-Head Attention

The multi-head attention mechanism comprises h unique attention layers, each referred to as a head. These heads function independently, focusing on distinct representation subspaces and various positions within the data. Such a design facilitates the representation of various data relationships, including sequential dependencies, thereby enabling a more comprehensive and nuanced interpretation of the input data.

Each attention head incorporates learnable, linear projections that typically project the query, key, and value matrices into a lower dimensional subspace, i.e. $C_K = C_V = C_{in}/h$, where C_{in} is the dimension of the input features. By projecting the dimension of each head into a fraction of the input dimension, the overall computational costs for multi-head attention are similar to a single-head attention layer. More precisely, the output of each attention head i is defined as

$$head_i = Attn(\mathbf{QW}_Q^i, \mathbf{KW}_K^i, \mathbf{VW}_V^i), \quad (3.20)$$

with learnable projection weights $\mathbf{W}_Q^i \in \mathbb{R}^{C_{in} \times C_K}$, $\mathbf{W}_K^i \in \mathbb{R}^{C_{in} \times C_K}$ and $\mathbf{W}_V^i \in \mathbb{R}^{C_{in} \times C_V}$. Multi-head attention further incorporates an output layer with weights $\mathbf{W}_O \in \mathbb{R}^{h \cdot C_K \times C_{in}}$ that concatenates and projects the results of each attention head $head_i$ according to

$$MultiHeadAttn(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = [attn_1 | \dots | attn_h] W_O. \quad (3.21)$$

3.3.2 Transformer Architecture

The transformer architecture as proposed by Vaswani et al. (2017) comprises an auto-regressive encoder-decoder structure that takes as input a sequence of data $\mathbf{X} \in \mathbb{R}^{N \times C}$, where N denotes the sequence length and C is the dimension of the input features. In the context of NLP, the input sequence is typically a set of learnable tokens, whereas in the context of computer vision, the input sequence is typically a vectorized image with $N = H \cdot W$, where H and W denote the image height and widths, respectively. The auto-regressive nature of the transformer allows the incorporation of all previously generated outputs when generating the next step. Figure 3.7 illustrated the entire transformer architecture. The transformer performs parallel computations and thus lacks inherent sequence awareness, i.e. the transformer can not distinguish between different arrangements of pixels or words and consequently can not properly reason about the context of arrangements. Therefore, positional information needs to be encoded into the input data.

Positional Encoding One prominent way to preserve spatial and temporal information is to associate an additional positional encoding with the input data. These positional encodings are either pre-defined or can be learned. While both ways can yield similar performance, pre-defined positional encodings can aid the model in generalizing to the unseen sequence length. One of the most widely used positional encodings, as proposed by Vaswani et al. (2017), is based on sine and cosine functions and is defined as

$$\begin{aligned} E_{pos}(n, 2c) &= \sin\left(n \cdot 10000^{\frac{-2c}{C}}\right) \\ E_{pos}(n, 2c + 1) &= \cos\left(n \cdot 10000^{\frac{-2c}{C}}\right), \end{aligned} \quad (3.22)$$

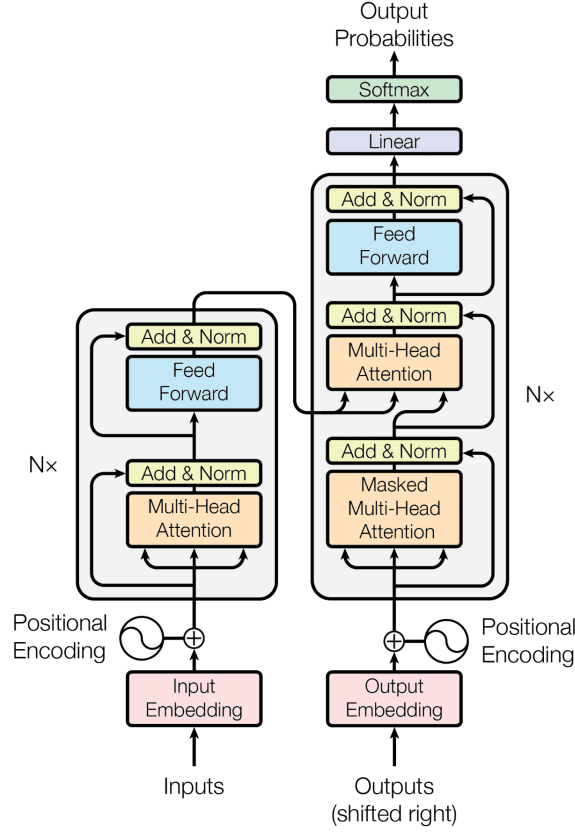


Figure 3.7: Illustration of the transformer architecture. The input features are augmented with positional encodings to preserve spatiotemporal information and then processed by N encoder layers. Each layer is comprised of multi-head self-attention and feed forward layers. Similarly, output embeddings are processed by N decoder layers. Each decoder layer comprises an additional multi-headed cross attention layer that takes as input the encoded input embeddings from the encoder. Figure from Vaswani et al. (2017)

where $n \in [0, N]$ denotes the n^{th} token and c denotes the c^{th} feature dimension. The positional encoding for input length data $N = 100$ and feature dimension $C = 512$ is illustrated in Figure 3.8.

Transformer Encoder The transformer encoder stage comprises K encoder layers, each containing a multi-head attention mechanism and a FFN. These sub-layers are utilized in a residual connection setup, followed by layer normalization (Ba et al., 2016), as depicted in Figure 3.7. Within each encoder layer, the attention mechanisms function as self-attention layers, meaning that the queries, keys, and values are all derived from the same input feature set. This design enables the self-attention layers to learn long-range dependencies within the input sequence, regardless of its length. In essence, self-attention facilitates the model's ability to capture and process relationships between different parts of the input, enhancing its understanding and representation of the data.

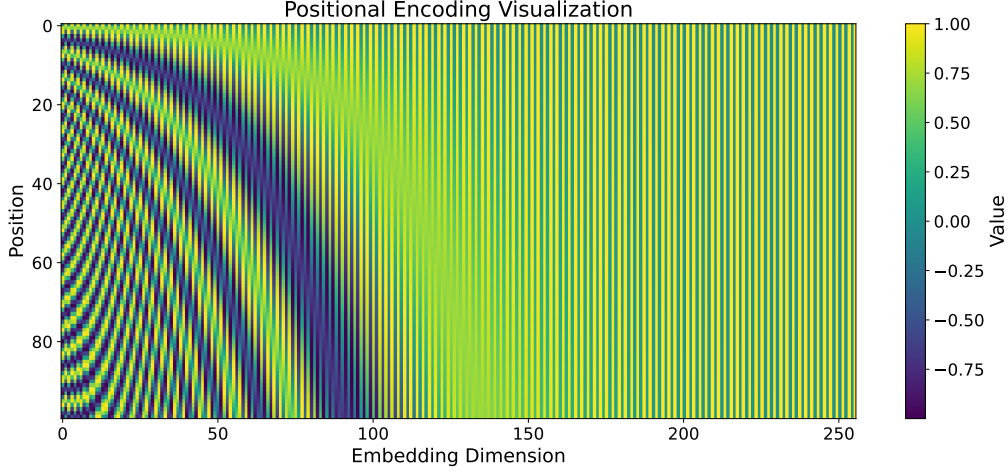


Figure 3.8: Illustration of a positional encoding as defined in (3.22) for an input sequence of length $N = 100$ and a feature dimension of $C=256$.

Transformer Decoder The transformer decoder stage follows a similar structure and consists of K decoder layers, each comprising three residual sub-layers. The first sub-layer is a multi-headed self-attention layer that encodes the long-range dependencies of a sequence of tokens $\mathbf{X}_D \in \mathbb{R}^{M \times C}$. Typically, $\mathbf{X}_D$ is a sequence of M learnable embeddings representing certain concepts, i.e., a word or a sub-word in the context of NLP. Unlike the encoder, the self-attention layer implements a causal self-attention that only considers the current and previous inputs, thus ensuring sequence awareness. This is achieved by masking the upper triangle of the attention matrix before the softmax.

The second sub-layer is a cross-attention layer which applies attention between $\mathbf{X}_D$ and the output of the encoder $\mathbf{X}_E \in \mathbb{R}^{N \times C}$. This enables the decoder to focus on specific parts of the input sequence X_E , which is important for tasks such as translation (i.e., RGB values to bounding box coordinates). Finally, the FFN is applied to each token separately.

Overall, each decoder layer captures different aspects of information. While self-attention contributes to understanding the structure of the query sequence, cross-attention relates the encoded query sequence to the input sequence.

3.3.2.1 Feed Forward Network

The FFN used in all encoder- and decoder-layers consists of two linear layers and employs an inverted bottleneck structure, which is applied to all encoded tokens in parallel. The first layer with weights $W_1 \in \mathbb{R}^{C \times sC}$ expands the input dimension by a pre-defined factor $s \in \mathbb{N}$ into a higher-dimensional feature space to facilitate the model's ability to learn complex patterns. The second layer with weights $W_2 \in \mathbb{R}^{sC \times C}$ projects the features back into the input dimension following

$$FFN(\mathbf{x}) = \max(0, xW_1 + b_1)W_2 + b_2, \quad (3.23)$$

where $b_1 \in \mathbb{R}^{sC}$ and $b_2 \in \mathbb{R}^C$ denote bias terms, respectively.

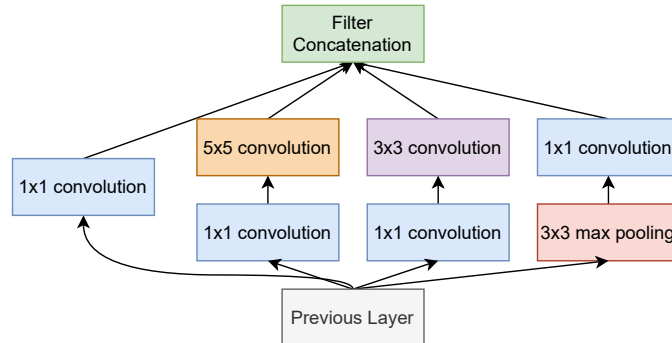


Figure 3.9: Illustration of an inception block.

3.4 Convolutional Neural Networks for Multi-Person Pose Tracking

CNNs have shown tremendous success in many computer vision applications such as multi-person pose estimation (Cai et al., 2020; Kreiss et al., 2019; Newell et al., 2016), object detection (Ren et al., 2015; Cai and Vasconcelos, 2017; Redmon et al., 2016) or person re-identification (Luo et al., 2019b; Zheng et al., 2017a; Miao et al., 2019). In the context of multi-person pose estimation and tracking, various network architectures have been proposed, such as multi-stage networks (Cao et al., 2017; Newell et al., 2016), multi-branch networks (Sun et al., 2019a), residual networks (Xiao et al., 2018; Newell et al., 2016) and various combinations of those techniques. All these architectures follow a pre-training and finetuning paradigm and build on top of pre-trained backbone networks such as VGG (Simonyan and Zisserman, 2015), Inception networks (Szegedy et al., 2015, 2016) or ResNets (He et al., 2016). The backbone networks are usually pre-trained for image classification on ImageNet (Deng et al., 2009). However, apart from a faster convergence on downstream tasks, research indicates (He et al., 2019a) that pre-trained models provide no further benefits such as regularization. In Chapter 6 we observed a similar behavior for person re-identification.

In the following Section 3.4.1, we summarize the most relevant CNNs backbone networks in the context of this thesis for downstream tasks like multi-person pose estimation and tracking, person re-identification, and person search.

3.4.1 Backbone Networks

VGG The VGG architecture (Simonyan and Zisserman, 2015) was developed as an improvement of AlexNet (Krizhevsky et al., 2012) to reduce the total number of parameters. AlexNet utilizes convolution layers with various kernel sizes (11×11 , 5×5 , and 3×3), where large kernel sizes were introduced to capture a large receptive field. Alternatively, the receptive field of a large kernel can also be replicated by multiple layers of a smaller kernel. For example, two layers of 3×3 convolutions have the same receptive field as a single layer of 5×5 convolution. Moreover, two layers of 3×3 convolutions require fewer parameters than a single 5×5 layer.

The VGG network relies on this intuition and contains a fixed kernel size of 3×3 and pooling layers with a kernel size of 2×2 . A reduced number of trainable parameters not only saves computational resources but also results in faster convergence.

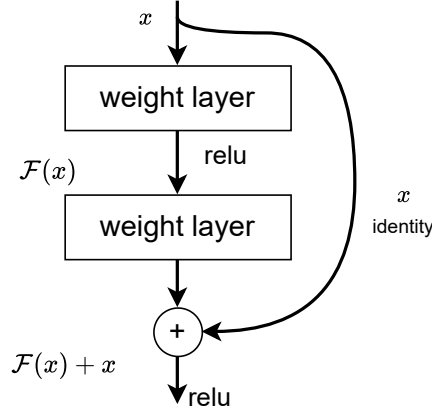


Figure 3.10: Illustration of a residual block.

Inception The inception architecture (Szegedy et al., 2015, 2016) increases the width of a CNN rather than expanding the depth of the network. As illustrated in Figure 3.9, each inception block consists of multiple parallel convolutional layers with different kernel sizes that individually process the input features. The final layer is a depth-wise concatenation of all resulting output features. Consequently, each inception module captures relevant features with different sizes of receptive fields. While 5x5 kernel sizes capture global information, 3x3 kernels capture local information, while the pooling layer extracts the most salient local features. All extracted features are concatenated and passed to the succeeding layer. This allows downstream layers to focus on the most relevant information for solving a given task. In Chapter 4, we propose a modified inception architecture for the task of multi-person pose estimation.

Residual Networks One of the most important backbone architectures used for various computer vision tasks is the ResNet proposed by He et al. (2016), which allows the circumventing of various problems associated with increasing depth of CNNs. As observed in (He and Sun, 2015; Srivastava et al., 2015), the performance of deep networks saturates at a certain depth and rapidly decreases with increasing depth. This rapid decrease in performance can be linked to the problem of vanishing gradients preventing early layers of the network from being optimized. Moreover, deeper networks have a reportedly higher training error (He and Sun, 2015; Srivastava et al., 2015; He et al., 2016). As a solution, He et al. (2016) introduce residual blocks that incorporate “skip connections”, also called identity mappings, as illustrated in Figure 3.10. Formally, let $\mathcal{H}(x)$ be the nonlinear function learned by a block of layers. By adding a skip connection, a residual block of the underlying mapping will be shifted towards $\mathcal{F}(x) = \mathcal{H}(x) - x$, such that $\mathcal{H}(x) = \mathcal{F}(x) + x$. The advantages of this formulation are twofold. First, skip connections prevent gradients from vanishing, which allows for training much deeper networks. Moreover, residual blocks can easily learn identity mappings by driving the weights of each residual block towards zero. This is an advantage over non-residual blocks, which have to learn identity mappings through non-linearities.

The ResNet describes a family of residual networks with various depths, varying from 18 layers (ResNet18) up to 152 layers (ResNet152), which are used for various downstream tasks such as object detection (Ren et al., 2015), multi-person pose estimation (Xiao et al., 2018) or person search (Chen et al., 2020a).

3.4.2 Multi-Person Pose Estimation and Tracking

Research has made tremendous progress in multi-person pose estimation and tracking (Liu et al., 2022; Feng et al., 2023; Wang et al., 2020c), especially within the last decade. Convolutional Neural Networks (CNNs) have been the backbone of state-of-the-art models in this domain, leveraging their powerful capability to extract and learn spatial hierarchies of features from images. However, the recent shift towards transformer-based architectures signifies an evolution in the approach to these problems. Transformers, known for their effectiveness in capturing long-range dependencies within data, offer a novel perspective in handling the complexities of multi-person scenarios. By efficiently modeling relationships between keypoints across and within individuals, transformers can potentially provide more nuanced and contextually aware solutions to pose estimation and tracking.

Multi-person pose estimation and tracking are interconnected problems that build on each other. Pose estimation serves as the foundational task that enables pose tracking. Accurate estimation is critical for tracking, as errors in identifying or localizing keypoints can lead to failure to maintain consistent identities across frames. Conversely, tracking can enhance pose estimation by providing temporal context that helps disambiguate difficult poses or resolve occlusions. For instance, knowing the trajectory of a person can aid in predicting the position of occluded keypoints, thus improving the robustness of pose estimation in challenging scenarios.

3.4.2.1 Multi-Person Pose Estimation

The task of multi-person pose estimation involves identifying and localizing the anatomical keypoints (i.e. wrists, hips, knees) of all persons within the same image, typically represented as 2D Gaussians centered around each keypoint. Multi-person pose estimation involves detecting individual persons within the same frame and precisely mapping out the skeletal structure in terms of detected keypoints. As the human body has 244 degrees of freedom (Prilutsky and Zatsiorsky, 2002), the estimation of poses is a complex task, especially in real-world scenarios. This challenge further amplifies with more subjects, especially in real-world scenarios. In-the-wild videos often contain a high degree of occlusion, particularly in crowded scenarios, which leads to missing or wrongly assigned keypoints. Moreover, motion blur, high variations in scale, interacting persons or persons in very unusual poses, i.e. during sport activities pose additional challenges for multi-person pose estimation approaches.

The current body of works on multi-person pose estimation can be categorized into bottom-up and top-down approaches (cf. Section 2.1.2). Bottom-up approaches directly estimate all persons' keypoints, leading to fast inference and real-time performance, even on mobile devices. To assemble all keypoints into poses, bottom-up approaches further predict associative features that indicate correspondences between different keypoint classes. Typical features for pose assembly include associative embeddings (Newell et al., 2017), vector fields such as part affinity fields (Cao et al., 2017) or displacement vectors (Zhou et al., 2020b; Zigang et al., 2021; Artacho and Savakis, 2023; Nie et al., 2019b). Top-down approaches, on the other hand, operate on cropped bounding boxes pre-

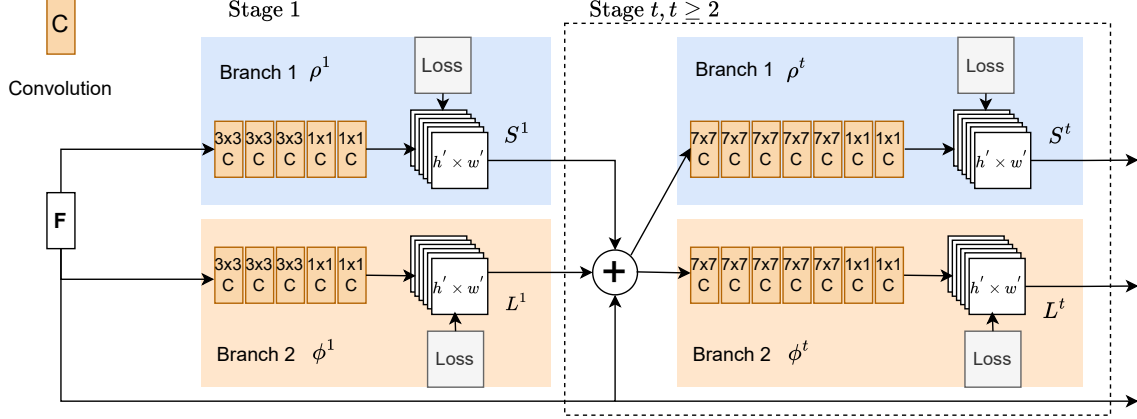


Figure 3.11: Illustration of the two-branch architecture for multi-person pose estimation proposed in [Cao et al. \(2017\)](#). Each branch processed image features $\mathbf{F}$ and estimates keypoint heatmaps $\mathbf{S}$ and part affinity fields $\mathbf{L}$. In addition to the image features, succeeding stages take as input keypoint heatmaps and part affinity fields of the previous stage.

viously obtained by a person detector. Thus, the inference time scales linearly with the number of persons within a video frame. As top-down approaches operate on a fixed scale, they mostly outperform bottom-up approaches regarding localization accuracy, which are affected by varying person scales.

Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields [Cao et al. \(2017\)](#) propose a bottom-up method that employs the first ten layers of VGG-19 ([Simonyan and Zisserman, 2015](#)) as the backbone network. The backbone network employs a global stride of 8 pixels, which has been shown to preserve most of the spatial information ([Yu et al., 2017](#); [Long et al., 2015](#)) and thus offers a good trade-off between computational costs and localization accuracy. The proposed work follows a multi-stage, multi-branch approach as illustrated in Figure 3.11. The first branch predicts keypoint heatmaps, denoted as $\mathbf{S}^1, \dots, \mathbf{S}^t$, while the second branch predicts part affinity fields (PAF) denoted as $\mathbf{L}^1, \dots, \mathbf{L}^t$. PAFs are vector fields that capture spatial relationships and orientation between predicted keypoints. Essentially, a PAF $\mathbf{L}$ represents the direction from one keypoint $\mathbf{x}_j$ to another $\mathbf{x}_k$, where each pixel along the limb consists of a 2D unit vector $v = \frac{\mathbf{x}_j - \mathbf{x}_k}{\|\mathbf{x}_j - \mathbf{x}_k\|_2}$ oriented towards the target keypoint. Formally, the PAF for limb l at pixel location $\mathbf{p} \in \mathbb{R}^2$ is defined as

$$\mathbf{L}(\mathbf{p}) = \begin{cases} v & \text{if } \mathbf{p} \text{ on limb } l \\ 0 & \text{otherwise.} \end{cases} \quad (3.24)$$

Thus, PAFs facilitate the accurate estimation of poses even in cluttered situations.

Each network stage passes the predicted keypoint heatmaps and part affinity fields along with the image features to the next stage (cf. Figure 3.11). Each succeeding stage increases the receptive field and refines the predictions of previous stages. Intermediate losses applied to the predictions of each stage prevent the gradients from vanishing. For more details, the interested reader is referred to ([Cao et al., 2017](#)).

3.4.2.2 Multi-Person Pose Tracking

Multi-person pose tracking extends multi-person pose estimation over sequential frames in a video. The goal is not only to estimate the poses in each frame but to maintain the identity of each individual over time, thus creating a trajectory of keypoints for each person. This adds another degree of complexity as pose tracking pipelines have to deal with various nuances of noise, such as motion blur, camera motion, occlusions, and variations in scale and viewpoint. Moreover, tracked poses can disappear for various frames and thus need to be properly re-associated. While early works (Iqbal et al., 2017; Andriluka et al., 2018; Insafutdinov et al., 2017) employed fully connected graphs, these methods are not applicable for real-world applications due to the large runtime. For that reason, most works (Doering et al., 2018; Xiu et al., 2018; Jin et al., 2019b; Raaj et al., 2019; Jin et al., 2019a) predict temporal associative features (e.g. warping-based methods, temporal flow fields, associative embeddings, keypoint correspondences) and employ online frame-wise greedy or Hungarian (Kuhn, 1955) matching.

Frame-wise matching can be formulated as an energy maximization problem over a bipartite graph. Let $\mathcal{P}_{t-1} = \{P_{t-1}^1, \dots, P_{t-1}^{N_{t-1}}\}$ and $\mathcal{P}_t = \{P_t^1, \dots, P_t^{N_t}\}$ be the sets of estimated poses at timesteps $t - 1$ and t , respectively. We further define $S(P_{t-1}, P_t)$ as function that measures the temporal similarity between poses P_{t-1} and P_t . We can then optimize the following objective

$$\begin{aligned} \hat{Z} &= \operatorname{argmax}_Z \sum_{P_{t-1}} \sum_{P_t} S(P_{t-1}, P_t) \cdot z_{P_{t-1}, P_t} \\ \text{s.t.} \quad &\forall P_t, \sum_{P_{t-1}} z_{P_{t-1}, P_t} \leq 1 \text{ and } \forall P_{t-1}, \sum_{P_t} z_{P_{t-1}, P_t} \leq 1, \end{aligned} \quad (3.25)$$

where z_{P_{t-1}, P_t} is a binary variable that indicates whether poses P_{t-1} and P_t are matched and Z is the set of edges between poses with maximum temporal similarity.

Temporal Flow Fields for Multi-Person Pose Tracking Doering et al. (2018) build upon the work of Cao et al. (2017) and propose to predict temporal flow fields (TFF) that allow to associate predicted keypoints between video frames. Temporal flow fields extend the idea of PAFs into the temporal domain. Unlike PAFs, TFFs predict the motion of each individual keypoint between two consecutive frames and thus are suitable for tracking poses. As illustrated in Figure 3.12a) the first stage consists of a Siamese CNN that takes as input two consecutive frames and predicts heatmaps and PAFs similar to Cao et al. (2017). Moreover, the spatial features of both frames are passed to the temporal network (Figure 3.12b)) along with the predicted heatmaps, facilitating real-time applications. The predicted TFFs are then employed as a similarity measure to solve a bipartite graph matching problem between the estimated poses of both frames I_{t-1}, I_t (as shown in (3.25)).

3.5 Person Re-Identification and Person Search

Person Re-Identification is an important task in computer vision, which aims to match images of an individual across different frames or camera views. This challenge of this task involves the extraction

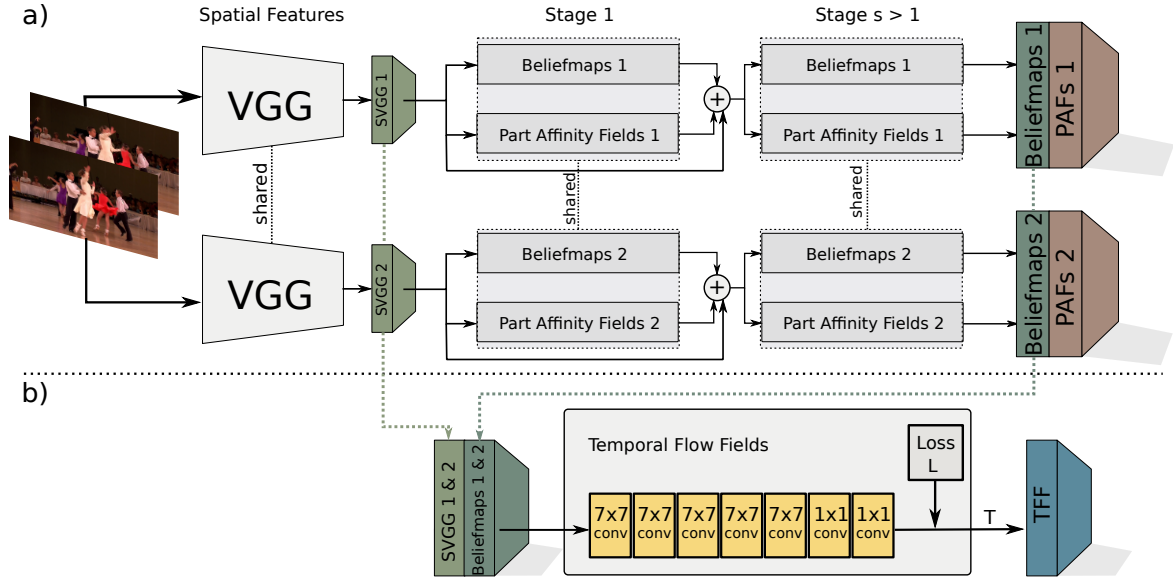


Figure 3.12: Proposed architecture of JointFlow (Doering et al., 2018). a) illustrates the Siamese pose estimation architecture, adopted from Cao et al. (2017) and b) is the architecture of the temporal flow fields prediction branch. Figure from Doering et al. (2018).

of robust, discriminative image features that are invariant to changes in appearance, such as illumination, camera perspective, occlusions, different clothing, or variations in pose. Despite the recent advancement of CNN and transformer-based networks, person re-identification remains challenging due to high intra-class variability and inter-class similarity among persons. Person re-identification and multi-person pose estimation both require robust and discriminative features for the association of individuals. Consequently, techniques developed for extracting discriminative features in Person Re-ID can be beneficial for the pose tracking task (Guo et al., 2018; Xu and Zhou, 2018), especially in maintaining identity after long periods of occlusion.

A widely used baseline for person re-identification (Hermans et al., 2017; Sun et al., 2018c; Luo et al., 2019b) employs a ResNet50 (He et al., 2016) backbone that incorporates sophisticated feature extraction and metric learning strategies. Luo et al. (2019b) incorporate multiple losses applied at different stages of their network, thereby facilitating a more structured feature space and more discriminative features. One of these losses is the triplet loss (Hermans et al., 2017)

$$\mathcal{L}_{trip} = \max(d_p - d_n + m, 0), \quad (3.26)$$

where d_p is the distance between two positive samples and vice versa, d_n is the distance between a positive and negative sample, and m denotes a margin that enforces separation. This loss function aims to increase inter-class separability and intra-class compactness. However, it may not always impose globally optimal constraints. In contrast, the ID loss divides the feature space into multiple subspaces by several hyperplanes. For instance, the ID loss for ground truth person id y is usually

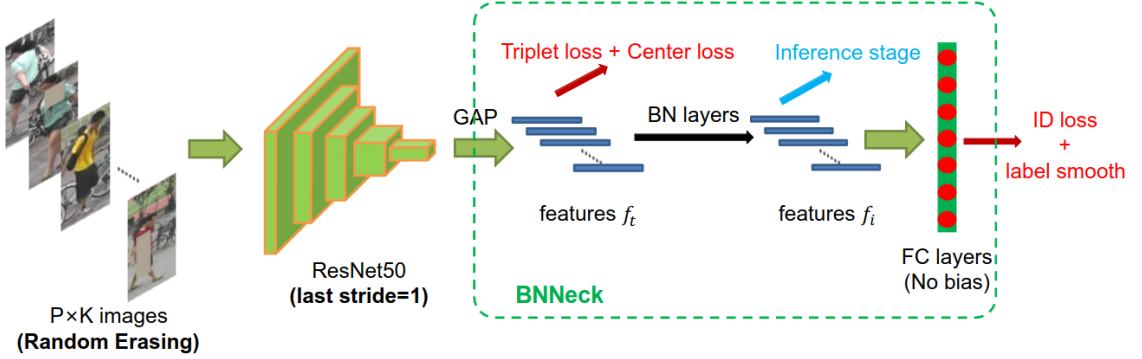


Figure 3.13: Standard person re-identification pipeline extended by Luo et al. (2019b) with a batch normalization neck. Figure from Luo et al. (2019b).

defined as the cross entropy following

$$\mathcal{L}_{ID} = - \sum_i^N q_i \cdot \log(p_i), \quad (3.27)$$

where $q_i = 1$ if $i = y$ and $q_i = 0$ otherwise and p_i is the predicted confidence for person id i . While the triplet loss optimizes the Euclidean distance, the ID loss optimizes the cosine distance. If combined during training, this can lead to conflicting behavior and sub-optimal performance. As illustrated in Figure 3.13, Luo et al. (2019b) employ a batch normalization layer after the pooling layer and before the classification layers. The batch normalization layer re-distributes the features close to the surface of a hypersphere and allows for easier convergence of the ID loss.

To remedy the drawbacks of the triplet loss and aiming to diminish intra-class variations, Luo et al. (2019b) further incorporate the Center Loss (Wen et al., 2016) defined as

$$\mathcal{L}_{center} = \frac{1}{2} \sum_i^N \|\mathbf{f}_i - c_{y_i}\|_2^2, \quad (3.28)$$

where N is the batch size, $\mathbf{f}_i$ are the features of the i -th image in the batch (before batch normalization), and c_{y_i} is the learnable centroid for the corresponding class y_i .

Person Search Person search extends the challenge of person re-identification by including the task of person detection rather than relying on pre-cropped images of individuals. This approach involves identifying and tracking individuals across a series of video frames from various scenes or camera angles by detecting each person in every frame and extracting discriminative features for re-identification. This comprehensive method introduces additional complexities, mainly because the effectiveness of detection and localization greatly influences re-identification accuracy. Therefore, person search systems must balance detection accuracy with re-identification performance.

Recent works on person search have leveraged joint frameworks that simultaneously train detection and re-identification components. Figure 3.14 illustrates two baseline methods. Figure 3.14 (a)

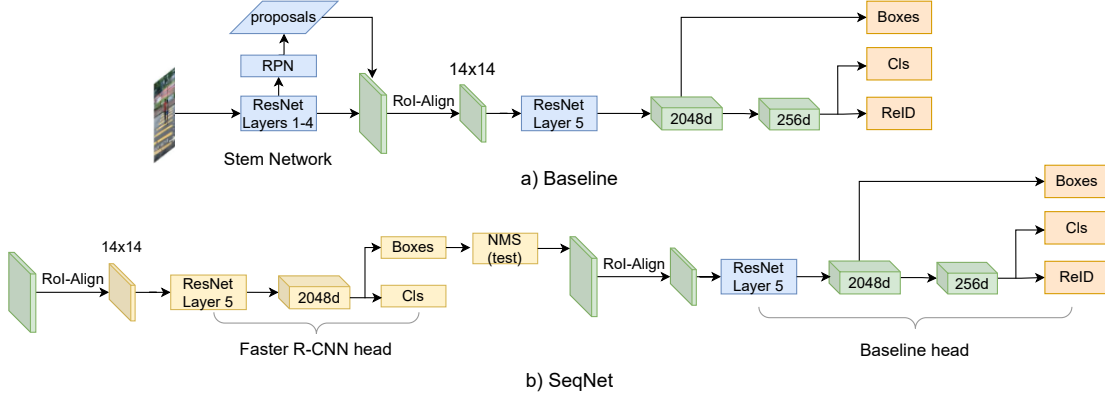


Figure 3.14: Illustration of the person search architecture proposed in SeqNet (Li and Miao, 2021). a) illustrates the baseline architecture for person approaches based on Faster R-CNN (Ren et al., 2015), as used in Chen et al. (2020b). b) SeqNet incorporates an additional Faster R-CNN head, which takes as input proposals from the RPN network (denoted in blue).

illustrates the work of Chen et al. (2020b) which extends a Faster R-CNN (Ren et al., 2015) by a norm-aware embedding (NAE) head. This norm-aware embedding head projects feature embeddings onto an n -dimensional hypersphere, facilitating a clearer distinction between relevant detections and background noise. SeqNet (Li and Miao, 2021), as shown in Figure 3.14 (b), proposes sequential prediction heads to boost the localization accuracy. The first stage (as depicted in yellow in Figure 3.14 (b)) is a Faster R-CNN head which estimates initial bounding boxes. The second stage employs the NAE head, which extracts person re-identification features and further improves the localization accuracy of previously detected bounding boxes.

In Chapter 5, we employ the work of Li and Miao (2021) as the reference method for person search on our newly introduced dataset PoseTrack21.

3.6 Evaluation Metrics

The following section briefly introduces the most common evaluation metrics for multi-person pose estimation, person re-identification, person search, multi-object tracking, and multi-person pose tracking. In Chapter 5, we introduce a new metric *Keypoint HOTA* for multi-person re-identification pose tracking.

3.6.1 2D Multi-Person Pose Estimation

Percentage of Correct Keypoints (PCK) PCK, as introduced by Sapp et al. (2011), classifies each individual body joint as correctly detected if its distance to the ground truth location is within a pre-defined threshold. In literature, multiple definitions of thresholds exist that come with various drawbacks. Sapp et al. (2011) rely on the height of a persons torso, whereas Yang and Ramanan (2013) utilize the maximum height of a persons bounding box. Both definitions are sensitive to body articulation and can become overly strict when the torso is truncated (Sapp et al., 2011) or when the person is lying down. Andriluka et al. (2014), on the other hand, propose to base the threshold on

the head size. In this way, the dependence on body articulation can be removed completely. As the metric is based on the head size, this PCK variant is often called PCKh. PCKh is one of the most commonly used metrics for 2D human pose estimation among all PCK variants.

Object Keypoint Similarity (OKS) PCK and PCKh treat each keypoint type equally, although different keypoint types tolerate different levels of precision. For example, eyes and wrists should be precisely located, whereas hips or shoulders cover a larger area. Consequently, an evaluation metric should incorporate varying thresholds for different keypoint types. For example, keypoints with small areas require a small threshold, and vice versa, keypoints with large areas require high thresholds to compensate for localization errors. Ronchi and Perona (2017) propose a Keypoint Similarity (KS) metric that measures the closeness among pairs of keypoints. KS evaluates for a detected keypoint $\mathbf{x}_i$ an unnormalized Gaussian centered around the ground-truth location $\hat{\mathbf{x}}_i$

$$KS(\hat{\mathbf{x}}_i, \mathbf{x}_i) = e^{-\frac{\|\hat{\mathbf{x}}_i - \mathbf{x}_i\|^2}{2s^2\sigma_i^2}}. \quad (3.29)$$

The standard deviation σ_i , specific to each keypoint type i , is obtained by evaluating 5000 redundantly annotated images of the MSCOCO (Lin et al., 2014) dataset. The ground truth instance area further scales the standard deviation, typically obtained from a person’s bounding box. Given a ground truth pose $\hat{\mathbf{J}}$ and a predicted pose $\mathbf{J}$, the OKS is computed as the average KS of all visible keypoints

$$\text{OKS}(\hat{\mathbf{J}}, \mathbf{J}) = \frac{\sum_i KS(\hat{\mathbf{x}}_i, \mathbf{x}_i) \delta(v_i > 0)}{\sum_i \delta(v_i > 0)}, \quad (3.30)$$

where v_i indicates visibility of ground-truth keypoint $\hat{\mathbf{x}}_i$.

mean Average Precision (mAP) Both PCK and OKS are only suitable for estimating a single-person pose. Further, both metrics do not penalize False-Positives (FPs), i.e., predicted keypoints that can not be associated with any ground truth keypoint. Mean Average Precision (mAP) as proposed by Insafutdinov et al. (2016) draws inspiration from the Average Precision (AP) and Recall (AR) metrics from object detection, originally introduced by Everingham et al. (2007): For a given set of ground truth poses, mAP greedily assigns detected poses to ground truth poses based on PCK or OKS, starting from the poses with the highest localization confidence. A ground truth pose can only be assigned with one detected pose, and all successfully assigned keypoints are categorized as True-positives (TPs). On the other hand, all estimated keypoints that can not be assigned with any ground truth keypoint are considered as FPs. Vice versa, all unassigned ground truth keypoints are regarded as False-Negatives (FNs). The AP is then calculated from the recall-precision curve obtained from the TP, FP, and FN for every keypoint class, respectively, and the mAP is the mean of all APs of each keypoint class.

3.6.2 Person Re-Identification and Person Search

Cumulative Matching Characteristic (CMC) The CMC (Gray et al., 2007) is one of the most popular evaluation metrics for person re-identification and measures the probability that a query im-

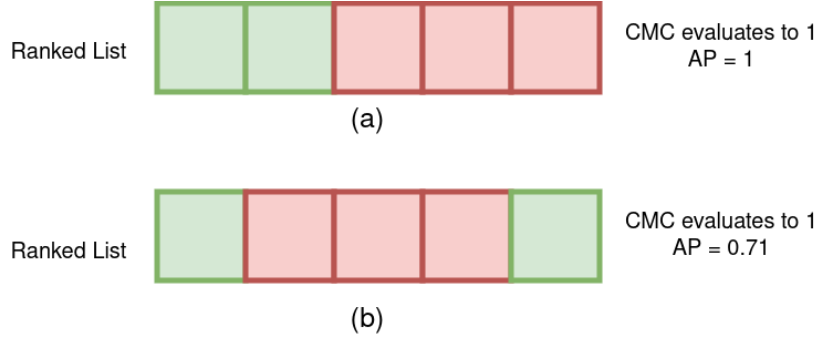


Figure 3.15: Toy example illustrating the limited expressivity of CMC. (a) and (b) depict ranked lists, each containing two true positives (denoted by green boxes). In (a), both true positives are ranked at the top, resulting in a CMC curve equal to 1 and an Average Precision (AP) of 1. In (b), the second true positive appears at the fifth position, lowering the precision and yielding an AP of 0.71. Despite this, the CMC curve remains 1.

age of a person is among the top- k of ranked predictions. While CMC is an effective measure to show the accuracy of top-ranked matches, it only provides limited information about the system’s full performance. For instance, the CMC metric does not consider recall. It is thus biased if multiple ground truths exist, i.e. in multi-gallery settings in which each identity in a gallery has multiple instances. Hence, CMC only focuses on top-ranked matches and lacks distinctiveness. Figure 3.15 shows a toy example that illustrates the limited expressivity of CMC in comparison to average precision (AP). The toy example contains two ranked lists, each with two true positives (highlighted in green). In both ranked lists in Figure 3.15 (a) and Figure 3.15 (b) the CMC metric evaluates to 1 indicating that all true positives have been found. However, the AP in Figure 3.15 (b) decreases to 0.71 due to the presence of false positives (highlighted in red) between the first and second true positive, which reduce the overall precision.

mean Average Precision (mAP) Both person re-identification and person search utilize the mAP metric as a performance measure. In person re-identification, the AP is obtained for every query image by calculating the area under the Precision-Recall curve. Person search, on the other hand, borrows the mAP metric from the object detection literature and measures the AP in terms of localization accuracy for all bounding boxes of a query identity in the gallery set. The AP for all queries is averaged to obtain the mAP for re-identification and person search, respectively.

3.6.3 Multi-Target Tracking

In the following section, we summarize the most important metrics for multi-target tracking tasks such as Multi-Object Tracking (MOT) and Multi-Person Pose Tracking (MPPT). In Chapter 5 we further propose an evaluation metric for Multi-Person Re-ID Pose Tracking.

CLEAR MOT The CLEAR MOT evaluation metric proposed by Bernardin and Stiefelhagen (2008) is one of the most used evaluation metrics for multi-object and multi-person pose tracking. CLEAR MOT incorporates the two sub-metrics Multi-Object Tracking Accuracy (MOTA) and

Multi-Object Tracking Precision (MOTP). The MOTA measures the accuracy of a tracker by greedily matching predictions to ground truth objects in a frame-wise manner. The matching is bijective and requires a similarity metrics S and a similarity threshold τ_s . During matching, detections and ground truth are only associated if $S \geq \tau_s$. For bounding box-based tracking such as MOT, S is typically the bounding box Intersection over Union (IoU). In keypoint-based tracking such as multi-person pose tracking S is typically PCKh or KS. In practice, multiple matches are possible for which $S \geq \tau_s$ holds true. For that reason, the matching is performed to maximize the MOTA score. A ground truth object in a frame f that is not associated with any detection is counted as FN. Vice versa, a prediction is counted as FP if it can not be associated with any ground truth. Moreover, MOTA incorporates association errors in terms of Identity Switches (IDSWs). In particular, IDSWs occur if a ground truth track is matched with a different track id. This scenario typically occurs if the tracker swaps the identity of two tracks or if a track was lost and re-initialized with a new track ID. Overall, the MOTA metric is defined as

$$MOTA = 1 - \frac{\sum_f |FN_f| + |FP_f| + |IDSW_f|}{\sum_f |GT_f|}, \quad (3.31)$$

where FN_f , FP_f , and $IDSW_f$ denote the set of false negatives, false positives, and identity switches in frame f , and GT_f is the set of ground truth objects in frame f .

MOTA does not incorporate localization error, and therefore, localization errors are measured by a second metric, MOTP. MOTP measures the average similarity score for all TPs

$$MOTP = \frac{\sum_f \sum_m |TP_f| S(\hat{x}_m, x_m)}{\sum_f |TP_f|}, \quad (3.32)$$

where $S(\hat{x}_m, x_m)$ denotes the similarity score between ground truth $\hat{x}_m$ and detection $\hat{x}_m$ and $|TP_f|$ is the total number of ground truth matches in frame f .

Trajectory-based Metrics Trajectory-based metrics as proposed in [Wu and Nevatia \(2006\)](#) aim to evaluate the performance on the entire trajectories and not in a frame-wise fashion as done by CLEAR metrics. More precisely, trajectory-based metrics calculate a bijective matching between predicted and ground truth tracks. To quantify a tracker's performance, every trajectory can be categorized as mostly tracked (MT), partially tracked (PT), and mostly lost (LT). A trajectory is classified as mostly tracked if it aligns with the ground truth trajectory at least 80% of the time. Vice versa, a track is classified as mostly lost if less than 20% can be associated with the ground truth trajectory. All other trajectories fall into the category of partially tracked. Moreover, fragmentations (FM) quantify how often a track transitions between different states, i.e., from "tracked" to "lost".

IDF1 is another trajectory-based metric proposed by [Ristani et al. \(2016b\)](#) that focuses on association accuracy over detection accuracy (in contrast to MOTA). This metric defines additional measures IDTPs (identity true positives) as the set of detections that align with the ground truth trajectories. Similarly, IDFPs (identity false positives) and IDFNs (identity false negatives) are the

remaining detections and ground truths. Overall, IDF1 is defined as

$$IDF1 = \frac{2|IDTP|}{2|IDTP| + |IDFN| + |IDFP|}. \quad (3.33)$$

Furthermore, [Ristani et al. \(2016b\)](#) introduce Identity Precision (IDP) and Identity Recall (IDR) which are defined as

$$IDP = \frac{|IDTP|}{|IDTP| + |IDFP|} \quad (3.34)$$

$$IDR = \frac{|IDTP|}{|IDTP| + |IDFN|}. \quad (3.35)$$

Higher Order Tracking Accuracy (HOTA) Previous metrics, such as MOTA and IDF1, have drawbacks and do not properly reflect a tracker's performance in various aspects. MOTA, for instance, is dominated by the detection performance and outweighs the association performance. Moreover, MOTA penalizes trackers that correct their own association mistakes, as every change in identity negatively affects the overall MOTA score. Consequently, MOTA can not account for global alignments between predictions and tracks. Furthermore, MOTA and IDF1 do not incorporate localization accuracy (as measured by MOTP).

On the other hand, the IDF1 metric only considers the best global mapping of a ground truth trajectory. This can lead to a decrease in performance with improving detection performance: Non-matched detections do not positively affect the IDF1 score. Moreover, IDF1 encourages trackers to estimate the total number of unique objects because detected tracks not matched to the globally best matching track are counted as negatives ([Luiten et al., 2020](#)).

The HOTA metric proposed by [Luiten et al. \(2020\)](#) builds upon MOTA and is designed to tackle the drawbacks of CLEAR and Trajectory-based metrics. HOTA aims to explicitly balance the effect of accurate localization, accurate detection, and association and is comprised of several sub-metrics. Similar to MOTA, the association between ground truth and predictions is performed in a frame-wise manner. To measure the association accuracy for any true positive c , HOTA introduces the concepts of TPA (true positive associations), FNA (false-negative associations), and FPA (false-positive associations) that are defined for all sets of TPs (true-positives). Consequently, $TPA(c) = \{k\}$ is the set of all TPs with the same track id assigned to the same ground truth trajectory. Moreover, $FNA(c)$ is the set of all FNs o belonging to the same ground truth trajectory and all TPs k associated with the same ground truth but belonging to different track IDs. Vice versa, $FPA(c)$ is the union of two sets: all FPs that share the same track ID with c and the set of all TPs that share the same track ID with c but are associated with different ground truth tracks. Overall, the association accuracy (AssA) is defined as the Jaccard index at various localization thresholds $\alpha \in [0.05, 0.95]$ following

$$AssA_\alpha = \frac{1}{|TP|} \sum_{c \in \{TP\}} A(c) \quad (3.36)$$

$$A(c) = \frac{|TPA(c)|}{|TPA(c)| + |FPA(c)| + |FNA(c)|}, \quad (3.37)$$

where $A(c)$ is the accuracy for any TP c . The detection accuracy (DetA) is defined in a similar

manner for various values of $\alpha \in [0.05, 0.95]$ following

$$DetA_\alpha = \frac{|TP|}{|TP| + |FN| + |FP|}. \quad (3.38)$$

Overall, HOTA is designed to measure the alignment of matched detections of a track averaged over all detections, while penalizing false or missing detections. HOTA is defined similarly to the geometric mean between detection accuracy and association accuracy to ensure an equal impact of both sub-metrics on the overall score, following

$$HOTA_\alpha = \sqrt{DetA_\alpha \cdot AssA_\alpha} \quad (3.39)$$

$$HOTA = \int_0^1 HOTA_\alpha d\alpha, \quad (3.40)$$

where the final HOTA score is measured as the area under the curve of all HOTA scores across various localization thresholds α .

For further insight, [Luiten et al. \(2020\)](#) propose additional sub-metrics or variations of HOTA that provide task-specific insights. For instance, the Localization Accuracy (LocA) is defined as

$$LocA = \int_0^1 \frac{1}{|TP_\alpha|} \sum_{c \in \{TP_\alpha\}} S(c) d\alpha, \quad (3.41)$$

where $S(c)$ denotes the similarity between the associated prediction and ground truth of TP c . On the other hand, the standard definition of HOTA is designed to evaluate long-term global tracking; thus, fragmentations are not considered. This poses a drawback for certain applications. To circumvent these drawbacks, [Luiten et al. \(2020\)](#) propose a fragmentation-aware HOTA (FA-HOTA) metric by incorporating an additional sub-metric Fragmentation Accuracy (FragA) that measures the average fragmentation over all TPs. The fragmentation accuracy is defined similarly to the AssA metric as

$$FragA = \frac{1}{|TP|} \sum_{c \in \{TP\}} \mathcal{F}(c), \mathcal{F}(c) = \frac{|FRA(c)|}{|TPA(c)| + |FNA(c)| + |FPA(c)|}, \quad (3.42)$$

where $FRA(c) \subseteq TPA(c)$ for all c that belong to the same segment. The Fragmentation-aware HOTA (FA-HOTA) score is then defined as

$$FA-HOTA_\alpha = \sqrt{DetA_\alpha \cdot \sqrt{AssA_\alpha \cdot FragA_\alpha}} \quad (3.43)$$

In Chapter 5, we propose an additional metric *Keypoint HOTA* which extends HOTA for multi-person re-ID pose tracking.

Self-supervised Keypoint Correspondences for Multi-Person Pose Estimation and Tracking in Videos

In this chapter, we present a method for multi-person pose estimation and tracking in videos. The effectiveness of deep learning methods in associating keypoints across frames is hindered by sparse annotations, and the lack of diversity in existing multi-person pose tracking datasets. To address these challenges, we introduce a self-supervised keypoint correspondence network trained on image data, leveraging simple yet powerful data augmentation techniques rooted in affine transformations. Our proposed network not only allows for robustly associating human keypoints across frames for multi-person pose tracking but also allows to recover missed pose detections effectively.

Individual Contribution: The following chapter is based on the publication [Rafi et al. \(2020\)](#):

Self-supervised Keypoint Correspondences for Multi-Person Pose Estimation and Tracking in Videos

Umer Rafi¹, [Andreas Doering](#)¹, Bastian Leibe, and Juergen Gall
IEEE European Conference on Computer Vision (ECCV), 2020

This publication was done in close collaboration between Umer Rafi and Andreas Doering. The initial idea to exploit keypoint correspondences for the tracking of poses was proposed by Umer Rafi. The method was then further implemented and refined in joint discussions between Umer Rafi and Andreas Doering. Andreas Doering implemented and carried out the empirical evaluation and the tracking pipeline. Bastian Leibe assisted with proofreading and provided valuable feedback. Juergen Gall provided scientific guidance.

Contents

4.1	Introduction	68
4.2	Method Overview	70
4.2.1	Pose Estimation Model	70
4.3	Keypoint Correspondence Network	71
4.3.1	Siamese Matching Module	72
4.3.2	Refinement Module	73
4.3.3	Training	73

¹equal contribution

4.4	Multi-Person Pose Tracking	73
4.4.1	Recover Missed Detections	74
4.4.2	Tracking	74
4.4.3	Track Merging	75
4.5	Experiments and Results	76
4.5.1	Implementation Details	76
4.5.2	Baselines	77
4.5.3	Effect of Joint Detection Threshold and Pose Recovery	78
4.5.4	Effect of Refinement Module and Duplicate Removal	79
4.5.5	Impact of τ_{corr}	79
4.5.6	Comparison with State-of-the-art Methods	79
4.5.7	Qualitative Results	80
4.5.8	Failure Cases	80
4.6	Conclusion	81

4.1 Introduction

Human pose estimation is a very active research field in computer vision that is relevant for many applications like computer games, security, sports, and autonomous driving. Over the years, the human pose estimation models have been greatly improved (Felzenszwalb and Huttenlocher, 2005; Nie et al., 2018; Cao et al., 2017; Kocabas et al., 2018; Xiao et al., 2018; Bertasius et al., 2019; Li et al., 2019b) due to the availability of large scale image datasets for human pose estimation (Lin et al., 2014; Andriluka et al., 2014; Wu et al., 2017). More recently, researchers started to tackle the more challenging problem of multi-person pose tracking (Iqbal et al., 2017; Insafutdinov et al., 2017; Xiao et al., 2018; Sun et al., 2019a; Zhang et al., 2019).

In multi-person pose tracking, the goal is to estimate human poses in all frames of a video and associate them over time. However, video annotations are costly and time-consuming. Consequently, recently proposed video datasets (Andriluka et al., 2018) are less diverse and sparsely annotated compared to large-scale image datasets for human pose estimation (Lin et al., 2014; Wu et al., 2017). This makes it challenging to learn deep networks for associating human keypoints across frames that are robust to nuisance factors such as motion blur, fast motions, and occlusions as they occur in videos.

State-of-the-art approaches (Xiao et al., 2018; Sun et al., 2019a; Guo et al., 2018) rely on optical flow or person re-identification (Zhang et al., 2019) in order to track the persons. Both approaches, however, have disadvantages. Optical flow fails if a person becomes occluded, resulting in a lost track. While person re-identification allows to associate persons even if they disappeared for a long time, it remains difficult to associate partially occluded persons with person re-identification models that operate on bounding boxes of the full person. Moreover, the limited annotations in pose tracking datasets require to train the models on additional datasets for person re-identification. We propose, therefore, to learn a network that infers keypoint correspondences for multiple persons. The correspondence network comprises a Siamese matching module that takes a frame with estimated human

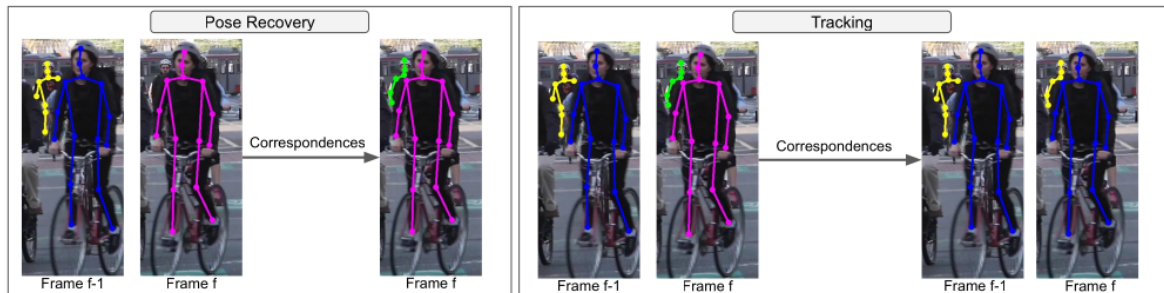


Figure 4.1: Our contributions: (Left) We use keypoint correspondences to recover missed pose detections by using the temporal context of the previous frame. (Right) We use keypoint correspondences to associate detected and recovered pose detections for the task of multi-person pose tracking.

poses as input and estimates the corresponding poses for a second frame. Such an approach has the advantage that it is not limited to a fixed temporal frame distance, and it allows to track persons when they are partially occluded. Our goal is to utilize keypoint correspondences to recover missed poses of a top-down human pose estimator, e.g., due to partial occlusion, and to utilize keypoint correspondences for multi-person tracking. The challenge, however, is to train such a network due to the sparsely annotated video datasets. In fact, in this chapter, we consider the extreme case where the network is not trained on any video data or a dataset where identities of persons are annotated. Instead, we show that such a network can be trained on an image dataset for multi-person pose estimation (Lin et al., 2014). Besides the human pose annotations, which are anyway needed to train the human pose estimator, the approach does not require any additional supervision. To improve the keypoint associations, we propose an additional refinement module that refines the affinity maps of the Siamese matching module.

To summarize, the contributions are:

- We propose an approach for multi-frame pose estimation and multi-person pose tracking that relies on self-supervised keypoint correspondences, which are learned from a large-scale image dataset with human pose annotations.
- Combined with a top-down pose estimation framework, we use keypoint correspondences in two ways as illustrated in Figure 4.1: We use keypoint correspondences to (i) recover pose detections that have been missed by the top-down pose estimation framework and to (ii) associate detected and recovered poses in different frames of a video.
- We evaluate the approach on the PoseTrack 2017 and 2018 datasets for the tasks of multi-frame pose estimation and multi-person pose tracking. Our approach achieves state-of-the-art results without using additional training data except for (Lin et al., 2014) for the proposed correspondence network.



Figure 4.2: Given a sequence of frames, we detect a set of person bounding boxes and perform top-down pose estimation. Our proposed method uses keypoint correspondences to (i) recover missed detections and (ii) to associate detected and recovered poses to perform tracking. The entire framework does not require any video data for training since the network for estimating keypoint correspondences is trained on single images using self-supervision.

4.2 Method Overview

In this work, we propose a multi-person pose tracking framework that is robust to motion blur and severe occlusions, even though it does not need any video data for training. As it is illustrated in Figure 4.2, we first estimate for each frame the human poses and then track them. For multi-person human pose estimation, we utilize an off-the-shelf object detector (Cai and Vasconcelos, 2017) to obtain a set of bounding boxes for the persons in each frame. For each bounding box, we then perform multi-person pose estimation in a top-down fashion by training an adapted GoogleNet (Szegedy et al., 2015), which we will discuss in Section 4.2.1.

To be robust to motion blur and severe occlusions, we do not use optical flow in contrast to previous works like (Xiao et al., 2018). Instead, we propose a network that estimates for a given frame with estimated keypoints the locations of the keypoints in another frame. We use this network for recovering human poses that have been missed by the top-down pose estimation framework as described in Section 4.4.1 and for associating detected and recovered poses across the video as described in Section 4.4.2.

The main challenge for the keypoint correspondence network is handling occluded keypoints and the limited amount of densely annotated video data. In order to address these issues, we do not train the network on video data but on single images using self-supervision. In this way, we can simulate disappearing keypoints by truncation and leverage large-scale image datasets like MS-COCO (Lin et al., 2014) for tracking. We will first describe the keypoint correspondence network in Section 4.3 and then discuss the tracking framework in Section 4.4.

4.2.1 Pose Estimation Model

Figure 4.3 shows our proposed two-stage pose estimation framework. Both stages utilize a batch-normalized GoogleNet (Szegedy et al., 2015) as the backbone. In the first stage, layers 1 to 17 of GoogleNet are employed, while the second stage uses layers 3 to 17. In the first stage, features

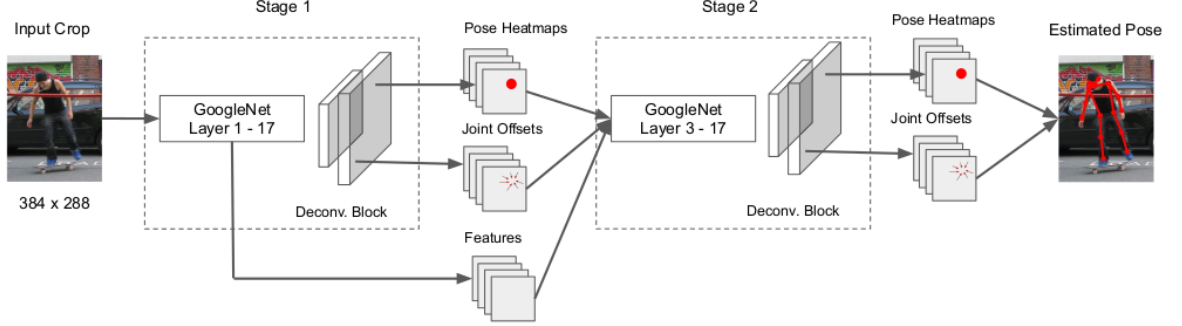


Figure 4.3: Our two-stage pose estimation framework. Each stage uses GoogleNet (Szegedy et al., 2015) as the backbone. The features extracted by the backbone in the first stage are fed into a deconvolution layer block to produce pose and joint offset maps. The backbone features, pose heatmaps, and joint offset maps from the first stage are fed into the second stage to produce refined pose and joint offset maps.

extracted by the backbone are passed through a deconvolution layer block to generate pose and joint offset maps, similar to Zhang et al. (2019). These backbone features, along with the pose heatmaps and joint offset maps from the first stage, are then input to the second stage, where refined pose and joint offset maps are produced. The final pose detections are obtained by combining the pose heatmaps with the joint offsets from the second stage.

Due to the pooling used in the backbone, the resolution of the pose heatmaps is reduced by a factor of 4 in height and width dimensions. Consequently, the up-sampled predicted pose is slightly away from the actual pose. Towards this end, we append a joint offset head to predict the deltas, i.e., Δx and Δy for each keypoint. The position of the j th keypoint $(\hat{x}_j, \hat{y}_j)$ at inference is computed as

$$(\hat{x}_j, \hat{y}_j) = (x_j + \Delta x_j, y_j + \Delta y_j). \quad (4.1)$$

where (x_j, y_j) is the up-sampled position from the pose heatmaps. During training, we minimize the L1 loss between the predicted and ground-truth deltas for the joint offset maps and use the binary cross entropy loss for the pose heatmaps.

4.3 Keypoint Correspondence Network

Given two images $\mathbf{I}_1$ and $\mathbf{I}_2$ with person keypoints $\mathcal{P} = \{P^p\}_{1:N_p}$ for all persons p in image $\mathbf{I}_1$, our goal is to find the corresponding keypoints in $\mathbf{I}_2$. Towards this end, we use a Siamese network as shown in Figure 4.4, which estimates for each keypoint an affinity map. The refinement module, described in Section 4.3.2, further improves the affinity maps.

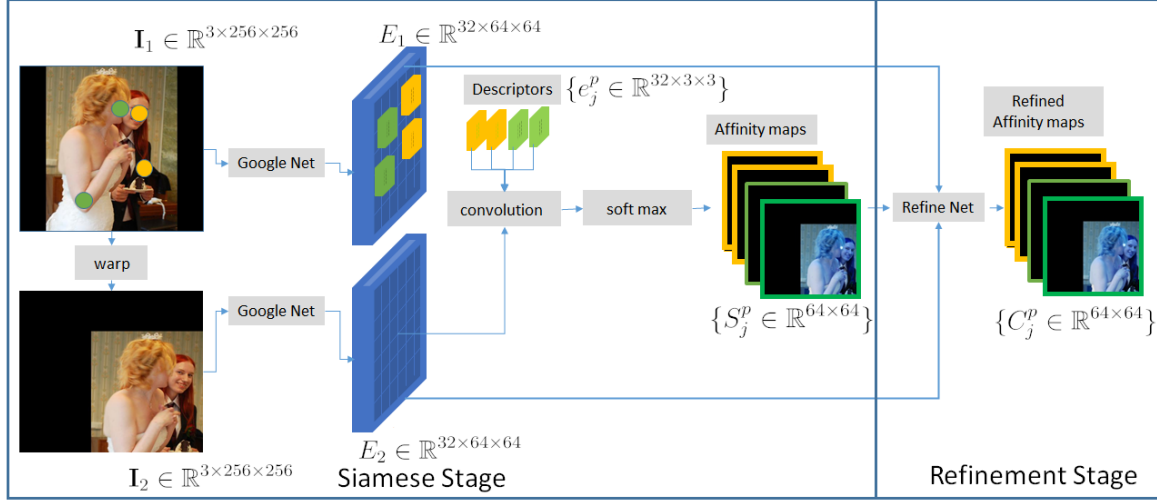


Figure 4.4: Keypoint correspondence network. The Siamese network takes images I_1 and I_2 and keypoints $\mathcal{P} = \{P^p\}_{1:N_p}$ for all persons p in image I_1 as input and generates the feature maps E_1 and E_2 , respectively. The keypoints of the different persons are shown in green and yellow, respectively. For each keypoint, a descriptor e_j^p is extracted from E_1 and convolved with the feature map E_2 to generate an affinity map A_j^p . To improve the affinity maps for each person, the refinement network takes E_1 , E_2 and the affinity maps A_j^p for person p as input and generates refined affinity maps C_j^p .

4.3.1 Siamese Matching Module

The keypoint correspondence network consists of a Siamese network. Each branch in the Siamese network is a batch normalized GoogleNet up to layer 17 with shared parameters (Szegedy et al., 2015). The Siamese network takes an image pair (I_1, I_2) and keypoints $\{P^p\}_{1:N_p}$ for persons $p \in \{1, \dots, P\}$ in the image I_1 as input. During training, I_2 is generated by applying a randomly sampled affine warp to I_1 . In this way, we do not need any annotated correspondences during training or pairs of images but train the network on single images with annotated poses. We use an image resolution of 256×256 for both images.

The Siamese network generates features $E_1 \in \mathbb{R}^{32 \times 64 \times 64}$ and $E_2 \in \mathbb{R}^{32 \times 64 \times 64}$ for images I_1 and I_2 , respectively. The features are then pixel-wise l_2 -normalized and local descriptors $e_j^p \in \mathbb{R}^{32 \times 3 \times 3}$ are generated for each keypoint $\mathbf{x}_j^p$ by extracting squared patches around the spatial position of a keypoint in the feature maps E_1 .

Given a local descriptor e_j^p , we compute its affinity map A_j^p over all pixels $x = \{1, \dots, 64\}$ and $y = \{1, \dots, 64\}$ in E_2 as:

$$A_j^p = e_j^p \otimes E_2 \quad (4.2)$$

where $\otimes$ denotes the convolution operation. Finally, a softmax operation is applied to the affinity map A_j^p , i.e.,

$$S_j^p(x, y) = \frac{\exp(A_j^p(x, y))}{\sum_{x', y'} \exp(A_j^p(x', y'))}. \quad (4.3)$$

We refine the affinity maps S_j^p further using a refinement module.

4.3.2 Refinement Module

Similar to related multi-stage approaches (Cao et al., 2017; Dantone et al., 2014; Kocabas et al., 2018), we append a second module to the keypoint correspondence network to improve the affinity maps generated by the Siamese matching module. We use a batch-normalized GoogleNet for the refinement module from layer 3 to layer 17. The refinement module concatenates E_1 , E_2 , and the affinity maps $\{A_j^p\}_{1:N_p}$ for a single person p and refines the affinity maps, which we denote by $C_j^p \in \mathbb{R}^{64 \times 64}$. The refinement module is therefore applied to the affinity maps for all persons $p \in \{1, \dots, P\}$. Before we describe in Section 4.4 how we will use the affinity maps for tracking C_j^p , we describe how the keypoint correspondence network is trained.

4.3.3 Training

Since we train our network using self-supervision, we train it using a single image $\mathbf{I}_1$ with annotated poses. We generate a second image $\mathbf{I}_2$ by applying a randomly sampled affine warp to $\mathbf{I}_1$. We then generate the ground-truth affinity map $\hat{S}_j^p$ for a keypoint $\hat{\mathbf{x}}$ belonging to person p as:

$$\hat{S}_j^p(x, y) = \begin{cases} 1 & \text{if } x = \hat{x}_j^p \text{ and } y = \hat{y}_j^p, \\ 0 & \text{otherwise,} \end{cases} \quad (4.4)$$

where $(\hat{x}_j^p, \hat{y}_j^p)$ is the spatial position of the ground-truth correspondence for keypoint $\mathbf{x}_j^p$ in image $\mathbf{I}_2$, which we know from the affine transformation. As illustrated in Figure 4.4, not all corresponding keypoints are present in image $\mathbf{I}_2$. In this case, the ground-truth affinity map is zero, and predicting a corresponding keypoint is penalized.

During training, we minimize the binary cross entropy loss between the predicted affinity maps S_j^p and C_j^p and the ground-truth affinity map $\hat{S}_j^p$:

$$\min_{\theta} \sum_{x,y} - \left(\hat{S}_j^p \log(S_j^p) + (1 - \hat{S}_j^p)(1 - \log(S_j^p)) \right), \quad (4.5)$$

$$\min_{\theta} \sum_{x,y} - \left(\hat{S}_j^p \log(C_j^p) + (1 - \hat{S}_j^p)(1 - \log(C_j^p)) \right), \quad (4.6)$$

where θ are the parameters of the keypoint correspondence framework.

4.4 Multi-Person Pose Tracking

We use the keypoint correspondence network in two ways. First, we use it to recover human poses that have been missed by the frame-wise top-down multi-person pose estimation step, which will be described in Section 4.4.1. Second, we use keypoint correspondences for tracking poses across frames of the video as described in Section 4.4.2.

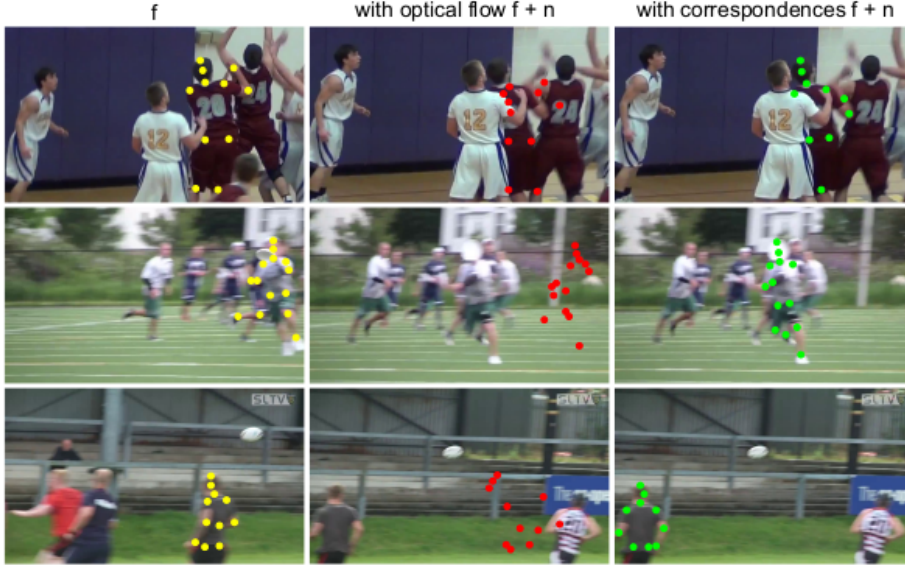


Figure 4.5: Recovering missed detections. (a) Person detected by the top-down pose estimation framework in frame $f-1$. (b) The person was missed by the top-down pose estimation framework in frame f due to occlusion. (c) Keypoint affinity maps of the missed person from frame $f-1$ to frame f . (d) Corresponding keypoints in frame f . (e) Estimated bounding box from the corresponding keypoints and the recovered pose.

4.4.1 Recover Missed Detections

For a given frame f , we first detect the human poses in the frame using the top-down multi-person pose estimator described in Section 4.2.1. While the person detector (Cai and Vasconcelos, 2017) performs well, it fails in situations with overlapping persons and motion blur. Consequently, the human pose is not estimated in these cases. Examples are shown in Figure 4.5(b).

Given the detected human poses $P_{f-1}^p = \{\mathbf{x}_{j,f-1}^p\}$ for persons $p \in \{1, \dots, P\}$ in frame $f-1$, we compute the corresponding refined affinity maps $C_j^p = \{C_j^p\}$ by using the keypoint correspondence network. For each keypoint $\mathbf{x}_{j,f-1}^p$, we then get the corresponding keypoint $\bar{\mathbf{x}}_{j,f}^p$ in frame f by taking the argmax of C_j^p and mapping it to the image resolution. Since the resolution of the affinity maps is lower than the image resolution and since the frame f might contain a keypoint that was occluded in the previous frame, we reestimate the propagated poses. This is done by computing for each person p a bounding box that encloses all keypoints $\bar{P}_f^p = \{\bar{\mathbf{x}}_{j,f}^p\}$ and using our proposed human pose estimation network to get a new pose for this bounding box. We denote the newly estimated poses by $\hat{P}_f^p$. The overall procedure is shown in Figure 4.5. We apply OKS-based non-maximum suppression (Xiao et al., 2018) to discard redundant poses.

4.4.2 Tracking

Given detected and recovered poses, we need to link them across video frames to obtain tracks of human poses. Tracking can be seen as a data association problem over estimated poses. Previously,

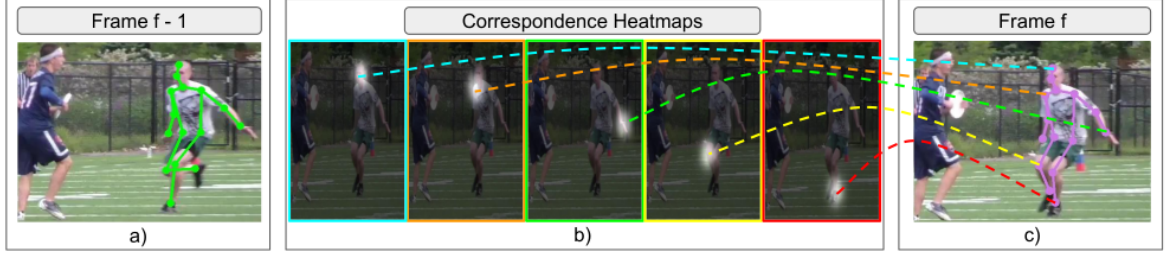


Figure 4.6: Pose to track association. (a) A tracked human pose till frame $f-1$. (b) Keypoint affinity maps of the track to frame f . (c) A pose instance of frame f . The dashed lines indicate the position of each detected joint of the pose instance in the correspondence affinity maps of the tracked pose in frame $f-1$.

the problem has been approached using bipartite graph matching (Girdhar et al., 2018) or greedy approaches (Xiao et al., 2018; Sun et al., 2019a; Doering et al., 2018). In this work, we greedily associate estimated poses over time by using the keypoint correspondences. We initialize tracks on the first frame and then associate new candidate poses to initial tracks one frame at a time.

Formally, our goal is to assign pose instances $\{U_f^p\} = \{P_f^p\} \cup \{\hat{P}_f^p\}$ in frame f for persons $p \in \{1, \dots, P\}$ to tracks $\{T_{f-1}^q\}$ till frame $f-1$ for persons $q \in \{1, \dots, Q\}$. Towards this end, we measure the similarity between a pose instance U_f^p and a track T_{f-1}^q as:

$$\text{Sim}(T_{f-1}^q, U_f^p) = \frac{\sum_{j=1}^{N_q} C_j^q(\mathbf{x}_{j,f}^p) \cdot \mathbb{I}_{C_j^q(\mathbf{x}_{j,f}^p) > \tau_{\text{corr}}}}{\sum_{j=1}^{N_q} \mathbb{I}_{C_j^q(\mathbf{x}_{j,f}^p) > \tau_{\text{corr}}}}, \quad (4.7)$$

where C_j^q is the affinity map of the keypoint j in track T_{f-1}^q for frame f . The affinity map is computed by the network described in Section 4.3. $C_j^q(\mathbf{x}_{j,f}^p)$ is the confidence value in the affinity map C_j^q at the location of the joint $\mathbf{x}_{j,f}^p$ for person p in frame f . N_q is the number of detected joints. An example is shown in Figure 4.6. We only consider $\mathbf{x}_{j,f}^p$ if its affinity is above τ_{corr} . If a pose U_f^p cannot be matched to a track T_{f-1}^q , a new track is initiated.

4.4.3 Track Merging

We propose a post-processing step in which we merge tracks of the same pose instance at different time steps by utilizing keypoint correspondences from multiple frames. Given two tracks T^q and T^p as illustrated in Figure 4.7, we select three pose instances $\{U_f^q\}$ with $f \in \{f_s^q, f_c^q, f_e^q\}$ at the start, center and end frames of track T^q . For each of the pose instances U_f^q , we compute the pose $\bar{U}_f^q$ for the starting frame f_s^p of track f^p using correspondences, as described in Section 4.4.1. We then employ OKS as similarity metric and calculate the average similarity between tracks T^q and T^p as

$$\text{Sim}_{\text{match}}(T^q, T^p) = \frac{\sum_{f \in \{f_s, f_c, f_e\}} \text{OKS}(\bar{U}_f^q, U_{f_s^p}^p)}{3}. \quad (4.8)$$

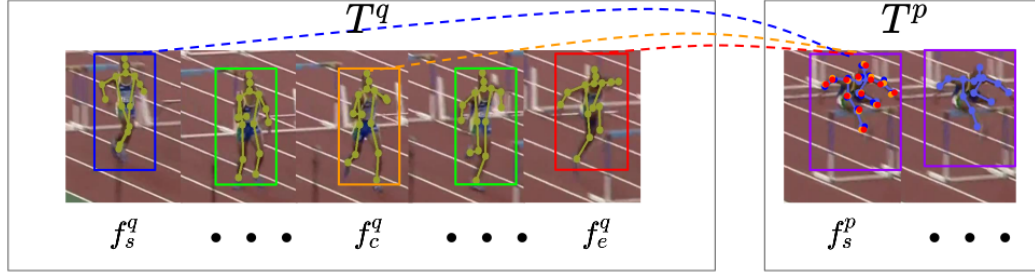


Figure 4.7: Track merging: For the start frame f_s^q , the center frame f_c^q and the last frame f_e^q of track T^q , we estimate poses from keypoint correspondences in the start frame f_s^p of T^p , as illustrated by the colored dashed lines. We use an OKS-based similarity metric to measure the average pose similarity between the poses from correspondences and the pose in the starting frame f_s^p of track T^p .

4.5 Experiments and Results

We evaluate our approach on the Posetrack 2017 and 2018 datasets (Andriluka et al., 2018). The datasets have 292 and 593 videos for training and 214 and 375 videos for evaluation, respectively. We evaluate multi-frame pose estimation and tracking results using the mAP and MOTA evaluation metrics.

4.5.1 Implementation Details

We provide additional implementation details for our top-down pose estimation and keypoint correspondence network below.

Top-down Pose Estimation: In our proposed top-down framework for frame-level pose estimation, we use cascade R-CNN (Cai and Vasconcelos, 2017) for person detection. Detected individuals are cropped to a size of 384×288 , and these crops are fed into our pose estimation framework. The number of parameters (39.5 M) of our model is significantly lower compared to related works such as FlowTrack (Xiao et al., 2018) (63.6 M) or EOGN (Zhang et al., 2019) (60.3 M).

We train the pose estimation framework on the MS-COCO dataset (Lin et al., 2014) for 260 epochs with a base learning rate of $1e^{-3}$. The learning rate is reduced to $1e^{-4}$ after 200 epochs. During training, we apply random flippings and rotations to input crops. We finetune the pose estimation framework on the PoseTrack 2017 dataset (Andriluka et al., 2018) for 12 epochs. The learning rate is further reduced to $1e^{-5}$ after epoch 7.

Keypoint Correspondence Network: We perform module-wise training. We first train the Siamese module. We then fix the Siamese module and train the refinement module. Both modules are trained for 100 epochs with base learning rate of $1e^{-4}$ reduced to $1e^{-5}$ after 50 epochs. We generate a second image for each training image by applying random translations, rotations, and flippings to the first image. The keypoint correspondence network is trained only on the MS-COCO dataset (Lin et al., 2014). We did not observe any improvements in our tracking results by finetuning

Tracking Method	GT Boxes	IDSW	MOTA
OKS	✓	6582	65.9
Optical Flow	✓	4419	68.4
Re-ID	✓	4164	67.1
Correspondences	✓	3583	70.5
OKS	✗	7207	60.4
Optical Flow	✗	5611	66.7
Re-ID	✗	4589	64.1
Correspondences	✗	3632	67.9

Table 4.1: Comparison with tracking baselines on the PoseTrack 2017 validation set. For the comparison, the detected poses based on ground-truth bounding boxes (GT Boxes) or detected bounding boxes are the same for each approach. Correspondence-based tracking consistently improves MOTA compared to the baselines and significantly reduces the number of identity switches (IDSW).

the correspondence model on the PoseTrack dataset. Training only on the PoseTrack dataset yielded sub-optimal tracking results since PoseTrack is sparsely annotated and contains far fewer person instances than MS-COCO.

4.5.2 Baselines

We compare our keypoint correspondence tracking to different standard tracking baselines for multi-person pose tracking as reported in Table 4.1. To measure the performance of each baseline, we report the number of identity switches and the MOTA score. For a fair comparison, we replace the keypoint correspondences in our framework with different baselines. For all experiments, we use the same detected poses, using either ground truth or detected bounding boxes.

OKS: OKS without taking the motion of the poses into account has been proposed in (Xiao et al., 2018). OKS measures the similarity between two poses and is independent of their appearance. It is not robust to large motion, occlusion, and large temporal offsets. This is reflected in Table 4.1 as this baseline achieves the lowest performance.

Optical Flow: Optical flow is a temporal baseline that has been proposed in (Xiao et al., 2018). We use optical flow to warp the poses from the previous frame to the current frame. We then apply OKS to associate the warped poses with candidate poses in the current frame. We use the pre-trained PWC-net (Sun et al., 2018a) for a fair comparison. Optical flow clearly outperforms OKS and achieves superior MOTA of 68.4 and 66.7 for GT and detected bounding boxes, respectively.

Person Re-id: Compared to optical flow and OKS, person re-identification is more robust to larger temporal offsets and large motion. However, the results indicate that person re-identification operating on the bounding boxes performs sub-optimally under the frequent partial occlusions in the PoseTrack datasets. For our experiments, we use the pre-trained re-identification model from (Luo et al., 2019b). Re-identification-based tracking achieves MOTA scores of 67.1 and 64.1 for GT and detected bounding boxes, respectively.

Joint Threshold	mAP	MOTA	Joint Threshold	mAP	MOTA
Detected Poses Only			Detected and Recovered Poses.		
0.0	80.1	48.1	0	82.0	48.1
0.1	79.7	63.3	0.1	81.4	64.1
0.2	78.9	66.1	0.2	80.5	67.2
0.3	77.7	67.6	0.3	79.2	68.8
0.4	75.9	68.0	0.4	77.2	69.2
0.5	73.1	67.1	0.5	74.2	68.2

Table 4.2: Effect of joint detection threshold and pose recovery on mAP and MOTA for the PoseTrack 2018 validation set. The results are shown for (left) detected poses only and (right) detected and recovered poses. As expected, recovering missed detections improves both MOTA and mAP. A good trade-off between mAP and MOTA is achieved by the joint detection threshold 0.3.

Design Choices	Refinement Module	Duplicate Removal	MOTA	mAP	IDSW
Correspondence Tracking	✓	✓	67.9	78.0	3632
Correspondence Tracking	✗	✓	66.9	77.7	4304
Correspondence Tracking	✓	✗	64.5	77.9	8288

Table 4.3: Comparison of mAP and MOTA for different design choices on the PoseTrack 2017 validation set.

The results show that correspondence-based tracking (1) consistently improves over the baselines for ground-truth and detected bounding boxes with MOTA scores of 70.5 and 67.9, respectively, and (2) significantly reduces the number of identity switches. Compared to optical flow, correspondences are more robust to partial occlusions, motion blur, and large motions. A qualitative comparison is provided in Section 4.5.7.

4.5.3 Effect of Joint Detection Threshold and Pose Recovery

We evaluate the impact of different joint detection thresholds on mAP and MOTA for the PoseTrack 2018 dataset as shown in Table 4.2. Since mAP does not penalize false-positive keypoints, thresholding decreases the pose estimation performance by discarding low confident joints. Vice versa, joint thresholding results in cleaner tracks and improves the tracking performance, as MOTA penalizes false-positive keypoint detections. A good trade-off between mAP and MOTA is achieved for the joint detection threshold 0.3 resulting in mAP and MOTA of 77.7 and 67.6, respectively.

While on the left hand side of Table 4.2 we report the results without recovering missed detections as described in Section 4.4.1, the table on the right hand side shows the impact on mAP and MOTA if missed detections are recovered. The recovering of missed detections improves the accuracy for all thresholds. For the joint detection threshold 0.3, mAP and MOTA are further improved to 79.2 and 68.8, respectively.

τ_{corr}	MOTA	mAP
0.1	67.9	77.9
0.2	67.9	77.9
0.3	67.9	78.0
0.4	67.9	78.0
0.5	67.8	78.0

Table 4.4: Impact of τ_{corr} on mAP and MOTA on the PoseTrack 2017 validation set.

4.5.4 Effect of Refinement Module and Duplicate Removal

We evaluate the effect of the refinement module and duplicate removal on the pose estimation and tracking performance. As shown in Table 4.3, omitting any of the introduced design choices results in a significant drop in MOTA of at least 1%, and increases the number of identity switches (IDSW). Our proposed correspondence refinement module improves the generated correspondence affinity maps which results in stronger tracking results. This is reflected by the MOTA and mAP scores that drop to 66.9 and 77.7, respectively, if we disable the refinement module. If duplicates are not removed, the MOTA and the mAP scores drop to 64.5 and 77.9, respectively.

4.5.5 Impact of τ_{corr}

We evaluate the impact of τ_{corr} on the pose estimation and tracking performance. As shown in Table 4.4, the threshold has a low impact. We use $\tau_{corr} = 0.3$ for all our experiments.

4.5.6 Comparison with State-of-the-art Methods

We compare the state-of-the-art for multi-frame pose estimation and multi-person pose tracking on the PoseTrack 2017 and 2018 datasets.

Multi-Frame Pose Estimation: For the task of multi-frame pose estimation, we compare the state-of-the-art on the PoseTrack 2017 and 2018 validation sets, respectively. Although our correspondences are trained without using any video data, our approach outperforms the proposed PoseWrapper (Bertasius et al., 2019) approach on the PoseTrack 2018 validation set with mAP of 82.0. It achieves a very competitive mAP on the PoseTrack 2017 validation set with mAP of 80.8 as shown in Table 4.5 on page 82.

Multi-Person Pose Tracking: We compare our tracking approach with the state-of-the-art for multi-person pose tracking on the PoseTrack 2017 and 2018 validation sets and leaderboards. In addition, we perform a post-processing step in which we merge broken tracks similar to the recovery of missed detections described in Section 4.4.1. This further improves the tracking performance.

We submitted our results to the PoseTrack 2017 and 2018 test servers. Our approach achieves top scoring MOTA of 60.0 on the PoseTrack 2017 leaderboard without any bells and whistles as shown in Table 4.6 on page 83. Our tracking performance is on par with state-of-the-art approaches on the PoseTrack 2017 validation set.

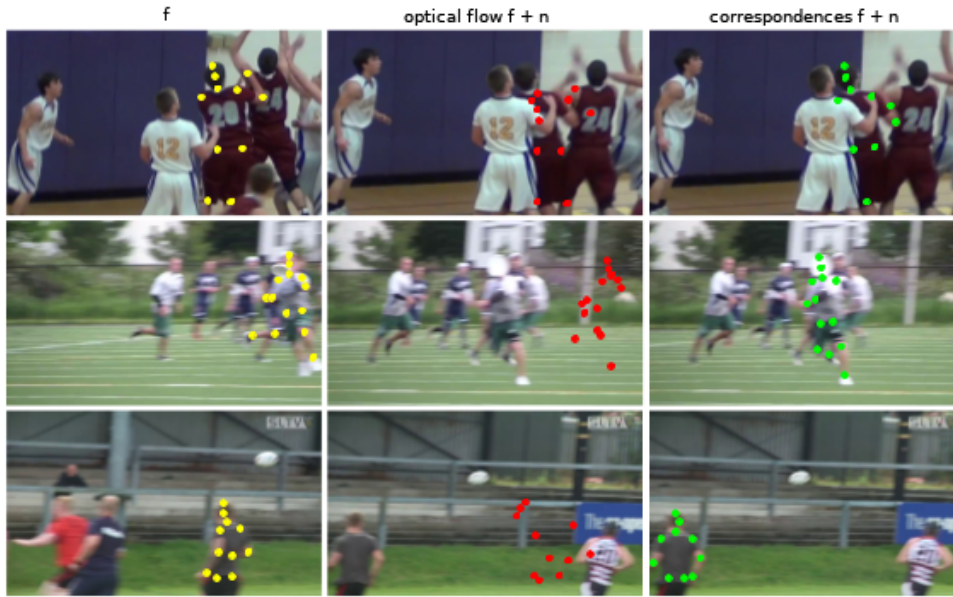


Figure 4.8: Qualitative comparison between optical flow and correspondences for the task of pose warping under occlusion, motion blur, and large motion. (a) Query pose in frame f . (b) Warped pose using optical flow. (c) Warped pose using correspondences. In contrast to optical flow, the correspondences warp the poses correctly despite occlusions, motion blur, or large motion.

Similarly, we achieve top scoring MOTA of 69.1 on the PoseTrack 2018 validation set as shown in Table 4.6. Our tracking results are very competitive with the winning entries on the PoseTrack 2018 leaderboard, although the winning entries use additional training data.

4.5.7 Qualitative Results

We qualitatively compare optical flow and correspondences for the task of pose warping under motion blur, occlusions, and large motion in Figure 4.8. While the column on the left-hand side shows the query pose in frame f , the middle and right-hand side columns show the warped poses generated by optical flow or correspondences, respectively. In contrast to optical flow, our approach is robust to occlusion and fast human or camera motion. Moreover, occlusion leads to missed keypoint detection in the pose estimator or person detector. In Figure 4.9 (page 84), we show qualitative results for the recovery of missed detections. Our approach, however, has also some limitations. For instance, we observe that we sometimes obtain two tracks for the same person if the person detector provides two or more bounding boxes for a person, like one bounding box for the upper body and one bounding box for the full body.

4.5.8 Failure Cases

Existing person detectors sometimes output duplicate detections for the same person. Such duplicate detections are hard to remove using non-maximum suppression. In our experiments, they increase the number of false-positives (FP) and lead to identity-switches. This impacts the overall tracking

performance, as the MOTA metric used in PoseTrack heavily penalizes FPs and IDSWs, as shown in Table 4.3. Figure 4.10 (page 85) illustrates such failure cases.

4.6 Conclusion

In this chapter, we have proposed a self-supervised keypoint correspondence framework for the tasks of multi-frame pose estimation and multi-person pose tracking. The proposed keypoint correspondence framework solves two tasks: (1) recovering missed detections and (2) associating human poses across video frames for the task of multi-person pose tracking. The proposed approach based on keypoint correspondences outperforms the state-of-the-art for the tasks of multi-frame pose estimation and multi-person pose tracking on the PoseTrack 2017 and 2018 datasets.

Dataset	Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	mAP
PoseTrack 17 Val Set	DetectNTrack (Girdhar et al., 2018)	72.8	75.6	65.3	54.3	63.5	60.9	51.8	64.1
	PoseFlow (Xiu et al., 2018)	66.7	73.3	68.3	61.1	67.5	67.0	61.3	66.5
	FlowTrack (Xiao et al., 2018)	81.7	83.4	80.0	72.4	75.3	74.8	67.1	76.7
	HRNet (Sun et al., 2019a)	82.1	83.6	80.4	73.3	75.5	75.3	68.5	77.3
	MDPN (Guo et al., 2018)	85.2	88.5	83.9	78.0	82.4	80.5	73.6	80.7
	PoseWarper (Bertasius et al., 2019)	81.4	88.3	83.9	78.0	82.4	80.5	73.6	81.2
	Ours	86.1	87.0	83.4	76.4	77.3	79.2	73.3	80.8
PoseTrack 18 Val Set	PoseFlow (Xiu et al., 2018)	63.9	78.7	77.4	71.0	73.7	73.0	69.7	71.9
	MDPN (Guo et al., 2018)	75.4	81.2	79.0	74.1	72.4	73.0	69.9	75.0
	PoseWarper (Bertasius et al., 2019)	79.9	86.3	82.4	77.5	79.8	78.8	73.2	79.7
	Ours	86.0	87.3	84.8	78.3	79.1	81.1	75.6	82.0

Table 4.5: Comparison to the state-of-the-art on the PoseTrack 2017 and 2018 validation set for multi-frame pose estimation.

	Approach	mAP	MOTA		Approach	mAP	MOTA
PoseTrack 17 val set	STEmbedding (Jin et al., 2019b)*	77.0	71.8	PoseTrack 18 val set	Ours + Merge	79.2	69.1
	EOGN (Zhang et al., 2019)	76.7	70.1		Ours	79.2	68.8
	PGPT (Bao et al., 2020)	77.2	68.4		MIPAL (Hwang et al., 2019)	74.6	65.7
	Ours + Merge	78.0	68.3		LightTrack (Ning and Huang, 2019)	71.2	64.9
	Ours	78.0	67.9		Miracle ⁺ (Yu et al., 2018)	80.9	64.0
	POINet (Ruan et al., 2019)	-	65.9		OpenSAI (Ning et al., 2019)	69.7	62.4
	HRNet (Sun et al., 2019a)	77.3	-		STAF (Raaj et al., 2019)	70.4	60.9
	FlowTrack (Xiao et al., 2018)	76.7	65.4				
PoseTrack 17 test set	EOGN (Zhang et al., 2019)	74.8	61.1	PoseTrack 18 test set	MSRA ⁺	74.0	61.4
	PGPT (Bao et al., 2020)	72.6	60.2		ALG ⁺	74.9	60.8
	Ours + Merge	74.2	60.0		Ours + Merge	74.4	60.7
	POINet (Ruan et al., 2019)	72.5	58.4		Miracle ⁺ (Yu et al., 2018)	70.9	57.4
	LightTrack (Ning and Huang, 2019)	66.8	58.0		MIPAL (Hwang et al., 2019)	67.8	54.9
	HRNet (Sun et al., 2019a)	75.0	58.0		CV-Human	64.7	54.5
	FlowTrack	74.6	57.8				

Table 4.6: Comparison to the state-of-the-art on the PoseTrack 2017 and 2018 validation and test sets. Approaches marked with ⁺ use additional external training data. Approaches marked with * do not report results on the official test set.

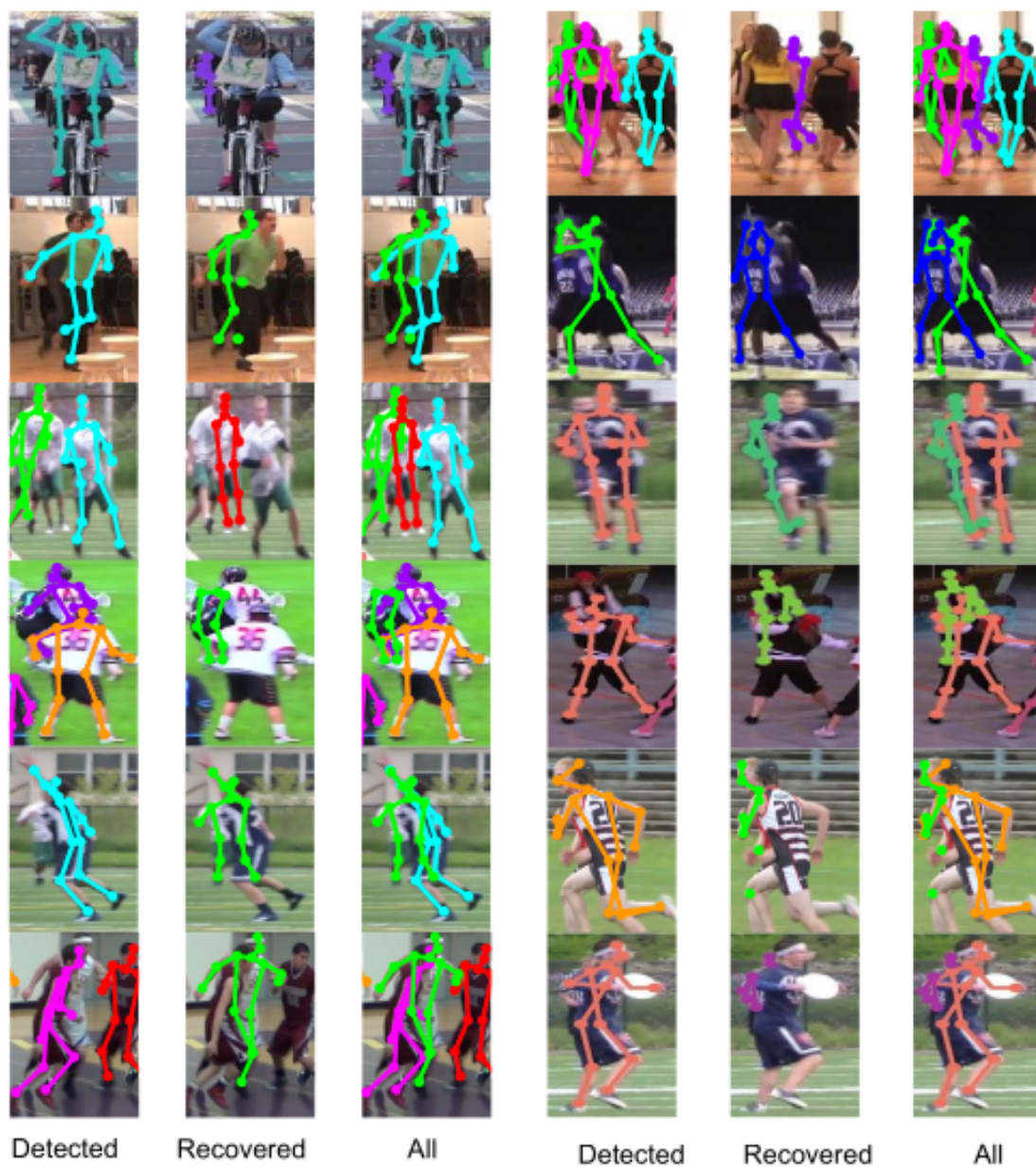


Figure 4.9: Qualitative results for recovering missed detections. Best seen using the zoom function of the PDF viewer.



Figure 4.10: Failure cases. Duplicates by the person detector lead to multiple tracks of the same person and negatively impact the tracking performance.

PoseTrack21: A Dataset for Person Search, Multi-Object Tracking and Multi-Person Pose Tracking

In this chapter, we introduce PoseTrack21, a novel dataset featuring dense annotations of person keypoints, person bounding boxes, and unique person identities. PoseTrack21 stands out due to its wide range of pose diversity, encompassing even smaller individuals. Beyond facilitating the evaluation of multi-person pose estimation and tracking methods, this dataset also supports the assessment of multi-object tracking and person search pipelines. Our experiments show that these related tasks, multi-person pose tracking, multi-object detection, and person search, share similarities and can complement each other effectively. In addition, we propose a new metric, Keypoint HOTA, which enhances our ability to measure the accuracy and robustness of multi-person pose tracking methods.

Individual Contribution: The following chapter is based on the publication [Doering et al. \(2022\)](#):

PoseTrack21: A Dataset for Person Search, Multi-Object Tracking and Multi-Person Pose Tracking

Andreas Doering, Di Chen, Shanshan Zhang, Bernt Schiele, and Juergen Gall
IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2022

This work was carried out in collaboration with Di Chen. Andreas Doering developed the annotation tool and provided annotations for the entire dataset, while Di Chen contributed to the annotation of unique person identities. Andreas Doering conducted the remaining work under the scientific supervision of Juergen Gall. During the publication process, Shanshan Zhang and Bernt Schiele assisted with revising the manuscript. Bernt Schiele generously provided additional funding for the annotation process.

Contents

5.1	Introduction	88
5.2	Related Datasets	89
5.3	The PoseTrack21 Dataset	92
5.4	Multi-Person Re-ID Pose Tracking	94
5.4.1	Proposed CorrTrack Baselines	94
5.4.2	Proposed Tracktor++ Baselines	95

5.5	Analysis	96
5.6	Dataset Statistics	102
5.7	Ablation Studies	102
5.8	Discussion	103

5.1 Introduction

Multi-person pose tracking (Xiao et al., 2018; Zhang et al., 2019; Snower et al., 2019; Wang et al., 2020c; Rafi et al., 2020; Yang et al., 2021b), multi-object tracking (Bergmann et al., 2019; Zhang et al., 2021; Wang et al., 2020d; Zhou et al., 2020b; Brasó and Leal-Taixé, 2020; Dai et al., 2021), and person search (Li and Miao, 2021; Chen et al., 2020b,a; Xiao et al., 2017b; Yan et al., 2021; Chen et al., 2021a) are very active research areas in computer vision and relevant for many applications like sports, autonomous driving, and security. Although these research areas share very common sub-tasks, they are currently studied independently, and the progress in one area does not necessarily yield improvement in the other areas. For instance, person search aims at re-identifying a query person within a set of target images or video frames. In contrast to person re-identification, person search also requires the detection of all persons in an image. It is more difficult due to inaccurate detections, missed detections, or false positives. Person search is thus a related sub-task of multi-object tracking where persons must be detected in all frames and associated across frames for tracking. If a person is occluded for several frames, the association problem is similar to person search. Approaches for multi-object tracking thus often use a person re-identification module (Zhou et al., 2021; Luo et al., 2019a; Hermans et al., 2017; Zhou et al., 2019; Hou et al., 2020a). Multi-object tracking, however, considers only the identities of a person within a video sequence but not across videos. In contrast, person search aims to identify persons even across videos. Finally, multi-person pose tracking is related to multi-object tracking. Still, instead of estimating bounding boxes for each person, the full pose needs to be estimated, including which joints are visible or not.

The reason why multi-person pose tracking, multi-object tracking, and person search are studied independently is the lack of a dataset that allows the evaluation of all three tasks jointly on real video sequences. Datasets for multi-object tracking (Leal-Taixé et al., 2015; Milan et al., 2016; Dendorfer et al., 2019; Ristani et al., 2016a) contain track-ids and bounding boxes, but no ground truth for human poses and person-ids across videos. Datasets for multi-person pose tracking (Iqbal et al., 2017; Andriluka et al., 2018) do not contain person-ids across video sequences either, but they also do not contain bounding boxes. While it is possible to compute bounding boxes from human poses, they are not accurate and reliable due to occlusions and truncations that occur highly frequently in these datasets. Synthetic datasets (Fabbri et al., 2018, 2021) that are generated using the computer game engine GTA are an exception, but they are intended for training and cannot be used for evaluating the performance of approaches on real-world data. Furthermore, they lack the pose diversity of real videos, which also include diverse sports sequences. In this work, we close this gap and propose PoseTrack21, a large-scale dataset for multi-person pose tracking, multi-object tracking, and person search. It is based on the extended set of videos from the PoseTrack 2018 training and validation set (Andriluka et al., 2018), but the videos are completely re-annotated. Besides the refined human poses, the dataset contains accurate annotations of joint occlusions, accurate bounding

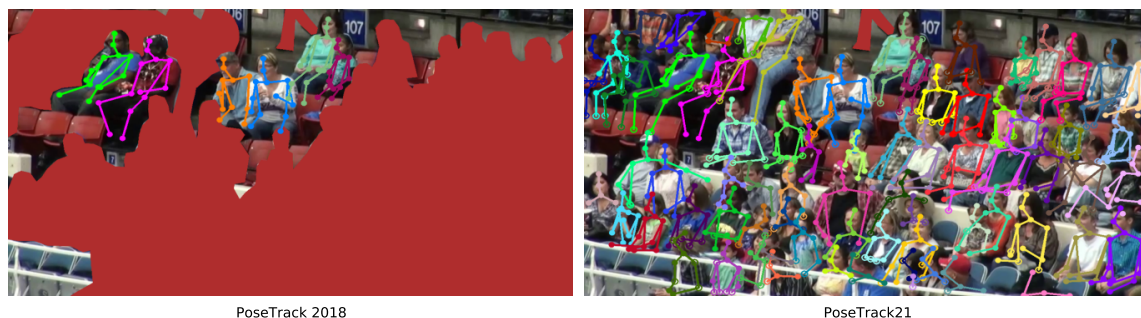


Figure 5.1: Comparison between PoseTrack 2018 (Andriluka et al., 2018) and PoseTrack21. We densely annotated crowded scenes to increase the difficulty of multi-person pose tracking, multi-object tracking, and person search. Ignore regions are drawn in red color. Best viewed in color with a PDF reader.

box annotations even for small persons, and person-ids within and across video sequences. The dataset can thus be used to evaluate approaches for multi-person pose tracking, multi-object tracking, and person search. Furthermore, the dataset allows us to compare methods for multi-person pose tracking and multi-object tracking, which has been impossible so far due to missing bounding boxes or human pose annotations. We thus propose a few baselines where we combine techniques for multi-person pose tracking, multi-object tracking, and person search and address the question of whether extending approaches for multi-object tracking to multi-person pose tracking or vice versa is more promising. We finally provide a detailed analysis regarding the strengths and limitations of existing methods and baselines. The dataset and source code of baselines are available at <https://github.com/andoer/PoseTrack21>.

5.2 Related Datasets

We discuss the most commonly used datasets for the tasks of multi-person pose tracking (Iqbal et al., 2017; Andriluka et al., 2018; Fabbri et al., 2018), multi-object tracking (Leal-Taixé et al., 2015; Milan et al., 2016; Dendorfer et al., 2019; Ristani et al., 2016a; Kumar et al., 2020) and person search (Xiao et al., 2017a; Zheng et al., 2017a; Kumar et al., 2020) and summarize the major differences in Table 5.1. Compared to previous multi-person pose tracking datasets, PoseTrack21 contains around 22% more human pose annotations per sequence. Further, we provide over 420,000 additional bounding box annotations. Therefore, our dataset contains many more person instances than previous datasets for multi-person pose tracking. In addition, PoseTrack21 is the only dataset for the task of multi-person pose tracking, multi-object tracking, and person search since it provides continuous person identities throughout the entire dataset. In the following, we describe the other datasets in more detail.

Multi-Person Pose Tracking: Multi-person pose tracking datasets (Iqbal et al., 2017; Andriluka et al., 2018) are large-scale datasets for multi-person pose estimation and tracking in videos. PoseTrack consists of challenging scenarios from multiple in-the-wild scenarios, such as sports and danc-

Dataset	# Boxes	# Poses	track-ids	person-ids	# queries	real
MP PoseTrack (Iqbal et al., 2017)		16,219	✓			✓
PoseTrack 2017 [†] (Andriluka et al., 2018)		80,144	✓			✓
PoseTrack 2018 [†] (Andriluka et al., 2018)		144,688	✓			✓
MOT15 (Leal-Taixé et al., 2015)	101,345		✓			✓
MOT17 (Milan et al., 2016)	292,733		✓			✓
MOT20 (Dendorfer et al., 2019)	1,652,040		✓			✓
DukeMTMC [◊] (Ristani et al., 2016a)	46,261		✓	✓		✓
PathTrack (Manen et al., 2017)	16,287 [‡]	✓			✓	
CUHK-SYSU (Xiao et al., 2017a)	96,143			✓	2,900	✓
PRW (Zheng et al., 2017a)	34,304			✓	932	✓
P-DESTRE (Kumar et al., 2020)	~ 14.8 M			✓	253	✓
PoseTrack-ReID [◊] (Fan et al., 2020)	84,443			*		✓
PoseTrack21	428,949	177,164	✓	✓	1,313	✓
JTA (Fabbri et al., 2018)		~10 M	✓			
MotSynth (Fabbri et al., 2021)		~ 40M		✓		

Table 5.1: Comparison with different datasets for multi-person pose tracking, MOT, and person search. Datasets marked with [†]: only training and validation set. *: no manual annotations. [◊]: dataset not available. [‡]: Only reports the total number of tracks.

ing, with a high degree of occlusion in many crowded scenes. The videos also strongly differ in terms of camera view and camera motion. The PoseTrack 2017 dataset (Iqbal et al., 2017; Andriluka et al., 2018) follows the split of the MPII Human Pose dataset (Andriluka et al., 2014), which splits the dataset into 292, 50 and 208 videos for training, validation and testing. In total, PoseTrack 2017 provides around 23,000 labeled frames with 153,615 annotated poses, where each pose is annotated with 15 keypoints. PoseTrack 2018 extends the previous dataset and contains 593, 170, and 375 videos for training, validation, and testing. The majority of the sequences range from 41 to 151 frames and contain 30 densely annotated frames around the middle of each sequence. In addition, validation and test sequences are densely annotated with a step size of four frames. In total, PoseTrack 2018 consists of 46,933 labeled frames¹.

We provide the statistics for the extended version in Table 5.1. As the test set is not publicly available, we can only report the number of total poses for the training and validation set. In addition to the human poses, PoseTrack provides ignore regions to exclude crowds and small people that have not been annotated, head bounding boxes to estimate the scale of a person, which is required for evaluation, and track identities. Unfortunately, track identities are not unique throughout the dataset, not even within a single sequence. For instance, if a person leaves a scene and re-enters, it is assigned a new track id. This can result in ambiguities for appearance-based similarity approaches.

Multi-Object Tracking: The most important benchmark for multi-object tracking is the MOTChallenge², which consists of three separate tracking benchmarks: 2D MOT 15 (Leal-Taixé et al., 2015), MOT16/17 (Milan et al., 2016) and MOT20 (Dendorfer et al., 2019). Each benchmark consists of challenging person tracking sequences, mostly in surveillance scenarios or street scenes

¹On training and validation. The testing set can not be measured, as the annotations are not publicly available.

²<https://motchallenge.net>

with several degrees of occlusions and crowded scenarios. For instance, the MOT20 dataset is split into a training and a testing subset, each containing four sequences. Additionally, the dataset contains annotations of different object classes, such as cars, reflections, or crowds, which are ignored during the evaluation. MOT20 sequences are mostly limited to surveillance scenarios in which the persons are in an upright position. The diversity of poses is thus much lower compared to datasets for multi-person pose tracking. Similar to PoseTrack, track identities are not always continued, e.g., if a person leaves and re-enters the scene, a new track id is associated. Especially in very crowded scenarios, this results in many identity switches, potentially harming the training of appearance-based association methods, such as person re-identification.

DukeMTMC (Ristani et al., 2016a) is another dataset for multi-object tracking with a surveillance-based setup. Differently, DukeMTMC provides recordings of the same scene from different cameras and provides a benchmark for multi-target multi-camera tracking. For that reason, this dataset provides unique person identities throughout the entire dataset. Unfortunately, this dataset is no longer available due to ethics issues. Manen et al. (2017) propose another large-scale MOT dataset called PathTrack, containing 720 sequences with a total length of 172 minutes and a total of 16287 person trajectories. Similar to PoseTrack, the dataset contains sequences of different categories, such as sports, dancing or street.

Person Search and Person Re-Identification: Both person search datasets (Xiao et al., 2017a; Zheng et al., 2017a; Kumar et al., 2020) and video-based person re-identification datasets (Zheng et al., 2016; Wu et al., 2018; Kumar et al., 2020) are divided into query and gallery subsets. The query subset contains all persons of interest that must be matched with the gallery subset. Query images are usually provided as tight person crops and mostly contain the person of interest only. However, in many real-world scenarios, such as surveillance or sports, persons are often occluded by obstacles. Further, these scenarios contain a lot of crowded scenes in which persons frequently occlude each other. This leads to many ambiguities, especially if a person is partially visible or multiple persons are present within a single crop. On the contrary, PoseTrack21 provides queries of variable difficulty ranging from single-person crops to highly occluded scenarios in which multiple persons are visible within a single crop. In this way, the dataset allows to study person search in a realistic setup and scenarios that differ from surveillance-like footage with limited pose variations. Figure 5.2 shows a few examples of the queries of PoseTrack21. Further, we provide pose annotations for every query person, which can be used as additional guidance for person search.

(Fan et al., 2020) is another dataset based on the PoseTrack 2017 dataset, which was annotated for the purpose of video-based person re-identification. The authors calculated bounding boxes and removed persons with less than 6 keypoints based on the keypoint annotations. In contrast to PoseTrack21, the identities have not been annotated. Instead, person identities were obtained by an unsupervised approach without additional verification. The annotations are furthermore not available. Similarly, (Chen et al., 2022) also evaluates video-based person re-identification on PoseTrack and extracts tracklets from the PoseTrack 2018 videos. Both protocols (Fan et al., 2020; Chen et al., 2022) are for person re-identification but not for person search.

Synthetic Datasets: The Joint Track Auto (JTA) (Fabbri et al., 2018) dataset is generated from a video game. As with other synthetic datasets, the dataset is intended for training, but it is not suitable



Figure 5.2: From left to right: person search queries in PoseTrack21 with increasing difficulty.

for evaluating the performance of approaches on real-world data. The dataset contains over 10M annotated human body poses within over 460,888 densely annotated frames. The dataset contains 256 videos for the training and validation sets, respectively. Unlike PoseTrack, track IDs are uniquely assigned within each sequence.

The MOTSynth (Fabbri et al., 2021) dataset is generated similarly from the same video game and combines 128 sequences from the JTA dataset with 256 new scenes. All scenes were rendered with different weather conditions and during day and night, totaling 576 and 192 generated scenes for training and validation with more than 40M bounding boxes, over 1.3M annotated frames, and 9519 unique person IDs. On top of that, MOTSynth provides 3D poses, segmentation masks, and depth information. In both datasets, however, the diversity of human poses is very low compared to PoseTrack.

5.3 The PoseTrack21 Dataset

For creating PoseTrack21, we use the videos of the training and validation set of the extended version of the PoseTrack 2018 (Andriluka et al., 2018) dataset. It contains 593 videos for training and 170 videos for validation. The annotation has been performed in several steps. First, we annotated the bounding boxes. Since PoseTrack provides pose annotations only for some keyframes, we first annotated the bounding boxes for all keyframes. To this end, we visualized the annotated poses in PoseTrack for a frame. We asked the annotators to annotate all persons where the head is visible since the head size is needed for the evaluation (Andriluka et al., 2014). This also included persons who had not been annotated in PoseTrack 2018, particularly small persons and persons in crowds. The annotators were asked to draw a tight bounding box that covers the entire person, including occluded body parts. We interpolated and manually revised the bounding boxes between the keyframes in the second step. In the third step, another person verified all bounding box annotations. In the fourth step, we marked ignore regions in each frame containing persons not annotated. The ignore regions

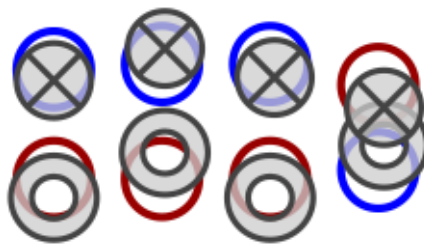


Figure 5.3: While the blue and red person (ground-truth) cross their path at the last frame, both tracks (cross, ring) keep straight on. In the last frame, the identities of both persons are therefore wrongly estimated, which is penalized by keypoint HOTA. HOTA (Luiten et al., 2020) does not penalize this case if the cross overlaps with the blue circle and the ring with the red circle higher than a threshold α .

have been verified by a second person and ensure that a method that makes a prediction for a person that has not been annotated is not penalized. In the fifth step, head bounding boxes are annotated for the validation set. The head bounding boxes are required for the evaluation metric (Andriluka et al., 2014) and not required for training. In parallel, unique person identities throughout the videos of the training and validation set have been annotated.

In the final step, we adapt and annotate person keypoints for all keyframes on the training and validation sets. The original annotations of the PoseTrack 2018 dataset consist of 15 keypoints, each containing a flag, whether it is annotated or not. Unfortunately, several cases exist in which these flags are not set reliably. In our dataset, we redefine the purpose of the keypoint flags and include occluded keypoints. In that way, pose estimation, re-ID, and tracking approaches can utilize occlusion information within their training pipelines. We define a joint $\mathbf{x} = (x, y, v)$ as occluded, if $x > 0$, $y > 0$ and $v = 0$. A joint is truncated if $x = 0$, $y = 0$, and $v = 0$. Otherwise, a joint is defined as visible if $v = 1$. After refining the original keypoints, we ran an off-the-shelf pose estimator (Rafi et al., 2020) to estimate the poses for all newly added bounding boxes on keyframes only, which were then refined manually afterwards. All annotated poses have been verified and, if necessary, corrected by a second person. In total, 23 annotators worked on the dataset with over 16,000 person-hours.

Person Search: In the context of person search, we sampled 1313 person queries with varying sizes, camera motion, and degree of occlusion. Especially in cases of occlusion, queries can contain multiple persons. As this results in ambiguities, we additionally provide keypoint information for the person of interest as shown in Figure 5.2. In this way, we aim to encourage researchers to focus on more challenging scenarios for person search.

Data Format: We provide different data formats for the respective tasks, which are very similar to those used in related datasets. We keep the format used in (Andriluka et al., 2018) for multi-person pose tracking and person search. We adopt the format proposed in (Leal-Taixé et al., 2015) for multi-object tracking. This has the advantage that researchers from different communities do not need to change their approaches, can easily read the annotations, and save the results for evaluation.

Method	mAP	MOTA	MOTP	FP	IDSW	FN	TP
CorrTrack [†] (Chapter 4)	72.0	62.6	87.7	58922	14896	164823	485634
CorrTrack (Chapter 4)	72.3	63.0	87.3	59130	15272	161995	488484
CorrTrack (Chapter 4) w. ReID	72.7	63.8	87.3	62604	9436	158720	491712
Tracktor++(Bergmann et al., 2019) w. poses	71.4	63.3	87.3	59850	8145	166886	483558
Tracktor++(Bergmann et al., 2019) w. correspondences	73.6	61.6	86.6	75663	20754	147929	502588
CorrTrack (Chapter 4) (offline)	72.3	63.9	87.3	59132	9577	161997	488482

Table 5.2: Multi-person pose tracking baselines evaluated with the keypoint MOTA metric on PoseTrack21. Approaches marked with [†] have a model trained on PoseTrack 2018 (Andriluka et al., 2018).

Keypoint HOTA: HOTA (Luiten et al., 2020) (cf. Section 3.6.3) is an evaluation metric that has been originally proposed for the task of MOT. It tries to balance detection accuracy and association accuracy of underlying tracks. For the association of detected tracks to ground truth tracks, HOTA utilizes a localization similarity, which relies on the IoU of bounding boxes. For the task of multi-person re-id pose tracking, we extend HOTA to keypoint HOTA. In particular, we rely on the head-normalized percentage of correct keypoints (PCKh) (Andriluka et al., 2014) as localization similarity. This allows us to measure the tracking performance on a keypoint level, and we define the person-level tracking performance as the average tracking performance among all keypoint classes.

Furthermore, HOTA aims to maximize true positive assignments and prefers an assignment with minimal identity switches (IDSW) to an assignment with high localization accuracy. This is achieved by an additional global alignment score (cf. (15) in (Luiten et al., 2020)). PoseTrack21 contains very challenging scenarios with high degrees of occlusions. For that reason, we aim to penalize re-identification errors strictly and do not use the global alignment score in keypoint HOTA since it tolerates identification errors if two persons are close, as illustrated in Figure 5.3.

The source code of the evaluation metrics is available at <https://github.com/andoer/PoseTrack21>.

5.4 Multi-Person Re-ID Pose Tracking

In the following, we describe the baselines we propose for multi-person re-id pose tracking. The first baseline, which will be described in Section 5.4.1, builds on the approach discussed in Chapter 4 that we extend by including a person re-identification module to re-identify persons after occlusions or after they re-enter the scene. The second type of baselines, which will be described in Section 5.4.2, extends the multi-object tracking approach (Bergmann et al., 2019) to multi-person pose tracking.

5.4.1 Proposed CorrTrack Baselines

In Chapter 4, we introduced CorrTrack, a multi-person pose tracking approach that utilizes a keypoint correspondence network. The approach comprises three steps. Given a new frame, the approach first detects the persons. For a fair comparison among the baselines, we use the same Faster R-CNN object detector (Ren et al., 2015), which is also used in (Bergmann et al., 2019), instead of the detector that has been used in Chapter 4. The approach then estimates the human pose for each detected bounding box using a pose estimator, which consists of multiple stages of an adapted

Method	mAP	HOTA	FA-HOTA	DetA	LocA	AssA	FragA
CorrTrack (Chapter 4)	72.3	51.13	51.07	45.48	81.94	58.02	57.75
CorrTrack (Chapter 4) w. ReID	72.7	52.71	52.59	46.56	81.93	60.21	59.66
Tracktor++(Bergmann et al., 2019) w. poses	71.4	52.21	52.03	46.30	81.95	59.41	58.61
Tracktor++(Bergmann et al., 2019) w. correspondences	73.6	48.90	48.43	44.67	81.26	54.05	52.02
CorrTrack (Chapter 4) (off-line)	72.3	52.42	52.29	45.48	81.94	60.93	60.37

Table 5.3: Multi-person pose tracking baselines evaluated with the keypoint HOTA metric on PoseTrack21 .

GoogleNet (Szegedy et al., 2015). We also use the same pose estimator for the other baselines for a fair comparison. In addition, the approach propagates the poses from the previous frame to the current frame using the keypoint correspondence network. Since occluded joints can become visible in the next frame, the approach re-estimates the poses for the propagated poses using the pose estimator. After applying a non-maximum suppression on the poses of the new frame, the remaining poses are matched to the poses of the previous frame by bipartite graph matching, where the similarity between two poses is measured by the affinity maps generated by the keypoint correspondence network. If a pose from the previous frame cannot be matched, the corresponding track ends. A new track starts if a pose of the new frame is not matched to a pose of the previous frame. This frame-wise matching results in a high number of identity switches. In Chapter 4, we also proposed an offline version where tracks can be merged in an additional post-processing step based on the keypoint correspondence similarity. We will report the results for the on-line and off-line variant.

In order to be able to track a person online and to reduce the number of identity switches due to occlusions or re-entering the scene, we keep a history of tracks that ended the latest $F = 10$ frames ago. If a pose in the new frame is not matched to a track with a pose in a previous frame, we match the pose to the tracks in the history. Note that a track is only added to the history if it is inactive, i.e., it does not contain a pose from the previous frame. For the matching, we use the SeqNet model (Li and Miao, 2021), which we will also evaluate for the task of person search. We compute the re-identification features for the bounding box of the pose that has not been matched and the average re-identification features of the inactive track where we use at most the last F frames of the tracks. The similarity between a pose and an inactive track is then computed by the cosine similarity of the corresponding feature vector. The matching between all inactive tracks and unmatched poses is then performed using the Hungarian algorithm (Kuhn, 1955). If the similarity of a match is higher than a threshold $\tau = 0.5$, the matched track is re-activated. The unmatched detections initiate new tracks. We denote this variant by *CorrTrack with ReID*.

5.4.2 Proposed Tracktor++ Baselines

Tracktor++ (Bergmann et al., 2019) is an online multi-object tracking approach based on Faster R-CNN (Ren et al., 2015). Tracktor++ aligns frames via image registration by enhanced correlation coefficient maximization (Evangelidis and Psarakis, 2008) during tracking. For sequences with low frame rates, Tracktor++ additionally applies a constant velocity assumption for all tracked objects. By applying the respective motion models, bounding boxes of active tracks are then warped into the current frame. Further, warped bounding boxes are refined by the bounding box regression branch of (Ren et al., 2015). After bounding box regression, warped boxes with low confidence are removed,

and the respective tracks are deactivated. Furthermore, non-maximum suppression based on bounding box intersection over union is applied on all remaining warped and detected bounding boxes. In the second step, unmatched detections are associated with inactive tracks. For the at most last $F = 10$ bounding boxes of an inactive track, the average appearance features extracted from a re-identification model (Hermans et al., 2017) are computed and compared to the appearance feature of each unmatched detection. The Euclidean distance is used as the distance between two appearance feature vectors. The remaining unassociated detections initiate new tracks. In two ways, we extend the approach (Bergmann et al., 2019) towards multi-person re-id pose tracking. In the first setting, we evaluate Tracktor++ on PoseTrack21 without the constant velocity assumption. Additionally, we remove small tracks with less than three frames as they are likely to be false positives. Afterwards, we estimate the pose for each track with the pose estimation model from Chapter 4. We denote this approach by *Tracktor++ with poses*.

In the second setting, we replace the motion model with a pose-warping module based on the keypoint correspondence network from Chapter 4. First, we warp the keypoints of the last pose of all active tracks into the next frame. Since occluded joints might become visible, we calculate a bounding box from the warped keypoints and use the bounding box regression module from (Bergmann et al., 2019) for bounding box refinement. Second, we re-estimate the poses with the pose estimation model. We perform tracking in a greedy fashion based on non-maximum suppression and pose similarity. The calculation of pose similarity between the warped track poses and estimated poses is performed as in (Xiao et al., 2018). Tracks that can not be associated become inactive. The association of unmatched detected poses with inactive tracks is done as before. We denote this approach by *Tracktor++ with correspondences*.

5.5 Analysis

We evaluate the performance of related state-of-the-art methods for the tasks of multi-person re-id pose tracking, multi-object tracking, and person search on our proposed dataset and analyze strengths and weaknesses.

Evaluation Metrics: In the context of multi-person pose tracking, we use the keypoint-based MOTA metric proposed in (Andriluka et al., 2018) for evaluation. Contrary to the standard MOTA metric (Leal-Taixé et al., 2015) for multi-object tracking, keypoint-based MOTA evaluates the tracking performance for every keypoint class individually. In the context of PoseTrack21, this results in 15 different MOTA scores, which are then averaged into a final MOTA score. The MOTA metric is generally highly impacted by the localization accuracy of keypoint detections (Luiten et al., 2020). Hence, better person detectors or pose estimators directly result in a stronger MOTA score.

In contrast, HOTA (Luiten et al., 2020) tries to balance detection accuracy and association accuracy of underlying tracks. For that reason, HOTA consists of sub-metrics measuring the detection accuracy (DetA), the localization accuracy (LocA), the association accuracy (AssA), and the fragmentation accuracy (FragA). FragA penalizes heavily fragmented tracks and extends HOTA to a fragmentation-aware HOTA metric (FA-HOTA).

For the evaluation of multi-person re-id pose tracking, we propose keypoint HOTA and replace HOTA’s localization similarity by the head-normalized percentage of correct keypoints (PCKh) (Andriluka et al., 2014). Further, and in contrast to HOTA, we strictly penalize identification errors.

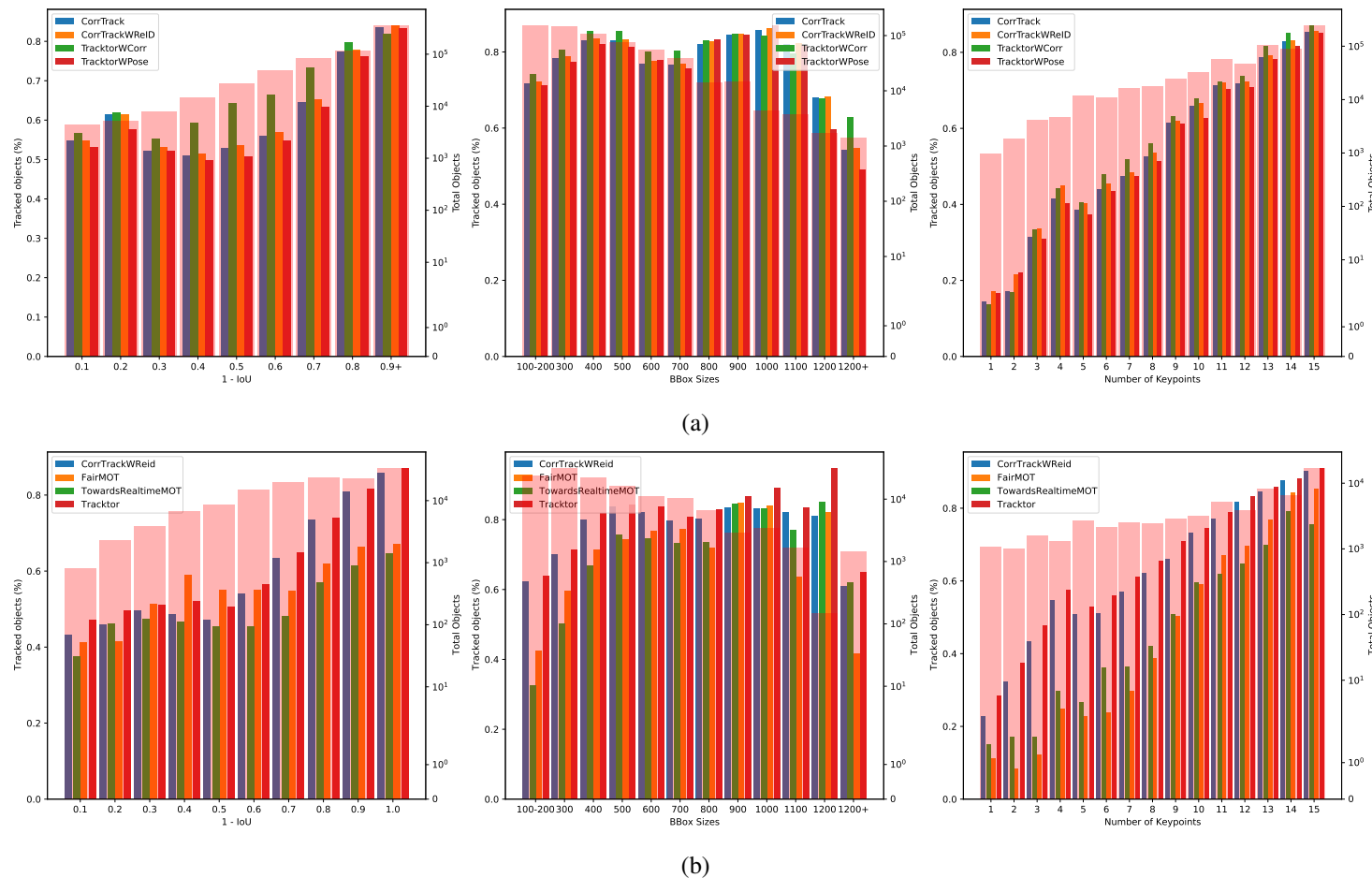


Figure 5.4: (a) Pose tracking ablation studies. The plots show the recall (tracked objects) and the number of objects for different bounding box visibility (1-IoU), the size of the bounding box, and the number of keypoints. (b) MOT ablation studies. The plots show the recall (tracked objects) and the number of objects for different bounding box visibility (1-IoU), the size of the bounding box, and the number of keypoints.

Multi-Person Re-ID Pose Tracking: For a fair comparison, we utilized the same person detector and the same pose estimation model for all our baselines. In particular, we use a Faster R-CNN (Ren et al., 2015) with a Resnet50-FPN (Lin et al., 2017a) as our person detector. The model is pre-trained on MSCOCO (Lin et al., 2014), which we obtained from the TorchVision model-zoo³. We further fine-tuned the person detector on PoseTrack21 for 30 epochs, following the training protocol proposed in (Bergmann et al., 2019). Similarly, we train a three-stage pose estimation model from Chapter 4 on MSCOCO and PoseTrack21 for 215 and 16 epochs, respectively. The learning rate was reduced from 1e-3 to 1e-4 after 200 epochs and further reduced to 1e-5 after 4 epochs on PoseTrack21.

We report mAP for the pose estimation performance based on the PCKh-metric proposed in (Andriluka et al., 2018) and evaluate the tracking performance of our baselines with two different metrics, namely MOTA (Andriluka et al., 2018; Leal-Taixé et al., 2015) and HOTA (Luiten et al., 2020). Table 5.2 and Table 5.3 summarize the results. Note the difference in mAP: It is common practice in multi-person pose tracking approaches (Xiao et al., 2018; Zhang et al., 2019; Snower et al., 2019; Wang et al., 2020c; Rafi et al., 2020; Yang et al., 2021b) to sacrifice mAP for better MOTA scores, consequently, differences in outlier handling result in different pose estimation results.

Nearly all proposed baselines outperform the online version of *CorrTrack* in terms of MOTA and HOTA. *CorrTrack w. ReID* even outperforms the offline version of *CorrTrack* in terms of HOTA. This shows that a tracklet history in combination with appearance-based feature matching can boost the overall tracking performance. On the other hand, keypoint correspondences do not seem to work reliably as a motion model (*Tracktor++ w. correspondences*) for two reasons: 1) keypoint correspondences can only warp visible keypoints of the previous frames, which results in bounding boxes that do not cover the entire person. In consequence, the appearance features have a limited descriptiveness. 2) the keypoint similarity-based non-maxima suppression (NMS) fails to remove duplicate detections that cover different keypoints. A look into Table 5.3 further confirms this behavior: the association accuracy (AssA) and fragmentation accuracy (FragA) are much lower compared to the remaining baselines. *CorrTrack w. ReID* and *Tracktor++ w. poses* show a similar tracking performance, though *CorrTrack w. ReID* has a higher FragA and generates less fragmented tracks.

Further, we evaluated the performance of all baseline methods with respect to different attributes, such as the size of the bounding box, bounding box visibility, and the number of keypoints. The results are shown in Figure 5.4a. Bounding box size denotes the maximum side length of a given bounding box. As tracking results did not contain bounding box information, bounding boxes were generated from keypoints. Interestingly, all baselines perform best for bounding box sizes between 400-500 and 800-1000 pixels. In the person visibility study, where we measured the IoU between the person’s ground truth bounding boxes and denoted visibility as $1 - \text{IoU}$, all baselines perform surprisingly well in highly occluded scenarios ($1 - \text{IoU} \in [0.1, 0.2]$). However, a closer look at the performance for different numbers of visible keypoints reveals that the recall massively drops when persons are only partially visible. Figure 5.4a further reflects our previous conclusion on the performance of *TracktorWCorr* that achieves a higher recall at the cost of more false positives and identity switches.

³<https://pytorch.org/vision/stable/models.html>

Method	IDF1	IDP	IDR	Rcll	Prcn	MOTA	MOTP
TRMOT (Wang et al., 2020d)	57.3	70.0	46.6	59.2	85.5	47.2	75.4
FairMOT (Zhang et al., 2021)	63.2	81.0	51.8	60.6	94.9	56.3	80.5
Tracktor++ (Bergmann et al., 2019)	69.3	76.4	63.5	71.6	86.2	59.5	80.7
CorrTrack + ReID	66.5	72.4	61.4	68.8	81.2	52.0	78.9

Table 5.4: Multi-object tracking baselines evaluated with the MOTA metric on PoseTrack21MOT.

Method	HOTA	DetA	AssA	LocA	RHOTA
TRMOT (Wang et al., 2020d)	46.85	40.91	54.98	79.92	49.06
FairMOT (Zhang et al., 2021)	53.53	47.43	61.45	83.16	55.37
Tracktor++(Bergmann et al., 2019)	58.29	52.71	65.43	83.09	62.58
CorrTrack + ReID	56.95	51.33	64.19	82.80	61.86

Table 5.5: Multi-object tracking baselines evaluated with the HOTA metric on PoseTrack21MOT.

Multi-Object Tracking: We evaluate the performance of state-of-the-art multi-object tracking approaches (Wang et al., 2020d; Bergmann et al., 2019; Zhang et al., 2021) on our PoseTrack21 dataset. In particular, each baseline provided a pre-trained model on the MSCOCO dataset. We fine-tuned each baseline on PoseTrack21, following the training protocols as proposed in (Wang et al., 2020d; Bergmann et al., 2019; Zhang et al., 2021).

For *Tracktor++*, we additionally trained TriNet (Hermans et al., 2017), a re-identification model based on ResNet50 (He et al., 2016) as proposed in (Bergmann et al., 2019): For each minibatch, we sampled 18 crops of size 256×128 from the PoseTrack21-MOT subset and trained TriNet for 29270 iterations. Both *TRMOT* (Wang et al., 2020d) and *FairMOT* (Zhang et al., 2021) directly train a dedicated re-identification head within their proposed object detection networks, using the cross-entropy loss.

Similar to Section 5.5, we measure the performance with the MOTA and HOTA metrics. We also evaluate IDF1, IDP, and IDR (Ristani et al., 2016b), which are common metrics for evaluating the MOT performance (cf. Section 3.6.3). We also report the recall-HOTA (RHOTA) (Luiten et al., 2020), which combines detection recall and association accuracy.

We evaluated the performance of an MOT baseline on the task of multi-person pose tracking in Section 5.5. Consequently, we want to assess how well multi-person pose tracking approaches can solve the task of MOT. In this regard, we calculated bounding boxes from all poses and removed all pose information from *CorrTrack w. ReID*, which we introduced in Section 5.4.1 and converted the results into the respective MOT format. As shown in Table 5.4 and Table 5.5, *CorrTrack w. ReID* achieves competitive results. This confirms that the MOT and pose tracking tasks complement each other.

In additional ablative experiments, we evaluated the overall tracking performance for different features, such as bounding box size, visibility, and the number of keypoints similar to Section 5.5. Based on different bounding box sizes, *Tracktor++* significantly outperforms the remaining baselines in the amount of tracked objects, independent of the size. Note that the total number of objects is reported on a semi-logarithmic scale. A similar behavior applies to the number of available keypoints. In terms of bounding box visibility ($1 - \text{IoU}$), *Tracktor++* struggles to properly track in occluded

Method	Recall	AP	mAP	top-1	top-5	top-10
OIM [†] (Xiao et al., 2017b)	68.05	63.84	57.58	86.82	88.65	89.49
NAE(Chen et al., 2020b)	67.98	62.29	55.83	84.31	87.59	88.96
NAE+ (Chen et al., 2020b)	74.45	62.56	54.48	83.93	87.89	89.11
HOIM (Chen et al., 2020a)	79.54	66.08	52.77	84.69	87.66	88.35
SeqNet (Li and Miao, 2021)	72.49	69.36	65.12	85.91	91.01	92.23

Table 5.6: Person search evaluation on PoseTrack21. Approaches marked with [†] where re-implemented (Chen et al., 2020b).

scenarios, in which *FairMOT* significantly outperforms the other baselines.

Person Search: On our PoseTrack21-PersonSearch subset, we evaluate (Xiao et al., 2017b; Chen et al., 2020b,a; Li and Miao, 2021) as our state-of-the-art baselines. All baselines rely on Faster R-CNN (Ren et al., 2015) with a ResNet50 (He et al., 2016) backbone. Additionally, Faster R-CNN is extended by an additional re-identification head. For *OIM* (Xiao et al., 2017b), we use the re-implementation of (Chen et al., 2020b). Following (Chen et al., 2020b), we train *OIM* for 22 epochs on an image resolution of 900×1500 with a learning rate of $3e-3$, which is additionally decayed by a factor of 10 after 16 epochs.

NAE (Chen et al., 2020b) extends (Xiao et al., 2017b) with a norm-aware embedding head, which minimizes the embedding norm of background features towards 0 and maximizes the norm of person embeddings towards 1. We train *NAE* as proposed in (Chen et al., 2020b) with the same training protocol used for *OIM*. *NAE+* (Chen et al., 2020b) is a pixel-wise extension of *NAE*, which is initialized with a pre-trained *NAE* network. *NAE+* is further fine-tuned for 11 epochs with a learning rate of $3e-3$, which is decayed by a factor of 10 after 9 epochs. *HOIM* (Chen et al., 2020a) extends (Xiao et al., 2017b) with a different re-identification loss, which considers multiple background embeddings as additional negative examples in combination with an InfoNCE loss (Xiao et al., 2017b). We follow (Chen et al., 2020a) and train the network similar to (Xiao et al., 2017b). *SeqNet* (Li and Miao, 2021), on the other hand, proposed a cascaded architecture for Faster R-CNN (Ren et al., 2015) (cf. Section 3.5). In particular, *SeqNet* consists of a second bounding box regression head which refines the predicted bounding boxes of the first stage. Further, the second stage comprises a norm-aware re-identification head from (Chen et al., 2020b). We train *SeqNet* for 20 epochs with a learning rate of $3e-3$, which is decayed by a factor of 10 after 16 epochs.

Due to a stronger person detection model, *SeqNet* outperforms all other baselines as shown in Table 5.6. Surprisingly, *OIM* outperforms the remaining baselines. We argue that the corresponding baselines are highly optimized on common person search datasets such as PRW (Zheng et al., 2017a) or CUHK-SYSU (Xiao et al., 2017a) in terms of hyperparameters, which we adopted from the respective datasets.

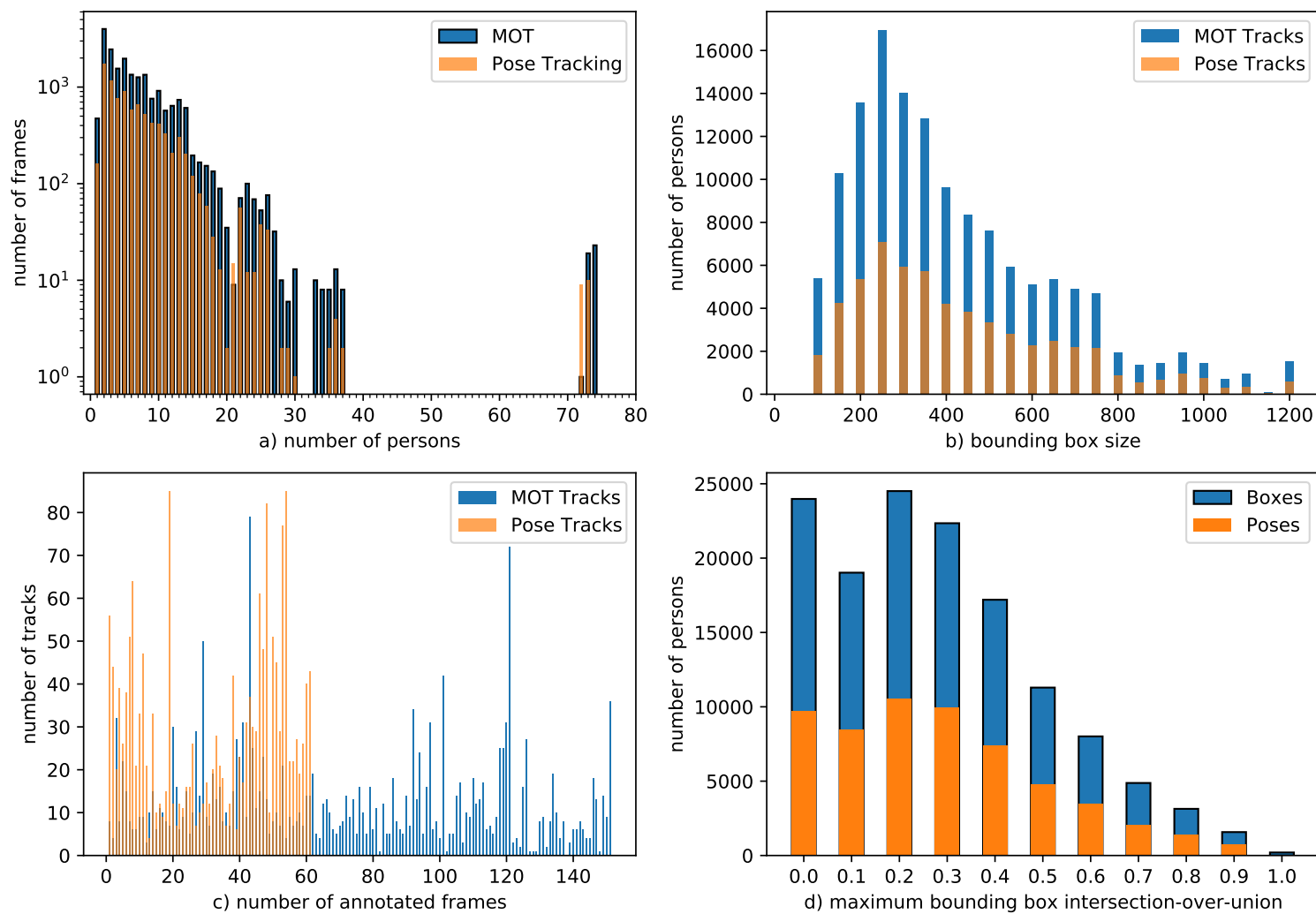


Figure 5.5: Statistics for the validation set of PoseTrack21.

5.6 Dataset Statistics

We visualize additional statistics for the validation set of PoseTrack21 in Figure 5.5. In particular, we illustrate the distribution of multi-object tracking (MOT) and multi-person pose tracking (Pose Tracking) instances within each frame in Figure 5.5 a). Some sequences in PoseTrack21 contain more annotated bounding boxes than poses. For that reason, there are some cases in which the number of frames containing n persons (e.g., $n = 72$) is higher for Pose Tracking than for MOT. For illustrative reasons, Figure 5.5 a) is scaled logarithmically. Additionally, we visualize the distribution of bounding box sizes in Figure 5.5 b). In Figure 5.5 c), we measure the number of annotated frames for MOT and pose tracking, respectively. PoseTrack21 follows PoseTrack 2018 (Andriluka et al., 2018), which provides densely annotated keypoints in the center of every sequence. The remaining frames are sparsely annotated with a step size of four frames. In contrast, PoseTrack21 provides annotated bounding boxes for every frame. Finally, we illustrate the total amount of persons for MOT and Pose Tracking, which have a maximum bounding box intersection over union in an interval of $[0, 1]$ in Figure 5.5 d). This shows that the majority of persons occlude each other with an intersection-over-union of at least 20%.

5.7 Ablation Studies

We provide additional ablative analysis for multi-person pose tracking, multi-object tracking, and person search.

Multi-Person Re-ID Pose Tracking: We additionally analyzed the performance of our proposed baselines in terms of false-positives and MOTA scores with respect to the size of the person bounding boxes. Figure 5.6a a) (page 104) shows the false positives rate for our baselines. As expected, the number of false positives decreases for larger bounding boxes. *TrackerWCorr* has significantly more false positives than the remaining baselines. This confirms that keypoint correspondences are not a suitable motion model and that keypoint-based non-maximum suppression fails to remove duplicate detections reliably.

As highlighted in Figure 5.6a b), all baselines achieve the highest MOTA scores for bounding box sizes between 800 and 1000 pixels, afterwards the performance drops significantly. The drop in performance might be caused by a weakness of our underlying person detector, which we used in all our Pose Tracking baselines. In particular, we used Faster R-CNN (Ren et al., 2015) with FPN (Lin et al., 2017a) as our person detector.

Multi-Object Tracking: In a similar fashion, we analyzed the performance of our multi-object tracking baselines (Bergmann et al., 2019; Zhang et al., 2021; Wang et al., 2020d) and *CorrTrack-WReid*. As shown in Figure 5.6b, MOT approaches show very similar performance patterns compared to multi-person pose tracking baselines (Figure 5.6a).

Person Search: Figure 5.7 (page 105) shows our ablative experiments for person search. In Figure 5.7 a-c), we plot the number of matches, mAP, and top-1 scores with respect to bounding box size. In Figure 5.7 d), we plot the number of matches with respect to bounding box visibility ($1 - \text{IoU}$).

Except for the top-1 score, the performance of all approaches degenerates for bounding box sizes larger than 1100 pixels, which is similar to multi-person pose tracking and multi-object tracking. We conclude that the respective person detector models are not sufficiently trained on large bounding boxes, which can be caused by the limited availability of large-scale persons during training. To our surprise, *SeqNet* (Li and Miao, 2021) performs worse than related approaches for box sizes between 1000 and 1200 pixels, even though *SeqNet* utilizes a cascaded detection architecture.

5.8 Discussion

In this chapter, we propose PoseTrack21, a joint dataset with annotated bounding boxes, human keypoints, and person-ids, suitable for the task of multi-person pose tracking, multi-object tracking, and person search. With this dataset, we want to encourage researchers to work on joint approaches, which can reliably solve multi-person pose tracking, multi-object tracking, and person search. As we have shown in our experiments, MOT approaches, in combination with a pose estimation model, can be utilized as reliable baselines for multi-person pose tracking. Vice versa, the pose estimation baselines perform fairly well in the context of MOT. Person search models, on the other hand, usually rely on an object detector with an extended re-identification head, which subsumes the exact baseline of modern MOT approaches. We believe that PoseTrack21 will help to address the discussed limitations and increase the synergy between the three related but separated research areas: multi-person pose tracking, multi-object tracking, and person search. Finally, it must be noted that the dataset will only be for research purposes, and it will be forbidden to use the dataset for training or evaluating commercial surveillance systems or other systems that can potentially harm societies or individuals.

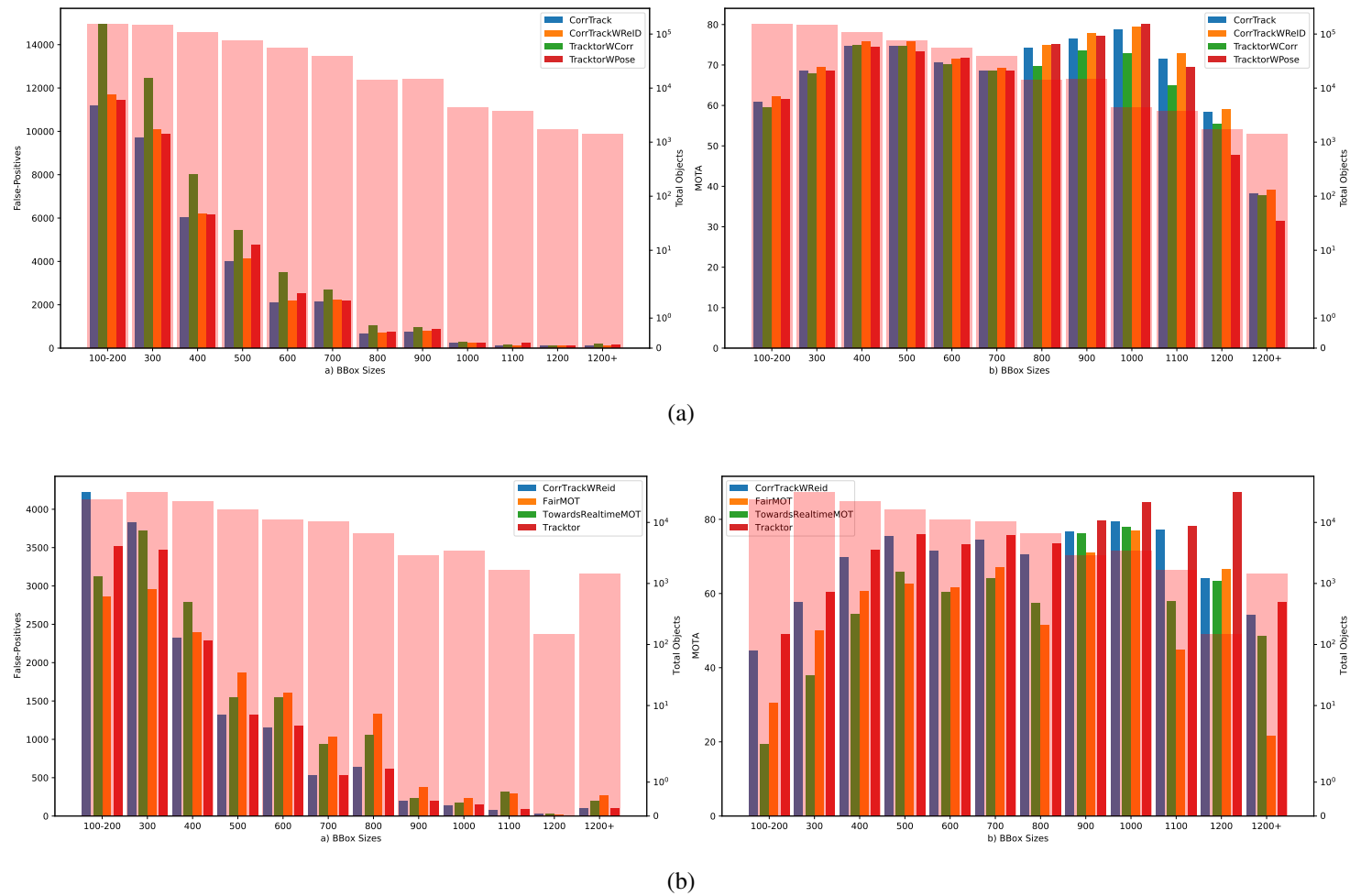


Figure 5.6: (a) Multi-person pose tracking: Ablative evaluation of false positives and MOTA with respect to bounding box size. **(b) Multi-object tracking:** Ablative evaluation of false positives and MOTA with respect to bounding box size. Best viewed with a PDF reader with zoom function.

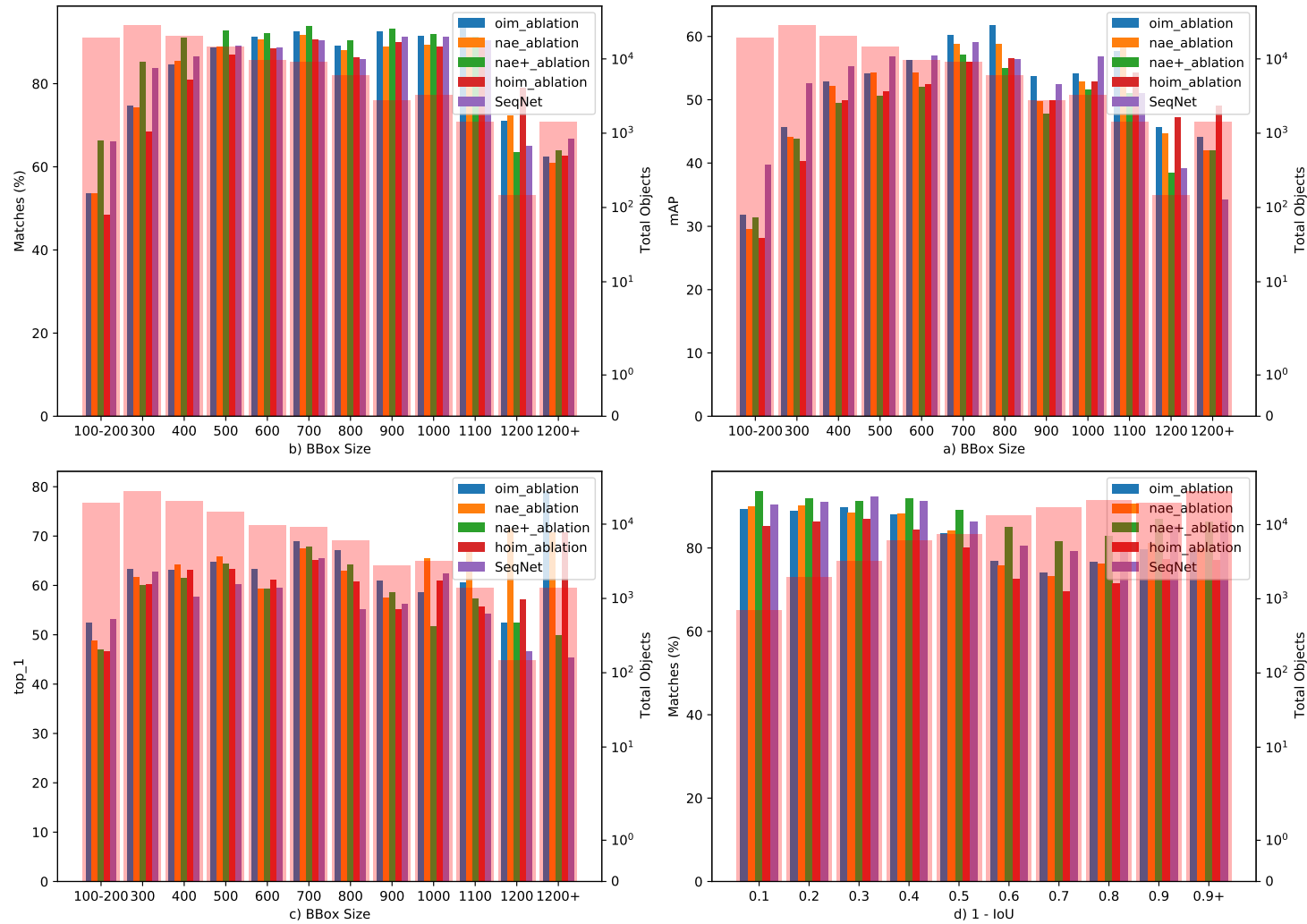


Figure 5.7: Person search: Ablative experiments with respect to bounding box size and visibility. Best viewed with a PDF reader with zoom function.

A Gated Attention Transformer for Multi-Person Pose Tracking

In the preceding chapters, we addressed key challenges in multi-person pose tracking. Previous datasets, such as PoseTrack 2018 (Andriluka et al., 2018), are sparsely annotated and lack person identity labels. To overcome these limitations, we introduced a self-supervised learning approach for keypoint correspondences in Chapter 4. Although this approach achieved state-of-the-art performance, errors in person detection still led to issues such as duplicate poses, particularly in crowded scenarios. These duplicates often share only a minimal subset of keypoints. Thus, these redundant poses are hard to remove and, consequently, initialize multiple tracks, which cause identity switches and diminishing tracking performance, as shown in Figure 4.10.

To address the shortcomings of earlier datasets, we presented PoseTrack21 in Chapter 5, a dataset with densely annotated poses and person identities. This provided a more reliable foundation for training and evaluating multi-person pose tracking systems. However, challenges such as handling occluded and unusual poses remained, as depicted in Figure 6.1.

Building on the progress from Chapters 4 and 5, this chapter introduces a Gated Attention Transformer framework for multi-person pose tracking in videos. This framework directly tackles the remaining challenges of occlusions, pose variations, and duplicate detections by integrating appearance embeddings and temporal pose similarities through a gating mechanism within attention layers. Furthermore, we employ a pose-conditioned re-identification network, leveraging the person identity annotations from PoseTrack21 to enhance track re-identification and reduce duplicate poses. This approach improves pose-to-track association, advancing the robustness of multi-person pose tracking.

Individual Contribution: The following chapter is based on the publication Doering and Gall (2023):

A Gated Attention Transformer for Multi-Person Pose Tracking

Andreas Doering and Juergen Gall

International Conference On Computer Vision Workshop (ICCV), 2023

This work was conducted by Andreas Doering under the scientific guidance of Juergen Gall.

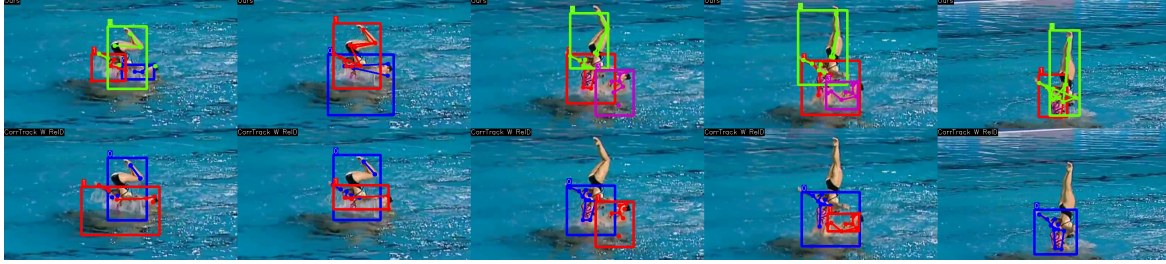


Figure 6.1: Qualitative examples of our proposed method on the PoseTrack21 dataset. The first row contains visual tracking results of our method and the second row shows visualizations of CorrTrack with ReID (Chapter 5).

Contents

6.1	Introduction	108
6.2	Gated Attention Transformer for Multi-Person Pose Tracking	109
6.2.1	Gated Attention Decoder	111
6.2.2	Confidence-Guided Track Update	113
6.2.3	Embeddings E_D^t and $E_{T \rightarrow D}$	114
6.2.4	Training Objective	115
6.3	Experiments	117
6.3.1	Datasets and Evaluation	117
6.3.2	Comparison with State of the Art	117
6.3.3	Implementation Details	119
6.3.4	Ablation Studies	120
6.3.5	Comparison to CORRTRACK + REID	124
6.4	Conclusion	124

6.1 Introduction

Multi-person pose tracking is highly relevant for a wide range of applications such as virtual reality, autonomous driving, or sports analysis. It requires accurately estimating and tracking the human poses of all persons throughout a video. Despite of the recent progress in multi-person pose tracking (Raaj et al., 2019; Doering et al., 2018; Hwang et al., 2019; Fu et al., 2021; Snower et al., 2019; Xiao et al., 2018; Rafi et al., 2020; Yang et al., 2021b; Wang et al., 2020c), the task remains very challenging due to camera motion, motion blur, occlusions, and a high variety in pose and scale (Chapter 5). Consequently, a tracking approach must be robust to detection errors and ambiguities. In particular, assigning highly occluded persons in unusual poses is very difficult, as shown in Fig. 6.1, where three persons perform a gymnastic exercise in water. For instance, the green bounding box intersects with three persons, and the respective pose overlaps with the keypoints of the other

persons. This poses a challenge, especially for online methods, as assignments will be made once a new frame arrives.

Related works such as (Snower et al., 2019; Rafi et al., 2020; Yang et al., 2021b; Xiao et al., 2018; Zhou et al., 2020a) try to tackle these challenges by generating future poses from a track’s history, which are then matched with detections based on pose similarities, e.g., based on Object Keypoint Similarity (OKS) (Rafi et al., 2020; Xiao et al., 2018; Zhou et al., 2020a) or a pose-based matching layer (Snower et al., 2019). Other works, such as (Wang et al., 2020c), process each sequence in an offline fashion, which is not feasible for real-time applications. As these works mainly rely on pose-based similarities for matching, these methods tend to fail to re-identify tracks that have been occluded for longer periods of time or undergo high pose deformations.

In this chapter, we thus focus on learning the association of detected persons to tracks in an online fashion and propose an approach that leverages the estimated poses, bounding boxes, and the appearance of the detected persons to assign them to previous tracks or initialize new tracks. Since we can neither rely solely on appearance-based features nor pose similarities due to multiple instances with similar appearance, changing camera views, or scene switches, often occurring in in-the-wild sequences, we introduce two types of embeddings. The detection and track embeddings are based on appearance and are used to measure the appearance similarity between detections and previous tracks. The additional edge embeddings directly encode the pose similarity between a pose and a track based on estimated poses and bounding boxes. While the pose similarity is a strong prior for tracking, it fails in case of fast motion or if the person disappears for some frames due to occlusion or being outside the camera view. We thus propose a gated attention transformer that combines and weights the attention matrices of both embedding types. A gated attention decoder updates all three embeddings, and a final matching layer assigns the detections to the tracks. The matching layer also removes duplicates, i.e., multiple detections for the same persons, and initiates new tracks. Furthermore, we employ a pose-conditioned re-identification model where the appearance features are normalized based on the heatmaps of the detected keypoints.

We evaluate our approach on the challenging PoseTrack 2018 (Andriluka et al., 2018) and PoseTrack21 (Chapter 5) datasets where it achieves state-of-the-art results. In summary, we propose i) a gated attention transformer that combines pose and appearance similarity in a novel way for pose-to-track association and ii) a novel matching layer for pose-to-track association and duplicate removal.

6.2 Gated Attention Transformer for Multi-Person Pose Tracking

Online methods for multi-person pose tracking often follow the tracking-by-detection paradigm (Snower et al., 2019; Girdhar et al., 2018; Doering et al., 2018; Raaj et al., 2019; Hwang et al., 2019; Xiao et al., 2018; Zhang et al., 2019; Rafi et al., 2020) and usually suffer from ambiguities and occlusions. An example is shown in the second row of Figure 6.1 where the blue id jumps between two persons. To make the matching between detections and previously tracked persons more robust, we propose a gated attention transformer that directly learns the matching by a gated matching layer. It combines appearance features and encoded temporal person similarities. Since the importance of appearance and pose similarity varies within a video and between videos, in particular when a person has been occluded for a few frames, the gated attention decoder and the matching layer use a gating mechanism to update the embeddings of the new poses and previous tracks and to match poses to

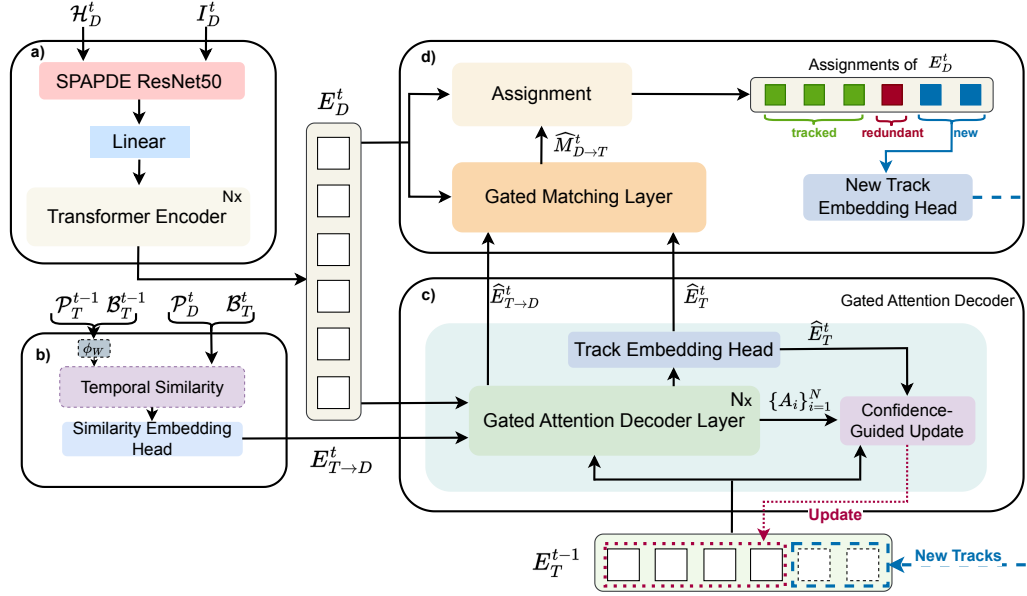


Figure 6.2: The proposed multi-person pose tracking architecture entails the following steps: a) Given a set of person crops and their respective keypoint heatmaps at time frame t , we compute pose-conditioned appearance features and feed them into N transformer encoder stages. This gives an embedding for each detection E_D^t , which will be used to measure the similarity to previous tracks. b) We also compute spatial similarities between tracks and detected persons by means of Intersection over Union (IoU) and Object Keypoint Similarity (OKS), which we then encode into a pose similarity embedding $E_{T \rightarrow D}^t$. This serves as a strong prior for matching. c) The gated attention decoder takes the embeddings of the previous tracks E_T^{t-1} , the detection embeddings E_D^t and the pose similarity embeddings $E_{T \rightarrow D}^t$ as input and updates the embeddings. It adaptively weights the spatial prior $E_{T \rightarrow D}^t$ and the appearance similarity between E_T^{t-1} and E_D^t . d) Finally, the matching stage assigns detections to tracks, removes redundant detections, and initializes new tracks.

tracks. For example, if there are several persons who look very similar in a frame, the pose similarity can guide the update of the appearance embedding. Vice versa, the update will be driven by the appearance embeddings if there is no spatial proximity between tracks and detections. Our approach for multi-person pose tracking is illustrated in Figure 6.2.

We assume that the human poses are extracted for a new frame t by a standard multi-person pose estimator where we utilize the detector from Chapter 5 for a fair comparison. Specifically, for a given frame at time t , our network takes as input the set of estimated bounding boxes B_D^t and poses P_D^t , and additionally detected keypoint heatmaps H_D^t extracted from the image crops I_D^t of detected persons. From the heatmaps and image crops, an embedding E_D^t for re-identification is computed as described in Section 6.2.3 (Figure 6.2a). To measure pose and spatial proximity between detections and tracks, similarities to the bounding boxes B_T^{t-1} and poses P_T^{t-1} of the last frame of each track are computed, which results in the edge embedding $E_{T \rightarrow D}^t$ (Figure 6.2b) between a track T and a detection D , which will be described in Section 6.2.3 as well. We denote $E_{T \rightarrow D}^t$ as pose similarity embedding. Given both embeddings E_D^t and $E_{T \rightarrow D}^t$ as well as the track embeddings E_T^{t-1} that have

been estimated in the previous frame $t-1$, the proposed *Gated Attention Decoder* (Figure 6.2c) and the *Gated Matching Layer* (Figure 6.2d) assign the current detections to previous tracks, update the track embeddings, and initialize new tracks. Both will be described in the following Section 6.2.1.

6.2.1 Gated Attention Decoder

Attention Layer: As the baseline method, we employ a transformer decoder as proposed in Vaswani et al. (2017), and we propose an attention-based matching layer as the main extension. As illustrated in Figure 6.3a, the attention layer calculates the similarities $A = \sigma(O_A)$ between the appearance features of the detections E_D^t and tracks E_T^{t-1} , where σ is the row-wise softmax layer. O_A represents the appearance-based similarity logits that are obtained by cross-attention between the appearance features of the detections E_D^t and tracks E_T^{t-1} :

$$O_A = \frac{E_T^{t-1} W_Q^\top (E_D^t W_K^\top)^\top}{\sqrt{d}}, \quad (6.1)$$

where W_Q and W_K are the learned projection weights for the queries and keys, respectively, and d is the dimensionality of the embeddings E_* .

To assign detections to tracks based on appearance similarity, we need to allow that none of the detections is assigned to a track, e.g., if a person has not been detected or is occluded. Prior to applying the softmax to the attention logits, i.e., $A = \sigma(O_A)$, we thus add a column of zeros. In other words, the last column of A indicates if a track does not match with any detection.

Finally, the attention layer calculates the proposed track embedding update as follows:

$$\Delta E_T^t = (A_{:, :-1} E_D^t) W_A^\top, \quad (6.2)$$

where $A_{:, :-1}$ denotes the attention weights without the last column and W_A are the weights of a linear layer.

Gated Attention Layer: Appearance similarities allow re-identifying an occluded person after some frames but are unreliable in case of motion blur or person instances with a similar appearance as common in team sports videos. Pose and spatial similarities, however, provide a strong matching prior but are less reliable in crowded scenarios as shown in Figure 6.1. While none can resolve all ambiguities, fusing the similarities automatically provides a stronger matching prior. Based on these intuitions, we propose the gated attention layer. As illustrated in Figure 6.3d, the gated attention layer extends Figure 6.3a and incorporates pose similarity weights $S_E = \sigma(O_E)$ between detections and tracks. The pose similarity logits $O_E = E_{T \rightarrow D}^t W_E^\top$ are obtained by $E_{T \rightarrow D}^t$ and the learned weight matrix W_E .

To assign detections to tracks, appearance-based similarities S_A and pose-based similarities S_E are fused by the α -Gate that weights the contribution of appearance-based attention weights and the pose-based attention weights by a hyperparameter α , which we evaluate in our experiments:

$$A = \alpha \cdot S_A + (1 - \alpha) \cdot S_E. \quad (6.3)$$

Similar to the attention layer (Figure 6.3a), we add a column of zeros to S_A and S_E and obtain ΔE_T^t

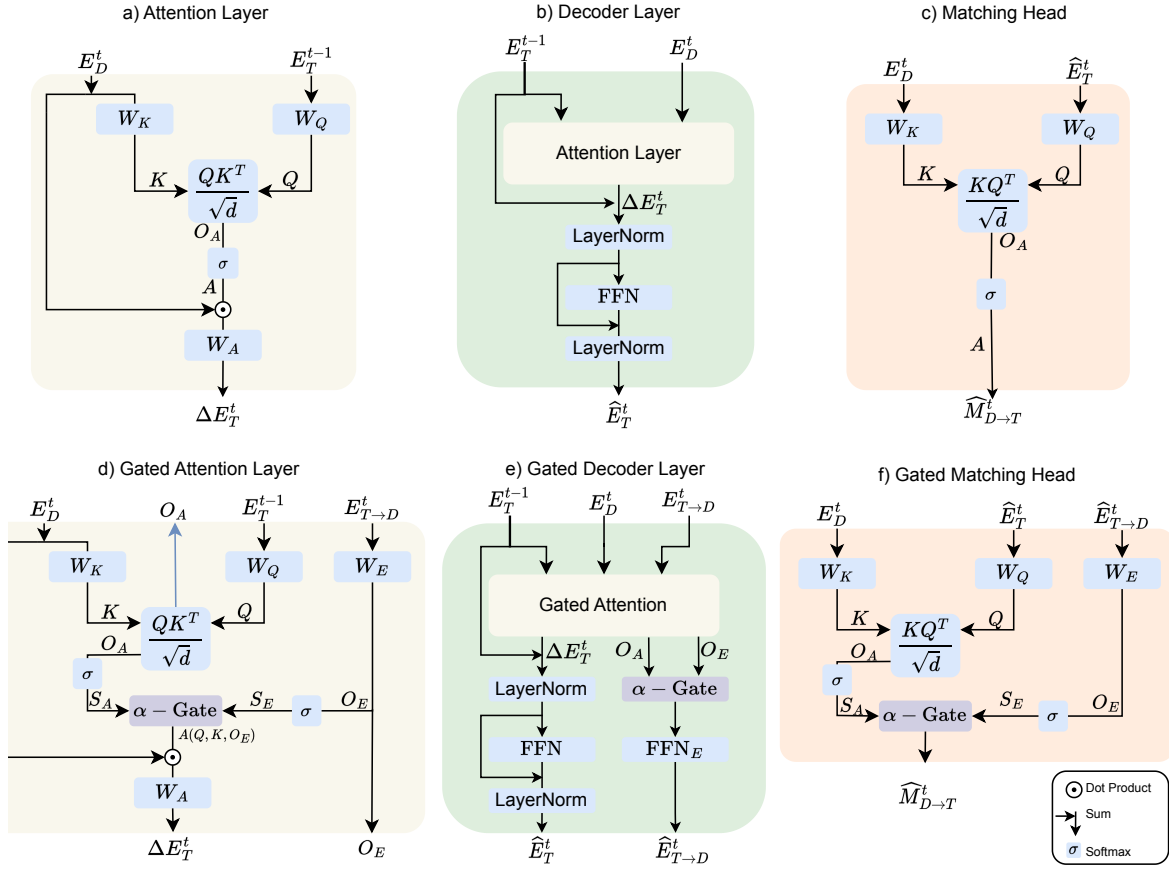


Figure 6.3: Illustration of our proposed d) Gated Attention layer, e) Gated Decoder layer, and f) Gated Matching layer. The top row shows a vanilla implementation without gating for each layer: a) Standard Attention layer with E_D^t as keys and E_T^{t-1} as queries. b) The Decoder layer takes as input the detection embeddings E_D^t and track embeddings E_T^{t-1} and updates the track embeddings $\hat{E}_T^t$ using the Attention layer. c) The Matching layer predicts an assignment matrix $\hat{M}_{D \rightarrow T}^t$ of detections to tracks. The proposed gated layers use the additional pose similarity embeddings $E_{T \rightarrow D}^t$ as strong prior. The α -Gates fuse the cross-attention matrix S_A between detections and tracks, which measures appearance similarity, and the pose similarity matrix S_E , which measures spatial and pose similarity between detections and tracks.

following (6.2). Intuitively, fusing the normalized similarities S_A and S_E automatically assigns a higher weight to the similarity measure where the matching confidence of a detection to a track is higher, resulting in a higher tracking accuracy, as we will show in the experiments.

Decoder Layer: The decoder layer shown in Figure 6.3b then updates the track E_T^{t-1} embeddings based on the output of the attention layer. As common for transformer blocks [Vaswani et al. \(2017\)](#),

we use a residual feed-forward network (FFN) as shown in Figure 6.3b. Specifically, we compute

$$\hat{E}_T^t = \text{LN}(\tilde{E}_T^t + \text{FFN}(\tilde{E}_T^t)), \quad \tilde{E}_T^t = \text{LN}(E_T^{t-1} + \Delta E_T^t), \quad (6.4)$$

where LN denotes layer normalization.

Gated Decoder Layer: The gated decoder layer as shown in Figure 6.3e) additionally employs an α -Gate to weight the contribution of appearance-based and pose similarity-based attention logits O_A and O_E similar to the gated attention layer. We then apply a feed-forward network FFN_E on the weighted sum of attention logits

$$\hat{E}_{T \rightarrow D}^t = \text{FFN}_E(\alpha \cdot O_A + (1 - \alpha) \cdot O_E), \quad (6.5)$$

to update the pose similarity embeddings, where FFN_E is a feed-forward network as FFN . Empirically, the gated decoder layer performs best if α -gating is performed on the attention logits, as we show in the experiments.

Gated Matching Layer: The matching layer (Figure 6.3c) and the gated matching layer (Figure 6.3f) comprise a structure similar to the attention layer and the gated attention layer, respectively, but differ in two aspects. 1) Both matching layers (Figure 6.3c) and (Figure 6.3f) do not use the attention weights to predict a track embedding update and therefore do not consist of a linear layer after the softmax and α -Gate, respectively. 2) The attention weights A (i.e., (6.3)) are used as assignment matrix $\hat{M}_{D \rightarrow T}^t$ to assign detections to tracks, as we show in Figure 6.2d. In the presence of duplicate detections, this allows the matching of multiple detections to a single track. Given the matching matrix $\hat{M}_{D \rightarrow T}^t$, we utilize Hungarian matching to assign detections to tracks. All the remaining detections i that have a matching probability $\hat{m}_{i \rightarrow j} > \tau_{dup}$ for any track j are considered duplicate detections and are removed. Any other detection will initialize a new track embedding (Figure 6.2d). All tracks not tracked for τ_{age} frames will be removed. In the experiments, we evaluate the impact of τ_{dup} and τ_{age} .

As shown in Figure 6.2, we employ an additional *Track Embedding Head* after the last decoder layer that predicts the final track embeddings and *New Track Embedding Head* that generates the embedding for newly initialized tracks. Both heads are illustrated in Figure 6.4a) and consist of two linear layers, where the first layer includes a LayerNorm (Ba et al., 2016) and GELU (Hendrycks and Gimpel, 2016).

6.2.2 Confidence-Guided Track Update

As shown in Figure 6.2c, we perform the final update of the tracks embedding before the final assignment of detections to tracks is carried out by the matching layer (Figure 6.2d). This is done for two reasons. Firstly, the final assignment might be wrong, and secondly, we do not want to update the track embedding when the detection embedding is noisy due to occlusion or motion blur. We employ a confidence-guided update of the track embeddings to prevent being updated by noisy or low-confident detections. Let $\mathcal{A} = \{A_{:,n-1}^n\}_{n=1}^N$ be the set of gated attention weights of all N decoder stages without the last column. As the number of detections per frame is dynamic, we apply max

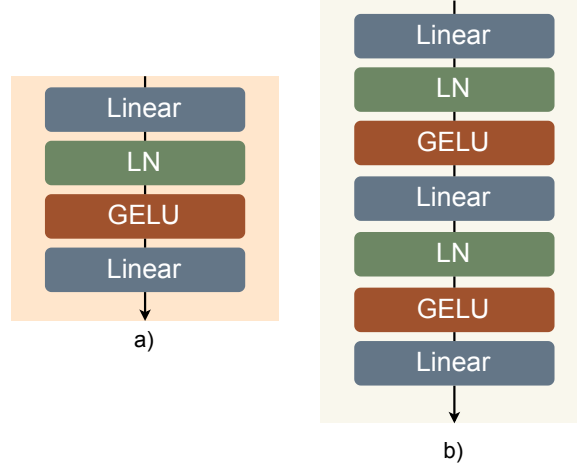


Figure 6.4: Illustration of the a) track embedding head and b) temporal pose similarity embedding head.

pooling on the rows of each attention matrix to obtain the maximum attention score for each track and estimate an importance weight w_j with a linear layer as

$$\hat{\mathcal{A}}_j = \text{concat}\{\max_i A_{ji}^n | n \in [1, N]\} \quad (6.6)$$

$$w_j = \sigma \left(\sum_n w_n \cdot \hat{\mathcal{A}}_j^n + b_n \right), \quad (6.7)$$

where σ is the sigmoid function and $\hat{\mathcal{A}}_j^n$ the maximum attention for track j at layer n . Ultimately, we update the embedding for each track j as a confidence-guided moving average by using the importance weight w_j following $E_{T_j}^t = (1 - w_j) \cdot E_{T_j}^{t-1} + w_j \cdot \hat{E}_{T_j}^t$. As we will show in our experiments, the confidence-guided embedding update improves the performance compared to solely relying on the *Track Embedding Head* (Figure 6.2c) while only adding a negligible overhead.

6.2.3 Embeddings E_D^t and $E_{T \rightarrow D}$

We finally describe how the detection embeddings E_D^t (Figure 6.2a) and pose similarity embeddings $E_{T \rightarrow D}$ (Figure 6.2b) are computed.

Detection Embeddings E_D^t : The embeddings we extract from each newly detected person must be robust to occlusion. Since the keypoint heatmaps $\mathcal{H}_D^t$ of the pose estimator provide an indicator of which keypoints are occluded, we condition the embedding E_D^t on $\mathcal{H}_D^t$. Specifically, we use a ResNet50 (He et al., 2016) and propose Spatially Adaptive Pose DEnormalization (SPAPDE) layers, which are inspired by Park et al. (2019), as an extension. Each SPAPDE layer operates on a set of keypoint heatmaps $\mathcal{H}_D^t \in \mathbb{R}^{N \times K \times \frac{256}{s} \times \frac{128}{s}}$ along with the current ResNet features $f \in \mathbb{R}^{N \times C \times \frac{256}{s} \times \frac{128}{s}}$ extracted from the corresponding image crops $\mathcal{I}_D^t \in \mathbb{R}^{N \times 3 \times 256 \times 128}$, where N is the number of persons in frame t , s denotes the respective scaling factor, K denotes the number of keypoints and C

denotes the number of feature channels.

In contrast to batch normalization (Ioffe and Szegedy, 2015), the normalized input features are scaled and shifted with respect to the keypoint heatmaps. In particular, SPAPDE computes the modulation parameters β and γ following

$$\gamma = \text{conv}(a), \quad \beta = \text{conv}(a), \quad a = \text{ReLU}(\text{conv}(\mathcal{H}_D^t)), \quad (6.8)$$

where conv denotes a 3×3 -convolution and $\beta, \gamma \in \mathbb{R}^{N \times C \times \frac{256}{s} \times \frac{128}{s}}$. The image features f are then conditioned on the heatmaps $\mathcal{H}_D^t$ as follows. Let $f_{n,c,y,x}$ be the feature value for the detected person n and feature channel c at pixel location (x, y) . The SPADPE layer first calculates the mean μ_c and standard deviation σ_c over all persons and pixels of f and then adaptively de-normalizes the image features by

$$\hat{f}_{n,c,y,x} = \gamma_{n,c,y,x} \cdot \frac{f_{n,c,y,x} - \mu_c}{\sigma_c} + \beta_{n,c,y,x}. \quad (6.9)$$

We replace every batch normalization layer within ResNet50 with SPAPDE layers and train the network following Luo et al. (2019b).

Moreover, we use a vanilla N-stage transformer encoder (Vaswani et al., 2017) without positional encoding to further disentangle the backbone features of severely occluded persons and noisy pose predictions and generate a set of encoded person features $E_D^T \in \mathcal{R}^{N \times 256}$ as shown in Figure 6.2a.

Pose Similarity Embeddings $E_{T \rightarrow D}$: The pose similarity features $E_{T \rightarrow D}^t$ (Figure 6.2b) used in our gated attention architecture are based on similarities between bounding boxes and poses using Intersection over Union (IoU) and Object Keypoint Similarity (OKS), respectively. In this chapter, we rely on three variants of OKS: the first variant considers keypoints that are present in both poses. While the first variant provides a good measure of keypoint alignment, its expressiveness suffers if two poses only share a small subset of keypoints. Therefore, the remaining two variants consider all keypoints present in one of the two poses respectively. To deal with motion, we use a warping function ϕ_W as in Chapter 4 that warps the last observed pose of all tracks into the current frame t . The pose similarity $E_{T \rightarrow D}$ between tracks and detections is then computed as follows:

$$E_{T \rightarrow D} = \phi_E \left(\left[\text{IOU}(\hat{\mathcal{B}}_T^{t-1}, \mathcal{B}_D^t) \parallel \text{OKS}(\hat{\mathcal{P}}_T^{t-1}, \mathcal{P}_D^t) \right] \right). \quad (6.10)$$

Here, $\hat{\mathcal{P}}_T^{t-1} = \phi_W(\mathcal{P}_T^{t-1})$ and $\hat{\mathcal{B}}_T^{t-1} = \phi_W(\mathcal{B}_T^{t-1})$ represent the set of warped track poses and track bounding boxes, respectively. The operator $[\cdot \parallel \cdot]$ represents concatenation, and ϕ_E denotes the pose similarity embedding head (Figure 6.4b), which consists of three linear layers with LayerNorm (Ba et al., 2016) and GELU (Hendrycks and Gimpel, 2016).

6.2.4 Training Objective

In the first step, we train the re-identification network following (Luo et al., 2019b): We apply the triplet loss (Hermans et al., 2017) and center loss (Wei et al., 2017) after the last pooling layer of ResNet50 and we employ the cross-entropy loss with label smoothing (Szegedy et al., 2016) on the classification layer. Subsequently, we freeze the re-identification network and proceed to train our network.

Approach	Online	AssA	FragA	DetA	HOTA	MOTA	mAP
CorrTrack (Chapter 5)	✓	58.02	57.75	45.48	51.13	63.0	72.3
CorrTrack w. ReID (Chapter 5)	✓	60.21	59.66	46.56	52.71	63.8	72.7
Tracktor++ w. Poses (Chapter 5)	✓	59.41	58.61	46.30	52.21	63.3	71.4
Tracktor++ w. Corr. (Chapter 5)	✓	54.05	52.02	44.67	48.90	61.6	73.6
Ours w/o gating	✓	44.92	41.96	45.30	44.82	52.4	70.2
Ours w. gating	✓	62.20	60.93	47.20	53.94	64.1	73.6
CorrTrack (Chapter 5)	✗	60.93	60.37	45.48	52.42	63.9	72.3

Table 6.1: Comparison to multi-person pose tracking methods on the PoseTrack21 dataset.

In our approach, the matching layer is trained using a cross-entropy loss, which is defined as follows:

$$\mathcal{L}_{match} = -\frac{1}{N_D} \sum_i y_i \cdot \log(p_{ij}^m) + (1 - y_i) \cdot p_{i0}^m, \quad (6.11)$$

where N_D is the total number of detections, $p_{i,j}^m$ represents the probability of matching the i -th detection to its corresponding ground truth track j , and p_{i0}^m is the probability of not matching the i -th detection to any track. The variable y_i takes the value 1 if the i -th detection is assigned to a ground truth track and 0 otherwise.

Since we discard duplicates after the final matching layer, we generate duplicates during training and allow multiple detection assignments to a single track. Specifically, we use detected poses and ground truth poses in the training process that can share the same person identity. We employ OKS-based greedy matching to the ground truth poses to assign identities to the detected poses during training. We then utilize a duplicate-aware cross-entropy loss function that operates on the attention weights of the encoder and decoder layers. The loss function is defined as follows:

$$\mathcal{L}_{attn} = -\frac{1}{N_T} \sum_j \log(p_j), \quad p_j = \left(\sum_i A_{ji} \mathbb{I}_i(j) \right) + \mathbb{I}_{\#i}(j) A_{j0}. \quad (6.12)$$

N_T represents the total number of tracks and p_j denotes the accumulated matching probability for track j , where A_{ji} is the attention weight of the respective encoder/decoder layer and $\mathbb{I}_i(j)$ is 1 if the identity of the current track j and the detection i are the same, and 0 otherwise. If none of the detections matches, i.e., $\mathbb{I}_{\#i}(j)$, we maximize the no-match probability A_{j0} , which is the last column of the attention matrix as discussed in Section 6.2.1. In other words, we want that A_{ji} is large for the correct assignment if and only if a match exists.

The final objective function is a combination of the cross-entropy loss function for the matching layer ($\mathcal{L}_{match}$) and the duplicate-aware cross-entropy loss functions for each encoder and decoder layer:

$$\mathcal{L} = \mathcal{L}_{match} + \sum_k \mathcal{L}_{attn}^{enc_k} + \sum_k \mathcal{L}_{attn}^{dec_k}, \quad (6.13)$$

where $\mathcal{L}_{attn}^{enc_k}$ and $\mathcal{L}_{attn}^{dec_k}$ represent the duplicate-aware cross-entropy loss functions for the k -th encoder and decoder layer, respectively.

Method	AssA	DetA	LocA	HOTA	MOTA
TRMOT (Wang et al., 2020d)	54.98	40.91	79.92	46.85	47.2
FairMOT (Zhang et al., 2021)	61.45	47.43	83.16	53.53	56.3
Tracktor++ (Bergmann et al., 2019)	65.43	52.71	83.09	58.29	59.5
CorrTrack + ReID (Chapter 5)	64.19	51.33	82.80	56.95	52.0
Ours	66.89	51.81	82.71	58.42	55.3

Table 6.2: Comparison to MOT methods on the PoseTrack21-MOT dataset.

6.3 Experiments

6.3.1 Datasets and Evaluation

We evaluate our work on the PoseTrack datasets (Andriluka et al. (2018) and Chapter 5). Both datasets are large-scale benchmarks for multi-person pose tracking and contain 593 videos for training and 170 for evaluation. The videos contain various activities and include highly diverse poses and severe occlusions as shown in Figure 6.1. Since for PoseTrack 2018, the evaluation server is no longer available, we only report results on the validation set. Compared to PoseTrack 2018, PoseTrack21 (Chapter 5) provides more annotations and additional benchmarks for multi-object tracking (MOT) and person search. We thus primarily focus on PoseTrack21 in our experiments. We use *keypoint HOTA* (Chapter 5) for evaluation. Keypoint HOTA consists of sub-metrics that measure the *detection accuracy* (DetA), the *association accuracy* (AssA), and the *fragmentation accuracy* (FragA). In addition, we report results for the *keypoint-based MOTA* metric (Andriluka et al., 2018). Both metrics are evaluated on a keypoint level and then averaged. We report the keypoint detection performances in terms of *mean average precision* (mAP) for completeness.

We follow common practice (Snower et al., 2019; Xiao et al., 2018; Rafi et al., 2020; Yang et al., 2021b; Doering et al., 2022; Wang et al., 2020c) and utilize a multi-frame pose estimation approach to compensate for missed detections due to motion blur and occlusions during inference. In particular, we use keypoint correspondences as in Chapters 4 and 5. In the following, we compare our approach to the state of the art.

6.3.2 Comparison with State of the Art

PoseTrack21: We first evaluate our model with and without gating on the PoseTrack21 validation set and compare the performance to methods proposed in Chapter 5 using the keypoint HOTA (Chapter 5) and the MOTA metrics (Andriluka et al., 2018). The results are shown in Table 6.1. While the performance *without* gated attention is quite low, our proposed gated attention transformer consistently outperforms existing methods, achieving a HOTA score of 53.94 and a MOTA score of 64.1. Compared to CORRTRACK W. REID, our approach boosts the association accuracy (AssA) and fragmentation accuracy (FragA) by +1.99% and +1.27% to 62.20 and 60.93, respectively. Additionally, the detection accuracy (DetA) and the mAP increase to 47.20 and 73.6, respectively. While our approach performs online multi-person pose tracking, it outperforms the offline approach CorrTrack.

We further evaluate our approach on the PoseTrack21-MOT benchmark and compare the performance to the methods in Chapter 5. As we show in Table 6.2, our approach consistently outperforms

Approach	Online	Val. Set	Detector	MOTA	mAP
STAF (Raaj et al., 2019)	✓	v1	-	60.9	70.4
T CPN++ (Yu et al., 2018)	✓	v1	Cascade R-CNN (Cai and Vasconcelos, 2017)	64.0	80.9
MIPAL (Hwang et al., 2019)	✓	v1	-	65.7	74.6
KeyTrack (Snower et al., 2019)	✓	v1	HTC (Chen et al., 2019a)	66.6	81.6
CorrTrack (Chapter 4)	✓	v1	Cascade R-CNN (Cai and Vasconcelos, 2017)	68.8	79.2
TKMRNet (Zhou et al., 2020a)	✓	v1	Faster R-CNN FPN DCN (Zhu et al., 2019)	68.9	76.7
CorrTrack (Chapter 4)	✗	v1	Cascade R-CNN (Cai and Vasconcelos, 2017)	69.1	79.2
CorrTrack (Chapter 4)	✓	v2	Cascade R-CNN (Cai and Vasconcelos, 2017)	63.6	75.9
LTIVA (Fu et al., 2021)	✓	v2	-	64.7	71.4
CombDet (Wang et al., 2020c)	✗	v2	ResNet-101 SNIPER (Singh et al., 2018)	68.7	81.5
LDGNN (Yang et al., 2021b)	✓	v2	Faster R-CNN FPN DCN (Zhu et al., 2018)	69.2	77.9
Ours	✓	v2	Cascade R-CNN (Cai and Vasconcelos, 2017)	64.5	76.4

Table 6.3: Comparison to the state of the art on PoseTrack 2018 Andriluka et al. (2018). Two versions of the validation set have been released containing 74 (v1) and 170 (v2) sequences, respectively.

existing methods in terms of AssA (66.89) and HOTA (58.42). Tracktor++ (Bergmann et al., 2019) achieves a slightly higher DetA and localization accuracy (LocA), resulting in higher MOTA. While Tracktor++ has been trained on the annotated bounding boxes for MOT, our approach has been trained for pose tracking, and we simply generate the bounding boxes from the estimated poses. Consequently, MOT methods achieve a better MOTA score due to higher bounding box detection and localization accuracy. As discussed in Luiten et al. (2020), HOTA is a better metric than MOTA for MOT.

PoseTrack 2018: The comparison with related works on PoseTrack 2018 is difficult for two reasons: i) The PoseTrack 2018 dataset is not available anymore, and it is no longer possible to submit results to the official test server; ii) the validation set was released in two different versions. The first version (v1) contains 74 whereas the second version (v2) contains 170 sequences, respectively. For completeness, we also include results reported for v1 in Table 6.3, but these numbers are not comparable. The results for CorrTrack (Chapter 4) show that version v2 is much more difficult. Compared to CorrTrack, our method improves the MOTA score by 0.9% to 64.5. The pose estimation performance increases from 75.9 to 76.4 in terms of mAP. Our approach performs similarly to Fu et al. (2021) in terms of MOTA while achieving a higher mAP. CombDet (Wang et al., 2020c) achieves higher accuracy, but it uses a stronger multi-frame person detector and is an offline approach, whereas our approach is an online approach. LDGNN (Yang et al., 2021b) also uses a better multi-frame pose estimator, but the code of the pose estimator is not publicly available.

Market1501 For a comparison with related works on person re-identification, we train our proposed person re-identification model SPADPE (Section 6.2.3) on the Market1501 dataset (Zheng et al., 2015). Market1501 is a small dataset for person re-identification and contains 32,688 annotated bounding boxes with a total of 1501 identities. The training and testing sets consist of 751 and 750 identities, respectively. As this dataset does not provide annotated keypoints, we utilize our pose estimator trained on PoseTrack21 to estimate the human pose of all person crops of Market1501. To evaluate the person re-identification model, we follow Zheng et al. (2015) and report the mean

Approach	r = 1	mAP
PSE (Sarfraz et al., 2018)	87.7	69.0
GLAD (Wei et al., 2017)	89.9	73.9
PGFA (Miao et al., 2019)	91.2	76.8
PCB (Sun et al., 2018c)	92.3	77.4
OAMN (Chen et al., 2021b)	92.3	79.8
HOReID (Wang et al., 2020b)	94.2	84.9
FED (Wang et al., 2022d)	95.0	86.3
BPBreID (Somers et al., 2023)	95.1	87.0
PAT (Li et al., 2021d)	95.4	88.0
NGFR (Yu et al., 2021)	95.5	88.1
TransReID (He et al., 2021)	95.0	88.2
RFCNet (Hou et al., 2020b)	95.2	89.2
PFD (Wang et al., 2022b)	95.5	89.7
FPC (Ye et al., 2022)	95.1	91.4
DIP (Li et al., 2022a)	95.6	90.3
BoT (Luo et al., 2019b)	95.4	94.2
DC-Former (Li et al., 2023)	96.0	90.4
Ours	<u>95.8</u>	<u>92.8</u>

Table 6.4: Comparison to state-of-the-art methods on Market1501 (Zheng et al., 2015). The best results are **bold**, and the second best results are underlined.

average precision (mAP) and rank-1 metrics. Since Market1501 contains pedestrians with very limited pose variations, our approach is not expected to achieve state-of-the-art results on Market1501 Table 6.4.

6.3.3 Implementation Details

Person Detector: For a fair comparison to related works on PoseTrack21, we utilize the same person detector and pose estimation model for all our experiments. In particular, we use the person detector from Chapter 5, which consists of a Faster R-CNN (Ren et al., 2015) with a ResNet50-FPN (Lin et al., 2017a) backbone. The detector was first pre-trained on MSCOCO Lin et al. (2014) and further fine-tuned on PoseTrack21 for 30 more epochs. As pose estimator, we employ the model from Chapters 4 and 5. The pose estimation model was trained on MSCOCO and PoseTrack21 for 215 and 16 epochs, respectively. On PoseTrack 2018, we use Cascade R-CNN (Cai and Vasconcelos, 2017) as the object detector and the pose estimator from CorrTrack (Chapter 4).

Re-Identification Model: We train the re-identification model for 244 epochs on the PoseTrack21 person search dataset with a batch size of 256 and a learning rate of 0.00035 that we decay to 0.000035 after 75 epochs. During the first ten epochs, we apply a linear learning rate warm-up. Additionally, we apply data augmentation through random scaling, random rotation, and horizontal flipping.

$\mathcal{L}_{attn}^{dec}$	Encoder	$\mathcal{L}_{attn}^{enc}$	CG-Update	HOTA
				53.04
✓				53.40
✓	✓			53.56
✓	✓	✓		53.61
✓	✓	✓	✓	53.94

Table 6.5: Impact of several components in our pose tracking network on the tracking performance. CG-Update denotes the Confidence-Guided Track Update as discussed in Section 6.2.1.

α -Gate Source	Learnable α	AssA	FragA	DetA	HOTA
Attention Logits O_A & O_E		62.20	60.93	47.20	53.94
Attention Logits O_A & O_E	✓	61.80	60.59	47.15	53.74
Attention Weights S_A & S_E		54.74	50.77	47.14	50.53

Table 6.6: Impact of using attention weights or attention logits in the α -Gate (6.5) of the Gated Attention Transformer on the overall performance on PoseTrack21.

Gated Attention Transformer: Our proposed gated attention transformer employs two encoder and two decoder stages. On the PoseTrack21 dataset, we train our transformer for 14 epochs with a learning rate of 0.0001, decayed by a factor of 10 after 13 epochs. We further incorporate a linear learning rate warm-up over the course of 16k iterations and optimize the network with the AdamW (Loshchilov and Hutter, 2019) optimizer. Each training sequence is split into sub-sequences of length three with an overlap of one frame. We follow the same settings on the PoseTrack 2018 dataset and train the gated attention transformer for 11 epochs.

6.3.4 Ablation Studies

6.3.4.1 Evaluation of the Network Architecture

We perform ablation experiments to examine the influence of each building block of our proposed method. All experiments are conducted on the PoseTrack21 dataset.

Loss Terms, Transformer Encoder and Confidence-Guided Track Update: We first evaluate the impact of each component in our tracking model on the keypoint HOTA score (Chapter 5) in Table 6.5. We start with the gated attention decoder with a matching layer as our base model, which we trained with the matching loss $\mathcal{L}_{match}$ (6.11). Subsequently, we incrementally activate several components and evaluate their impact on the overall performance. Direct supervision on the gated attention layers using $\mathcal{L}_{attn}^{dec}$ (6.12) increases the overall performance from 53.04 to 53.40 in terms of HOTA. Using additional transformer encoder layers (Figure 6.2a) boosts the performance to 53.56. Using the loss $\mathcal{L}_{attn}^{enc}$ (6.12) also for the encoder increases HOTA to 53.61. The impact of the Confidence-Guided Track Update (Section 6.2.2) is shown in the last row. It further increases the tracking performance by 0.33 to a HOTA score of 53.94.

Temporal Person Similarity	HOTA
IoU	53.59
OKS	53.81
IoU + OKS	53.94

(a)

Re-ID Network	Pose-Conditioned	HOTA
ResNet50 Luo et al. (2019b)		53.47
SPAPDE (ResNet50)	✓	53.94

(b)

Table 6.7: (a) Impact of different temporal person similarities used to calculate the pose similarity embeddings. (b) Impact of the proposed SPAPDE network for re-identification on the overall performance on PoseTrack21.

Impact of different pose similarity embeddings To update the pose similarity embeddings in the gated decoder layer, we utilize an α -Gate to weight the contributions of the appearance-based and pose similarity-based attention logits (6.5). While we evaluate the impact of α in Fig. 6.5a, Table 6.6 shows that learning α does not improve the results. We further evaluate the tracking performance when the fusion is not done at the logits, i.e., before the softmax, but after the softmax as in (6.2), i.e.,

$$\hat{E}_{T \rightarrow D}^t = FFN_E(A_{:, :-1}), \quad (6.14)$$

where $A_{:, :-1}$ denotes the attention weights without the last column (Section 6.2.1). The last row in Table 6.6 shows that the performance largely decreases from a HOTA score of 53.94 to 50.53 in this case.

Pose Similarity Embedding Update: In a subsequent experiment, we evaluate the impact of different embedded pose similarities, as shown in Table 6.7a. IoU only provides coarse information and does not allow distinguishing between spatially closely located person instances. Even though a temporal similarity based on IoU achieves a HOTA score of 53.59 and outperforms all related works (Table 6.1), OKS-based temporal similarity performs better, and the combination of both performs best.

Re-Identification Model: To evaluate the impact of the re-identification network on our proposed gated attention transformer, we trained our tracking model with two different re-identification networks: We used the re-identification network from ([Luo et al., 2019b](#)) and the proposed pose-conditioned SPAPDE network (Section 6.2.3). Table 6.7b shows that adding Spatially Adaptive Pose DE-normalization layers (SPAPDE) to the network increases HOTA from 53.47 to 53.94.

Moreover, we evaluate the person re-identification model on PoseTrack21 and measure the performance in terms of mean average precision (mAP), i.e., we calculate the area under the Precision-Recall curve for each query and average over all queries. As we show in Table 6.8, training a ResNet50 ([He et al., 2016](#)) with proper data sampling results in a significant performance gain and results in a mAP score of 68.65. In more detail, we sample $K = 6$ different instances of the same person identity for every batch. Surprisingly, training the ResNet50 from scratch results in a better mAP performance (68.65) compared to training a ResNet50 pre-trained on ImageNet ([Deng et al., 2009](#)) (68.31). As discussed in Section 6.2.3, we follow ([Luo et al., 2019b](#)), further increasing the performance to 73.78. To incorporate pose information, we replace each batch normalization layer with SPAPDE layers ((6.9)), which incorporate pose information by the keypoint heatmaps. The best

Model	GT Poses	Kernel Width	Sampling	pre-trained	mAP	loss
ResNet50				✓	64.24	triplet
ResNet50			✓	✓	68.31	triplet
ResNet50			✓		68.65	triplet
ResNet50 & back-of-tricks Luo et al. (2019b)			✓		73.38	triplet + ce + center
SPAPDE ResNet50	✓	2	✓		66.77	triplet
SPAPDE ResNet50	✓	5	✓		70.39	triplet
SPAPDE ResNet50	✓	10	✓		71.17	triplet
SPAPDE ResNet50	✓	15	✓		69.62	triplet
SPAPDE ResNet50 & bag-of-tricks Luo et al. (2019b)	✓	10	✓		78.00	triplet + ce + center
SPAPDE ResNet50 & bag-of-tricks Luo et al. (2019b)		10	✓		74.42	triplet + ce + center

Table 6.8: Person re-identification performance (mAP) on the PoseTrack21 dataset for various model settings.

Track Embedding Initialization	AssA	FragA	DetA	HOTA
Detection Embedding	61.87	60.73	47.15	53.76
New Track Embedding Head	62.20	60.93	47.20	53.94

Table 6.9: Impact of the *New Track Embedding Head* on the overall tracking performance on PoseTrack21.

performance is achieved by using keypoint heatmaps with a kernel size of 10. Specifically, we denote the kernel size as the standard deviation of a Gaussian distribution. For each keypoint, we calculate a Gaussian distribution with the mean set to the respective keypoint location and a standard deviation of 10. Further following Luo et al. (2019b), we achieve a total performance of 78.0 and 74.42 with ground truth and estimated poses, respectively. By adding SPAPDE, mAP thus increases from 73.78 to 74.42.

New Track Embedding Head: Given detections in frame t that can not be matched to any existing track, we use the *New Track Embedding Head* (Figure 6.2d)) to generate new track embeddings from the unmatched detection embeddings. In Table 6.9, we evaluate the impact of the *New Track Embedding Head* compared to initializing new tracks directly from the detection embeddings. The *New Track Embedding Head* provides a better initialization of the track embeddings.

6.3.4.2 Hyperparameter Evaluation

Finally, we evaluate the impact of α , τ_{age} and τ_{dup} on the overall tracking performance on the PoseTrack21 dataset. If not otherwise specified, we set $\alpha = 0.3$, $\tau_{age} = 60$ and $\tau_{dup} = 0.4$.

Impact of α : We evaluate the impact of α in the α -Gate ((6.3) and (6.5)) in Figure 6.5a. α weights the contribution of the appearance-based attention weights and the pose-based attention weights, where $\alpha = 1.0$ only considers appearance-based attention weights and, vice versa, $\alpha = 0.0$ only considers pose similarity-based attention weights. As we can observe in Figure 6.5a, the tracking performance drastically decreases for $\alpha > 0.4$. Appearance-based person features are very sensitive to persons with similar appearances, as it is common in sports videos, resulting in false associations and a high degree of identity switches. For $0 \leq \alpha \leq 0.3$, we can observe a linear increase in

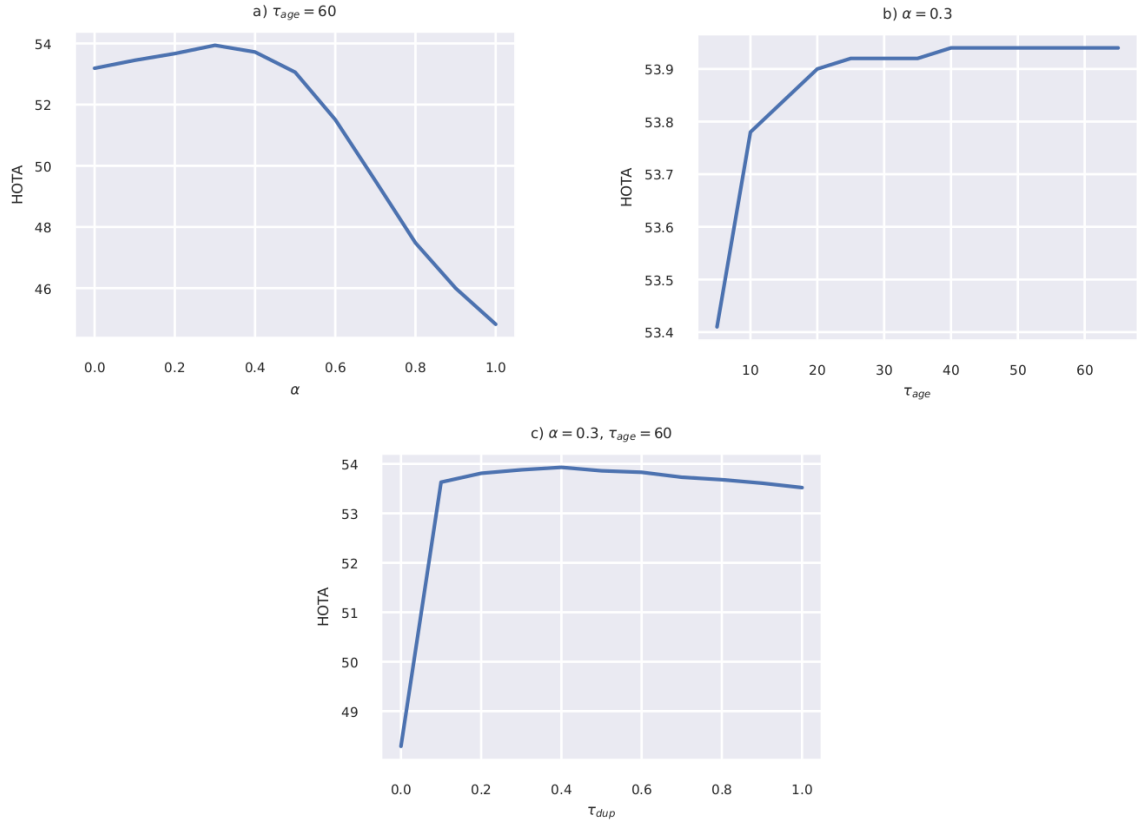


Figure 6.5: Impact of the parameters α , maximum track ages τ_{age} and the duplicate confidence threshold τ_{dup} on the tracking performance. a) visualizes the impact of α with $\tau_{age} = 60$ and $\tau_{dup} = 0.4$. b) shows the performance evaluation with respect to the maximum track age τ_{age} with $\alpha = 0.3$ and $\tau_{dup} = 0.4$. c) illustrates the impact of τ_{dup} with $\alpha = 0.3$ and $\tau_{age} = 60$.

the overall performance, peaking at $\alpha = 0.3$. This shows that pose-based and appearance-based similarities complement each other. Pose similarities provide strong guidance between consecutive frames, while appearance-based features allow to recover inactive tracks, e.g., due to occlusion.

Impact of τ_{age} : We terminate tracks that have not been tracked for more than τ_{age} frames and do not include them for the detection-to-track matching anymore. Figure 6.5b shows that the accuracy saturates at $\tau_{age} = 40$.

Impact of τ_{dup} : During tracking, we remove unmatched detections if they have a matching confidence $m_{ij} > \tau_{dup}$ with an already matched track (Section 6.2.1). Figure 6.5c shows that the accuracy drops without such a threshold since duplicates generate new tracks in this case.

6.3.5 Comparison to CORRTRACK + REID

Runtime Analysis Ultimately, we measure the average runtime of our proposed Gated Attention Transformer compared to the baseline method CORRTRACK + REID. As both methods are not limited to a specific pose estimation framework, we estimate the tracking runtime independent of the person detector and pose estimation pipelines. While the pose estimation pipeline runs at 2.3 frames per second (fps), the tracking stage of CORRTRACK + REID achieves an average runtime of 6.54 fps. In contrast, our proposed tracker is **3.8** times faster and runs with an average runtime of 25.07 fps.

Qualitative Results Fig. 6.6 shows additional qualitative examples of our proposed method in direct comparison to CORRTRACK + REID (Chapter 5) on the PoseTrack21 dataset. Additional videos are available at <https://youtu.be/uFXD4mWPajo> and <https://youtu.be/IG6GkWSINQU>.

6.4 Conclusion

In this chapter, we introduced a novel Gated Attention Transformer framework for multi-person pose tracking, building on the advancements from the earlier chapters. In Chapter 4, we addressed the challenge of sparse keypoint annotations in previous datasets (i.e., PoseTrack 2017 and PoseTrack 2018 (Andriluka et al., 2018)) by proposing a self-supervised learning approach for keypoint correspondences. However, issues such as duplicate poses and identity switches, especially in crowded scenes and for occluded persons with highly unusual poses, persisted.

In Chapter 5, we introduced PoseTrack21, a densely annotated dataset with person identities, offering a more robust training and evaluation foundation. Building upon Chapter 4, we proposed baseline methods incorporating re-identification features into the tracking process. Yet, occlusions and unusual poses remained challenging to handle by the proposed baseline methods.

The Gated Attention Transformer framework, introduced in this chapter, directly addresses these unresolved issues. Our method employs a duplicate-aware association and dynamically adapts via gates the impact of pose-based and appearance-based similarities based on the attention probabilities of each similarity measure. Thus, our method improved the handling of occlusions and pose variations, which were challenges in both Chapters 4 and 5.

Evaluated on PoseTrack21, our approach outperforms prior methods, showing improvements in multi-person pose tracking. On PoseTrack 2018, the approach is only outperformed by methods that use a more expensive human pose estimator. We also evaluated the impact of the proposed Spatially Adaptive Pose DEnormalization (SPAPDE) on the pose tracking performance on PoseTrack21, which positively impacts the overall tracking performance.

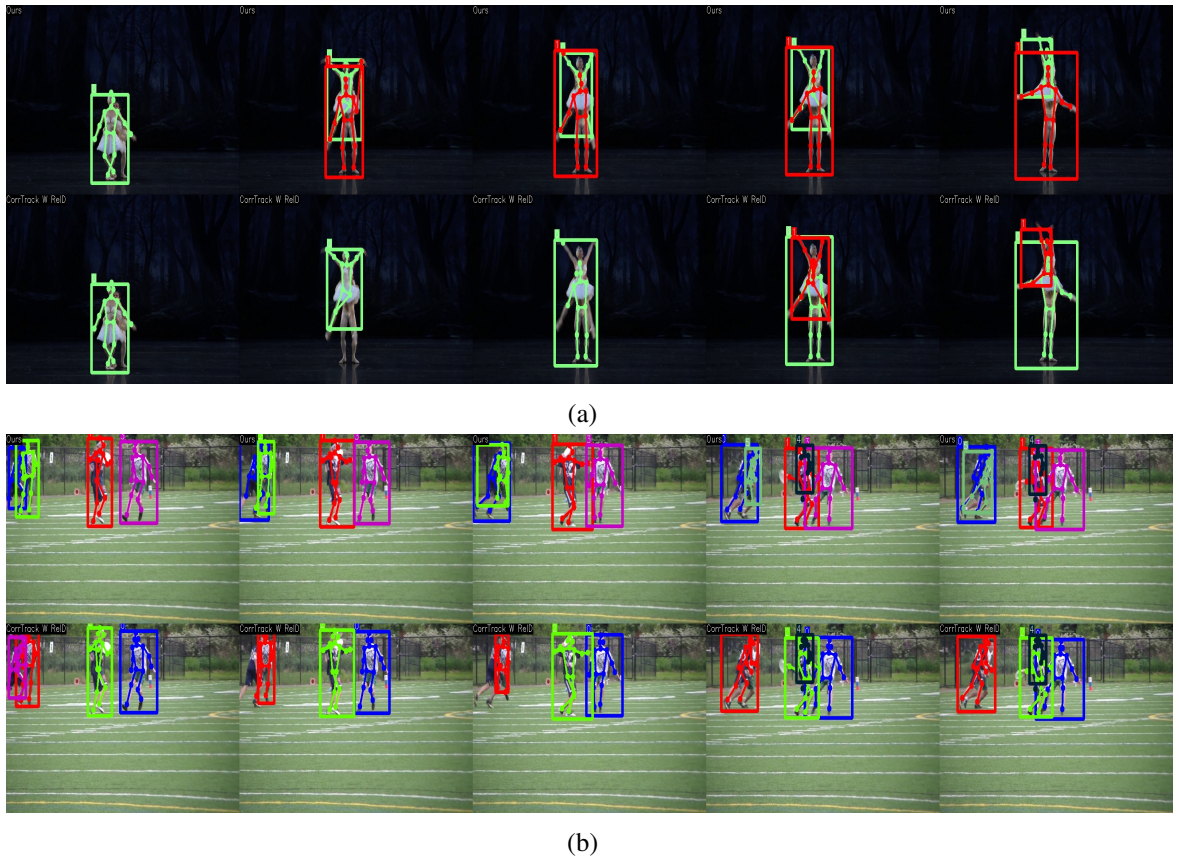


Figure 6.6: Additional qualitative examples of our proposed method on the PoseTrack21 dataset. In both (a) and (b), the first row contains visual tracking results of our method and the second row shows visualizations of CorrTrack with ReID (Doering et al., 2022).

Conclusion

Contents

7.1 Overview	127
7.2 Contributions and Discussion	128
7.3 Future Work	129

7.1 Overview

This thesis aims to advance the field of 2D multi-person pose estimation and tracking in unconstrained images and videos. Over the past years, 2D multi-person pose tracking has witnessed remarkable progress. However, existing evaluation metrics on multi-person pose tracking, such as MOTA, penalize detection inaccuracies more than tracking performance. Consequently, several works developed methods to increase pose localization accuracy. Task-specific similarity metrics, such as those based on appearance, have been demonstrated to significantly outperform prior works that aim to solve spatiotemporal graphs in tracking performance and runtime. Nevertheless, existing metrics used in multi-person pose tracking are often suitable for the association of poses between a limited time window of consecutive frames, and frame rates and object movements mostly impact them. This leads to broken tracks that are often unrecoverable, particularly after long periods of occlusion. The contributions of this thesis include the development of appearance-based similarity metrics for tracking, which increase robustness towards several nuances factors of noise, such as occlusion and fast motions. Additionally, we proposed methods to recover tracks after periods of occlusion. The ability to robustly track a person or a person's pose is not limited to a single sequence but should work throughout multiple scenes and videos. Tracking human poses throughout multiple streams of video data is essential for many applications ranging from smart surveillance systems to interactive virtual reality environments, sports analytics, and advanced human-computer interaction systems. This capability allows for continuous monitoring and analysis of human behavior, enabling real-time responses and deeper insights into activities and movements across various contexts and environments. Existing datasets do not provide sufficient annotations, such as person identities. We proposed PoseTrack21, a new dataset that comprises rich annotations, including keypoints, bounding boxes, and person identities. Furthermore, we proposed a new evaluation metric that balances the contribution of localization accuracy and tracking accuracy, enabling the evaluation of pose tracking performance throughout multiple sequences.

In summary, this thesis significantly contributes to the field of 2D multi-person pose estimation and tracking by introducing novel methods, datasets, and evaluation metrics. The developed methods

in the context of this thesis are applicable in entirely unconstrained scenarios and provide a foundation for future research in the field, and have the potential to impact various applications, including sports video analysis, surveillance, and human-robot interaction.

7.2 Contributions and Discussion

In Chapter 4, we contended that current multi-person pose estimation and tracking datasets lack comprehensive annotations and diversity. As a result, training deep learning methods to effectively handle various sources of noise, such as motion blur, becomes challenging. To address this issue, we introduced a keypoint correspondence framework that has been trained using self-supervision, eliminating the requirement for video data. The framework allows the recovery of missed keypoints detections and allows to associate human poses across video frames. We further show that keypoint correspondences outperform flow- and motion-based methods (Xiao et al., 2018; Sun et al., 2019a; Raaj et al., 2019; Ning and Huang, 2019) that are prone to fast camera motions and occlusions. We evaluated the approach on the PoseTrack 2017 and 2018 datasets (Andriluka et al., 2018), where we improved accuracy for the tasks of multi-frame multi-person pose estimation and multi-person pose tracking.

In Chapter 5, we introduced PoseTrack21, a densely annotated dataset containing annotated bounding boxes, human keypoints, and person-ids. Our experiments demonstrated that MOT approaches, in combination with pose estimation, serve as suitable baselines for multi-person pose tracking. Conversely, the pose estimation baselines also perform well within the context of MOT. Most person search baselines are built upon an object detector with an extended re-identification head. As a result, PoseTrack21 enables researchers to develop joint approaches capable of reliably addressing multi-person pose tracking, multi-object tracking, and person search. Furthermore, we introduced keypoint HOTA, a new evaluation metric that addresses the imbalance between detection and association accuracy present in previous metrics like MOTA. Our metric also allows for evaluating multi-person re-id pose tracking performance, which measures a tracker’s ability to track human poses across videos. Therefore, PoseTrack21 serves as an important benchmark for future work.

Due to the unique characteristics of the MOTA metric, which has been utilized in previous datasets such as PoseTrack 2017 and PoseTrack 2018 for multi-person pose tracking (Andriluka et al., 2018), it is evident that the metric is highly influenced by the accuracy of keypoint detection (Luiten et al., 2020), rather than the performance of the tracker’s association (cf. Section 3.6.3). As a result, many related works have emphasized enhancing the performance of pose estimators to achieve state-of-the-art pose tracking results. As discussed in Chapter 6, we propose a novel gated attention approach for multi-person pose tracking, which is built on top of an off-the-shelf pose estimation approach we introduced in Chapter 4. One of the challenges faced by multi-person pose tracking systems is the handling of redundant detections, which can occur due to inaccuracies in the underlying person detector or pose estimator. If not properly managed, redundant detections can lead to broken tracks and numerous identity switches, ultimately diminishing the overall performance of the tracker. Our method addresses this challenge by incorporating a duplicate-aware association and dynamically adjusting the impact of duplicates through attention gates based on pose-based and appearance-based similarities. The evaluation of our method on the PoseTrack21 dataset demonstrates its superiority over previous works. Additionally, on PoseTrack 2018, our approach is only

surpassed by methods that utilize more sophisticated human pose estimators. Furthermore, to extract robust appearance-based features, we introduced Spatially Adaptive Pose Denormalization (SPAPDE) for pose-conditioned feature extraction. Our experiments demonstrate the positive impact of SPAPDE on pose tracking performance on PoseTrack21, contributing to an overall enhancement in tracking performance.

7.3 Future Work

As discussed at the beginning of this thesis, to develop autonomous systems that can analyze and anticipate human behavior for a natural interaction between humans and machines, we require a holistic understanding of the human body in various challenging scenarios, including severe occlusion and crowded scenes.

Handling Complex Articulations and Person-Person Interactions

In Chapter 5, we introduced PoseTrack21, a large-scale dataset comprising various real-world sequences, including a wide range of sports videos. These videos naturally involve complex articulations and interactions between people, presenting significant challenges for accurately estimating keypoints in these scenarios. Current state-of-the-art methods struggle with these challenges, particularly in person-person interactions. Most contemporary approaches assign equal weights to all samples during training, failing to prioritize more difficult cases. However, [Zhou et al. \(2020a\)](#) has explored the potential of re-weighting and re-estimating challenging keypoints to improve accuracy. Beyond these methods, investigating online negative mining techniques could be a promising direction for future research.

Moreover, close person-person interactions often lead to misplaced keypoints, e.g., the pose estimator localizes the left wrist of person A at the location of person B's right wrist. While early works ([Insafutdinov et al., 2017](#); [Iqbal et al., 2017](#)) suggested solving fully-connected spatiotemporal graphs for multi-person pose estimation and tracking, their computational complexity is not feasible for real-world applications. To overcome these limitations, [Ding et al. \(2022\)](#) propose intra-human and inter-human relation components into their pose estimation architecture and bridge the gap to human-human interactions. As the behavior of humans is strongly impacted by their surrounding humans, especially in crowded scenes and during sports activities, it is crucial to account for these interactions to achieve accurate pose estimation and tracking performance. Overall, the integration of intra-human and inter-human relations into pose estimation and pose tracking frameworks can lead to a significant advancement in the field. It not only improves the accuracy and reliability of pose estimations in challenging scenarios but also bridges the gap towards a more comprehensive understanding of human behavior in interactive and dynamic environments. As such, these innovations hold substantial potential for applications ranging from surveillance and public safety to sports analytics and virtual reality and should be further investigated by future works.

End-to-End Multi-Person Re-ID Pose Estimation and Tracking

In Chapter 5, we discussed that most multi-person pose estimation and tracking systems are built from disjoint components. The ultimate goal is to develop a unified system that can detect persons,

estimate poses, and track them. Previous works have mainly focused on multi-person pose tracking within a single sequence. However, future works should investigate re-identifying and tracking poses across multiple cameras and sequences, thus requiring additional components, such as person re-identification systems. While person re-identification methods have been successfully employed in various tasks, such as multi-object tracking (Zhang et al., 2022) or multi-camera multi-object tracking (Amosa et al., 2023), many methods use off-the-shelf re-identification methods. One of the reasons for disjoint approaches was the lack of datasets that provide sufficient annotations required for the training of such end-to-end systems. With the release of large-scale synthetic datasets (Fabbri et al., 2021) in combination with our proposed dataset, PoseTrack21, future research should focus on unified models.

Furthermore, most tracking methods rely on the Hungarian algorithm Kuhn (1955) for the assignment of detections and tracks. Since the Hungarian algorithm is not differentiable, another interesting direction would be to investigate differentiable approximations, such as the Sinkhorn algorithm (Sinkhorn and Knopp, 1967; Cuturi, 2013), which has already been successfully incorporated for object detection De Plaen et al. (2023). The Sinkhorn algorithm can be seen as a continuous relaxation of the assignment problem, paving the way for fully differentiable trackers.

3D Multi-Person Pose Estimation and Tracking

3D multi-person pose estimation and tracking is one of the most challenging problems in articulated pose estimation and remains far from being applicable to unconstrained, in-the-wild scenarios. Due to the inherently difficult nature of 3D pose estimation, many systems rely on multiple cameras (Wang et al., 2021e; Srivastav et al., 2024; Tu et al., 2020), which necessitate calibration and are often restricted to controlled environments. A common approach (Zheng et al., 2021; Park et al., 2023; Su et al., 2022) in 3D multi-person pose estimation is to lift 2D poses to 3D instead of directly regressing 3D human poses from pixel information. This method allows for the generation of an unlimited number of 2D-3D pairs under various conditions using given camera parameters.

In contrast to single-person 3D pose estimation, estimating the 3D positions of multiple individuals requires accurately determining the geometric relationships among 3D keypoints of different persons and estimating the depth of all individuals. Park et al. (2023) employ an off-the-shelf multi-person pose tracking pipeline based on the works of Sun et al. (2019a) and Zhang et al. (2022) and proposes a transformer architecture that simultaneously regresses 3D poses from a sequence of tracked 2D keypoints. Similarly, (Zou et al., 2022) aggregate spatiotemporal information and propose to unify multi-person 3D pose estimation, tracking, and forecasting into a single framework. This approach addresses each sub-problem independently.

Similar to related works that lift 2D to 3D poses, the process of 2D pose estimation and tracking suffers from noise, especially in the presence of occlusion, and the accuracy of the 3D estimates is highly dependent on the correctness of the 2D estimates. While existing methods explore promising directions, most lack robustness when dealing with unseen views, are vulnerable to occlusion and exhibit severe jittering in the output.

In Chapter 4, we trained a keypoint correspondence network in a self-supervised fashion using random affine transformations on image data, ensuring that the network learns to identify and match keypoints accurately despite variations in orientation and scale. Similarly, Srivastav et al. (2024)

employ random affine transformations applied to a sequence of multi-view images for training their multi-person 3D pose estimation baseline. This technique ensures that the pose estimation network can robustly handle the variability in human poses and viewpoints present in real-world scenarios. Multi-view consistency is crucial in ensuring that the keypoints inferred from different views are coherent and accurately represent the 3D structure of the scene. In contrast, self-supervised learning from monocular images presents a unique set of challenges and opportunities. Without the benefit of multiple viewpoints, the network must rely solely on visual cues present within a single image. By harnessing the abundant unlabelled visual data available, self-supervised learning can significantly enhance the generalization capabilities of the network, enabling it to perform well across diverse environments and conditions. Due to the limited amount of training data available, self-supervised learning is a promising direction for future works.

Holistic Understanding of Humans

To build autonomous systems that can naturally interact with humans, a holistic understanding of the human body is essential, including but not limited to face, hands, and body shape. While 2D pose estimation and tracking provide critical information about the arrangement of human joints, it lacks depth information, which is crucial for understanding the actual spatial configuration and interactions in 3D space, required for autonomous systems. 3D human pose and shape regression (HPS) systems leverage techniques such as parametric models (e.g., SMPL) that allow the representation of the human body more realistically. Thus, HPS can combine the advantages of dense pose regression, body part segmentation, and 3d pose estimation into a single problem formulation. In recent years, HPS systems have become highly robust and accurate. One objective of HPS regression is to minimize the projection error between 2D and 3D keypoints. Dwivedi et al. (2024) recently identified a common problem in HPS systems, which either achieve good 3D accuracy or good alignment with the image. Their work shows that a wrong estimation of the camera model is responsible for the misalignment; thus, more research in this direction is required. This might include full-body pose estimation to resolve ambiguities in global orientation estimation. Aside from improving accuracy, robustness, and real-time capabilities, an interesting direction for future work would be the development of a holistic system that directly predicts 3D human pose and shape from RGB input in unconstrained environments, instead of relying on sub-systems, e.g., for person detection or 2D pose estimation.

Moreover, accurate 3D poses and shapes act as a strong prior for the analysis of human-human, human-machine, or human-object interactions, which is an essential component for autonomous systems and for the ability to anticipate human behavior. Future work should also expand their methods to 3D multi-person human pose and shape estimation.

Bibliography

- Aharon, Nir; Orfaig, Roy, and Bobrovsky, Ben-Zion. BoT-SORT: Robust Associations Multi-Pedestrian Tracking. *ArXiv*, 2022. (Cited on page 28.)
- Amosa, Temitope Ibrahim; Sebastian, Patrick; Izhar, Lila Iznita; Ibrahim, Oladimeji; Ayinla, Lukman Shehu; Bahashwan, Abdulrahman Abdullah; Bala, Abubakar, and Samaila, Yau Alhaji. Multi-camera multi-object tracking: A review of current trends and future advances. *Neurocomputing*, 2023. (Cited on page 130.)
- Andriluka, M.; Iqbal, U.; Ensafutdinov, E.; Pishchulin, L.; Milan, A.; Gall, J., and B., Schiele. PoseTrack: A Benchmark for Human Pose Estimation and Tracking. In *CVPR*, 2018. (Cited on pages ix, xi, 2, 3, 17, 20, 57, 68, 76, 88, 89, 90, 92, 93, 94, 96, 98, 102, 107, 109, 117, 118, 124 and 128.)
- Andriluka, Mykhaylo; Roth, Stefan, and Schiele, Bernt. People-tracking-by-detection and people-detection-by-tracking. In *CVPR*, 2008. (Cited on page 20.)
- Andriluka, Mykhaylo; Roth, Stefan, and Schiele, Bernt. Pictorial Structures Revisited: People Detection and Articulated Pose Estimation. In *CVPR*, 2009. (Cited on page 7.)
- Andriluka, Mykhaylo; Roth, Stefan, and Schiele, Bernt. Discriminative Appearance Models for Pictorial Structures. *ICJV*, 2012. (Cited on page 10.)
- Andriluka, Mykhaylo; Pishchulin, Leonid; Gehler, Peter, and Schiele, Bernt. 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. In *CVPR*, 2014. (Cited on pages 3, 60, 68, 90, 92, 93, 94 and 96.)
- Artacho, Bruno and Savakis, Andreas. UniPose: Unified Human Pose Estimation in Single Images and Videos. In *CVPR*, 2020. (Cited on pages 18 and 19.)
- Artacho, Bruno and Savakis, Andreas. BAPose: Bottom-Up Pose Estimation with Disentangled Waterfall Representations. In *WACVW*, 2023. (Cited on pages 9, 14, 16 and 55.)
- Ba, Jimmy Lei; Kiros, Jamie Ryan, and Hinton, Geoffrey E. Layer Normalization, 2016. (Cited on pages 46, 47, 51, 113 and 115.)
- Bae, Seung-Hwan and Yoon, Kuk-Jin. Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning. In *CVPR*, 2014. (Cited on page 28.)
- Bae, Seung-Hwan and Yoon, Kuk-Jin. Confidence-Based Data Association and Discriminative Deep Appearance Learning for Robust Online Multi-Object Tracking. *TPAMI*, 2018. (Cited on pages 26 and 28.)
- Ballas, Nicolas; Yao, L.; Pal, Christopher Joseph, and Courville, Aaron C. Delving deeper into convolutional networks for learning video representations. *arxiv-Preprint*, 2015. (Cited on page 30.)

- Bao, Qian; Liu, Wu; Cheng, Yuhao; Zhou, Boyan, and Mei, Tao. Pose-Guided Tracking-by-Detection: Robust Multi-Person Pose Tracking. In *IEEE Transactions on Multimedia*, 2020. (Cited on page 83.)
- Berclaz, Jerome; Fleuret, Francois; Turetken, Engin, and Fua, Pascal. Multiple Object Tracking Using K-Shortest Paths Optimization. *TPAMI*, 2011. (Cited on pages 32 and 33.)
- Bergmann, Philipp; Meinhardt, Tim, and Leal-Taixé, Laura. Tracking Without Bells and Whistles. In *ICCV*, 2019. (Cited on pages 26, 30, 88, 94, 95, 96, 98, 99, 102, 117 and 118.)
- Bernardin, Keni and Stiefelhagen, Rainer. Evaluating multiple object tracking performance: The clear mot metrics. *EURASIP J. Image Video Process.*, 2008, 2008. (Cited on page 62.)
- Bertasius, Gedas; Feichtenhofer, Christoph; Tran, Du; Shi, Jianbo, and Torresani, Lorenzo. Learning Temporal Pose Estimation from Sparsely-Labeled Videos. In *NeurIPS*, 2019. (Cited on pages 18, 19, 68, 79 and 82.)
- Bewley, Alex; Ge, Zongyuan; Ott, Lionel; Ramos, Fabio, and Upcroft, Ben. Simple online and realtime tracking. In *ICIP*, 2016. (Cited on pages 26 and 28.)
- Bochinski, Erik; Eiselein, Volker, and Sikora, Thomas. High-Speed tracking-by-detection without using image information. *AVSS*, 2017. (Cited on page 27.)
- Bourdev, Lubomir D. and Malik, Jitendra. Poselets: Body part detectors trained using 3D human pose annotations. In *ICCV*, 2009. (Cited on page 8.)
- Brasó, Guillem and Leal-Taixé, Laura. Learning a Neural Solver for Multiple Object Tracking. *CVPR*, 2020. (Cited on pages 32, 33 and 88.)
- Brasó, Guillem; Kister, Nikita, and Leal-Taixé, Laura. The Center of Attention: Center-Keypoint Grouping via Attention for Multi-Person Pose Estimation. *ICCV*, 2021. (Cited on page 16.)
- Brasó, Guillem; Cetintas, Orcun, and Leal-Taixé, Laura. Multi-object tracking and segmentation via neural message passing. *IJCV*, 2022. (Cited on pages 32 and 33.)
- Bregler, Christoph. Learning and Recognizing Human Dynamics in Video Sequences. In *CVPR*, 1997. (Cited on page 7.)
- Breiman, Leo. Random Forests. *Machine Learning*, 2001. (Cited on page 10.)
- Butt, A. and Collins, R. Multi-target Tracking by Lagrangian Relaxation to Min-Cost Network Flow. In *CVPR*, 2013. (Cited on page 32.)
- Cai, Jiarui; Xu, Mingze; Li, Wei; Xiong, Yuanjun; Xia, Wei; Tu, Zhuowen, and Soatto, Stefan O. MeMOT: Multi-Object Tracking with Memory. *CVPR*, 2022. (Cited on pages 31 and 32.)
- Cai, Yuanhao; Wang, Zhicheng; Luo, Zhengxiong; Yin, Binyi; Du, Angang; Wang, Haoqian; Zhou, Xinyu; Zhou, Erjin; Zhang, Xiangyu, and Sun, Jian. Learning Delicate Local Representations for Multi-Person Pose Estimation. In *ECCV*, 2020. (Cited on pages 11 and 53.)

- Cai, Zhaowei and Vasconcelos, Nuno. Cascade R-CNN: Delving Into High Quality Object Detection. In *CVPR*, 2017. (Cited on pages 25, 36, 53, 70, 74, 76, 118 and 119.)
- Cao, Jinkun; Weng, Xinshuo; Khirodkar, Rawal; Pang, Jiangmiao, and Kitani, Kris. Observation-Centric SORT: Rethinking SORT for Robust Multi-Object Tracking. *CVPR*, 2023. (Cited on page 27.)
- Cao, Zhe; Simon, Tomas; Wei, Shih-En, and Sheikh, Yaser. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. In *CVPR*, 2017. (Cited on pages vii, 7, 9, 14, 15, 20, 22, 53, 55, 56, 57, 58, 68 and 73.)
- Carion, Nicolas; Massa, Francisco; Synnaeve, Gabriel; Usunier, Nicolas; Kirillov, Alexander, and Zagoruyko, Sergey. End-to-End Object Detection with Transformers. In *ECCV*, 2020. (Cited on pages 13, 26, 31 and 32.)
- Cetintas, Orcun; Brasó, Guillem, and Leal-Taixé, Laura. Unifying short and long-term tracking with graph hierarchies. In *CVPR*, 2023. (Cited on pages 32 and 33.)
- Chen, Di; Zhang, Shanshan; Ouyang, Wanli; Yang, Jian, and Tai, Ying. Person search via a mask-guided two-stream cnn model. In *ECCV*, 2018a. (Cited on page 34.)
- Chen, Di; Zhang, Shanshan; Ouyang, Wanli; Yang, Jian, and Schiele, Bernt. Hierarchical Online Instance Matching for Person Search. In *AAAI*, 2020a. (Cited on pages 35, 55, 88 and 100.)
- Chen, Di; Zhang, Shanshan; Yang, Jian, and Schiele, Bernt. Norm-Aware Embedding for Efficient Person Search. In *CVPR*, 2020b. (Cited on pages vii, xi, 35, 36, 60, 88 and 100.)
- Chen, Di; Doering, Andreas; Zhang, Shanshan; Yang, Jian; Gall, Juergen, and Schiele, Bernt. Key-point Message Passing for Video-based Person Re-Identification. *AAAI*, 2022. (Cited on pages 33 and 91.)
- Chen, Kai; Pang, Jiangmiao; Wang, Jiaqi; Xiong, Yu; Li, Xiaoxiao; Sun, Shuyang; Feng, Wansen; Liu, Ziwei; Shi, Jianping; Ouyang, Wanli; Loy, Chen Change, and Lin, Dahua. Hybrid Task Cascade for Instance Segmentation. *CVPR*, 2019a. (Cited on page 118.)
- Chen, Lequan; Xie, Wei; Tu, Zhigang; Guo, Jinglei; Tao, Yaping, and Wang, Xinming. Multi-Attribute Enhancement Network for Person Search. *IJCNN*, 2021a. (Cited on page 88.)
- Chen, Liang-Chieh; Papandreou, George; Kokkinos, Iasonas; Murphy, Kevin, and Yuille, Alan L. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. In *ICLR*, 2015. (Cited on page 10.)
- Chen, Longtao; Peng, Xiaojiang, and Ren, Mingwu. Recurrent Metric Networks and Batch Multiple Hypothesis for Multi-Object Tracking. *IEEE Access*, 2019b. (Cited on pages 28 and 29.)
- Chen, Peixian; Liu, Wenfeng; Dai, Pingyang; Liu, Jianzhuang; Ye, Qixiang; Xu, Mingliang; Chen, Qi'an, and Ji, Rongrong. Occlude them all: Occlusion-aware attention network for occluded person re-id. In *ICCV*, 2021b. (Cited on pages 33, 34 and 119.)

- Chen, Weihua; Xu, Xianzhe; Jia, Jian; Luo, Hao; Wang, Yaohua; Wang, Fan; Jin, Rong, and Sun, Xiuyu. Beyond appearance: a semantic controllable self-supervised learning framework for human-centric visual tasks. In *CVPR*, 2023. (Cited on page 33.)
- Chen, Xianjie and Yuille, Alan L. Articulated Pose Estimation by a Graphical Model with Image Dependent Pairwise Relations. In *NIPS*, 2014. (Cited on page 8.)
- Chen, Yilun; Wang, Zhicheng; Peng, Yuxiang; Zhang, Zhiqiang; Yu, Gang, and Sun, Jian. Cascaded Pyramid Network for Multi-Person Pose Estimation. In *CVPR*, 2018b. (Cited on pages 11 and 23.)
- Chen, Z. and Lee, H.-J. Knowledge-guided visual perception of 3-D human gait from a single image sequence. *IEEE Transactions on Systems, Man, and Cybernetics*, 1992. (Cited on page 7.)
- Cheng, Bowen; Xiao, Bin; Wang, Jingdong; Shi, Honghui; Huang, Thomas S., and Zhang, Lei. HigherHRNet: Scale-Aware Representation Learning for Bottom-Up Human Pose Estimation. In *CVPR*, 2020. (Cited on pages 15, 16 and 17.)
- Choi, Wongun. Near-online multi-target tracking with aggregated local flow descriptor. In *ICCV*, 2015. (Cited on page 32.)
- Chu, Peng and Ling, Haibin. FAMNet: Joint Learning of Feature, Affinity and Multi-Dimensional Assignment for Online Multiple Object Tracking. *ICCV*, 2019. (Cited on page 29.)
- Chu, Peng; Fan, Heng; Tan, Chiu Chiang, and Ling, Haibin. Online Multi-Object Tracking With Instance-Aware Tracker and Dynamic Model Refreshment. *WACV*, 2019. (Cited on page 29.)
- Ciaparrone, Gioele; Luque Sánchez, Francisco; Tabik, Siham; Troiano, Luigi; Tagliaferri, Roberto, and Herrera, Francisco. Deep learning in video multi-object tracking: A survey. *Neurocomputing*, 2020. (Cited on pages 24 and 25.)
- Criminisi, A. and Shotton, J. *Decision Forests for Computer Vision and Medical Image Analysis*. Springer Publishing Company, Incorporated, 2013. ISBN 1447149289, 9781447149286. (Cited on page 8.)
- Cuturi, Marco. Sinkhorn distances: Lightspeed computation of optimal transport. In Burges, C.J.; Bottou, L.; Welling, M.; Ghahramani, Z., and Weinberger, K.Q., editors, *NeurIPS*, 2013. (Cited on page 130.)
- Cybenko, G. Approximation by superpositions of a sigmoidal function. *MCSS*, 1989. (Cited on page 40.)
- Dai, Jifeng; Qi, Haozhi; Xiong, Yuwen; Li, Yi; Zhang, Guodong; Hu, Han, and Wei, Yichen. Deformable Convolutional Networks. In *ICCV*, 2017. (Cited on page 18.)
- Dai, Ju; Zhang, Pingping; Wang, Dong; Lu, Huchuan, and Wang, Hongyu. Video person re-identification by temporal residual learning. *TIP*, 2018. (Cited on page 33.)

- Dai, Peng; Weng, Renliang; Choi, Wongun; Zhang, Changshui; He, Zhangping, and Ding, Wei. Learning a Proposal Classifier for Multiple Object Tracking. *CVPR*, 2021. (Cited on pages 33 and 88.)
- Dalal, Navneet and Triggs, Bill. Histograms of Oriented Gradients for Human Detection. In *CVPR*, 2005. (Cited on page 8.)
- Dantone, M.; Gall, J.; Leistner, C., and van Gool, L. Body Parts Dependent Joint Regressors for Human Pose Estimation in Still Images. *TPAMI*, 2014. (Cited on pages 8, 9 and 73.)
- Daphne Tsatsoulis, P. and Forsyth, David. Easy Minimax Estimation with Random Forests for Human Pose Estimation. In *ECCV*, 2015. (Cited on page 8.)
- De Plaen, Henri; De Plaen, Pierre-François; Suykens, Johan A. K.; Proesmans, Marc; Tuytelaars, Tinne, and Van Gool, Luc. Unbalanced optimal transport: A unified framework for object detection. In *CVPR*, 2023. (Cited on page 130.)
- Dendorfer, Patrick; Rezatofighi, Hamid; Milan, Anton; Shi, Javen; Cremers, Daniel; Reid, Ian D.; Roth, Stefan; Schindler, Konrad, and Leal-Taixé, Laura. MOT20: A benchmark for multi object tracking in crowded scenes. *CVPR*, 2019. (Cited on pages 88, 89 and 90.)
- Dendorfer, Patrick; Yugay, Vladimir; Ošep, Aljoša, and Leal-Taixé, Laura. Quo vadis: Is trajectory forecasting the key towards long-term multi-object tracking? In *NeuRIPS*, 2022. (Cited on page 31.)
- Deng, Jia; Dong, Wei; Socher, Richard; Li, Li-Jia; Li, Kai, and Fei-Fei, Li. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. (Cited on pages 53 and 121.)
- Dijkstra, E. W. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1959. (Cited on page 24.)
- Ding, Yiwei; Deng, Wenjin; Zheng, Yinglin; Liu, Pengfei; Wang, Meihong; Cheng, Xuan; Bao, Jianmin; Chen, Dong, and Zeng, Ming. I²R-Net: Intra- and Inter-Human Relation Network for Multi-Person Pose Estimation. In *IJCAI*, 2022. (Cited on pages 12, 13 and 129.)
- Dinh, Laurent; Sohl-Dickstein, Jascha, and Bengio, Samy. Density estimation using Real NVP. In *ICLR*, 2017. (Cited on page 12.)
- Doering, Andreas. Multi person pose estimation in videos. Master's thesis, University of Bonn, 2018. (Cited on pages vii, 41 and 42.)
- Doering, Andreas and Gall, Juergen. A Gated Attention Transformer for Multi-Person Pose Tracking. In *AMFG*, 2023. (Cited on pages 5 and 107.)
- Doering, Andreas; Iqbal, Umar, and Gall, Juergen. Joint Flow: Temporal Flow Fields for Multi Person Tracking. *BMVC*, 2018. (Cited on pages vii, 2, 7, 20, 21, 22, 57, 58, 75, 108 and 109.)

- Doering, Andreas; Chen, Di; Zhang, Shanshan; Schiele, Bernt, and Gall, Juergen. PoseTrack21: A Dataset for Person Search, Multi-Object Tracking and Multi-Person Pose Tracking. In *CVPR*, 2022. (Cited on pages x, 4, 87, 117 and 125.)
- Dong, Wenkai; Zhang, Zhaoxiang; Song, Chunfeng, and Tan, Tieniu. Bi-directional interaction network for person search. *CVPR*, 2020a. (Cited on page 35.)
- Dong, Wenkai; Zhang, Zhaoxiang; Song, Chunfeng, and Tan, Tieniu. Instance guided proposal network for person search. *CVPR*, 2020b. (Cited on page 35.)
- Dosovitskiy, Alexey; Beyer, Lucas; Kolesnikov, Alexander; Weissenborn, Dirk; Zhai, Xiaohua; Unterthiner, Thomas; Dehghani, Mostafa; Minderer, Matthias; Heigold, Georg; Gelly, Sylvain; Uszkoreit, Jakob, and Houlsby, Neil. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICCV*, 2021. (Cited on pages 13 and 34.)
- Du, Yunhao; Zhao, Zhicheng; Song, Yang; Zhao, Yanyun; Su, Fei; Gong, Tao, and Meng, Hongying. Strongsort: Make deepsort great again. *IEEE Transactions on Multimedia*, 2023. (Cited on page 28.)
- Duan, Kaiwen; Bai, Song; Xie, Lingxi; Qi, Honggang; Huang, Qingming, and Tian, Qi. CenterNet: Keypoint Triplets for Object Detection. *ICCV*, 2019. (Cited on pages 9, 25, 26, 30, 31 and 32.)
- Dumoulin, Vincent and Visin, Francesco. A guide to convolution arithmetic for deep learning. *arXiv-Preprint*, 2016. (Cited on page 48.)
- Dwivedi, Sai Kumar; Sun, Yu; Patel, Priyanka; Feng, Yao, and Black, Michael J. TokenHMR: Advancing human mesh recovery with a tokenized pose representation. In *CVPR*, 2024. (Cited on page 131.)
- Eichner, Marcin and Ferrari, Vittorio. We Are Family: Joint Pose Estimation of Multiple Persons. In *ECCV*, 2010. (Cited on page 9.)
- Evangelidis, Georgios D. and Psarakis, Emmanouil Z. Parametric Image Alignment Using Enhanced Correlation Coefficient Maximization. *TPAMI*, 2008. (Cited on pages 26 and 95.)
- Everingham, M.; Van Gool, L.; Williams, C. K. I.; Winn, J., and Zisserman, A. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. <http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html>, 2007. (Cited on page 61.)
- Fabbri, Matteo; Lanzi, Fabio; Calderara, Simone; Palazzi, Andrea; Vezzani, Roberto, and Cucchiara, Rita. Learning to Detect and Track Visible and Occluded Body Joints in a Virtual World. In *ECCV*, 2018. (Cited on pages 88, 89, 90 and 91.)
- Fabbri, Matteo; Brasó, Guillem; Maueri, Gianluca; Cetintas, Orcun; Gasparini, Riccardo; Osep, Aljosa; Calderara, Simone; Leal-Taixé, Laura, and Cucchiara, Rita. MOTSynth: How Can Synthetic Data Help Pedestrian Detection and Tracking? *ICCV*, 2021. (Cited on pages 88, 90, 92 and 130.)

- Fan, Xing; Jiang, Wei; Luo, Hao; Mao, Weijie, and Yu, Hongyan. Instance Hard Triplet Loss for In-video Person Re-identification. *Applied Sciences*, 2020. (Cited on pages 90 and 91.)
- Fan, Zhipeng; Liu, Jun, and Wang, Yao. Motion Adaptive Pose Estimation from Compressed Videos. In *ICCV*, 2021. (Cited on pages 18 and 19.)
- Fang, Hao-Shu; Li, Jiefeng; Tang, Hongyang; Xu, Chao; Zhu, Haoyi; Xiu, Yuliang; Li, Yong-Lu, and Lu, Cewu. AlphaPose: Whole-Body Regional Multi-Person Pose Estimation and Tracking in Real-Time. In *arXiv-Preprint*, 2022. (Cited on page 12.)
- Fang, Haoshu; Sun, Jianhua; Wang, Runzhong; Gou, Minghao; Li, Yong-Lu, and Lu, Cewu. Instaboost: Boosting instance segmentation via probability map guided copy-pasting. *ICCV*, 2019. (Cited on page 30.)
- Fang, Kuan; Xiang, Yu; Li, Xiaocheng, and Savarese, Silvio. Recurrent Autoregressive Networks for Online Multi-object Tracking. *WACV*, 2017. (Cited on pages 28 and 29.)
- Feichtenhofer, Christoph; Pinz, Axel, and Zisserman, Andrew. Detect to Track and Track to Detect. In *ICCV*, 2017. (Cited on pages 29 and 30.)
- Felzenszwalb, Pedro F. and Huttenlocher, Daniel P. Pictorial Structures for Object Recognition. *IJCV*, 2005. (Cited on pages 8 and 68.)
- Feng, Runyang; Gao, Yixing; Ma, Xueqing; Tse, Tze Ho Elden, and Chang, Hyung Jin. Mutual Information-Based Temporal Difference Learning for Human Pose Estimation in Video. In *CVPR*, 2023. (Cited on pages 18 and 55.)
- Ferrari, Vittorio; Marín-Jiménez, Manuel, and Zisserman, Andrew. 2D Human Pose Estimation in TV Shows. In *Statistical and Geometrical Approaches to Visual Motion Analysis*, 2009. (Cited on page 8.)
- Fiaz, Mustansar; Cholakkal, Hisham; Anwar, Rao Muhammad, and Khan, Fahad Shahbaz. Sat: Scale-augmented transformer for person search. In *WACV*, 2023. (Cited on page 36.)
- Fischler, M. A. and Elschlager, R. A. The Representation and Matching of Pictorial Structures. *TC*, 1973. (Cited on page 8.)
- Fu, Yang; Liu, Sifei; Iqbal, Umar; Mello, Shalini De; Shi, Humphrey, and Kautz, Jan. Learning to Track Instances without Video Annotations. *CVPR*, 2021. (Cited on pages 21, 22, 108 and 118.)
- Fu, Zeyu; Angelini, Federico; Naqvi, Syed Mohsen, and Chambers, Jonathon A. GM-PHD Filter Based Online Multiple Human Tracking Using Deep Discriminative Correlation Matching. *ICASSP*, 2018. (Cited on page 28.)
- Fukushima, Kunihiro. Visual feature extraction by a multilayered network of analog threshold elements. *IEEE Trans. Syst. Sci. Cybern.*, 1969. (Cited on page 42.)

- Fukushima, Kunihiro. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 1980. (Cited on page 47.)
- Gao, Cunyuan; Yao, Rui; Zhao, Jiaqi; Zhou, Yong; Hu, Fuyuan, and Li, Leida. Structure-aware person search with self-attention and online instance aggregation matching. *Neurocomputing*, 2019. (Cited on page 35.)
- Gao, Shang; Wang, Jingya; Lu, Huchuan, and Liu, Zimo. Pose-guided Visible Part Matching for Occluded Person ReID. In *CVPR*, 2020. (Cited on page 33.)
- Ge, Zheng; Liu, Songtao; Wang, Feng; Li, Zeming, and Sun, Jian. YOLOX: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. (Cited on pages 25 and 32.)
- Girbau, Andreu; Marques, Ferran, and Satoh, Shin'ichi. Multiple Object Tracking from appearance by hierarchically clustering tracklets. In *BMVC*, 2022. (Cited on pages 28 and 29.)
- Girdhar, Rohit; Gkioxari, Georgia; Torresani, Lorenzo; Paluri, Manohar, and Tran, Du. Detect-and-Track: Efficient Pose Estimation in Videos. In *CVPR*, 2018. (Cited on pages 2, 20, 21, 24, 75, 82 and 109.)
- Girshick, Ross. Fast R-CNN. In *ICCV*, 2015. (Cited on pages 14 and 25.)
- Girshick, Ross; Donahue, Jeff; Darrell, Trevor, and Malik, Jitendra. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. (Cited on page 25.)
- Gkioxari, Georgia; Toshev, Alexander, and Jaitly, Navdeep. Chained Predictions Using Convolutional Neural Networks. In *ECCV*, 2016. (Cited on pages 18 and 19.)
- Glorot, Xavier; Bordes, Antoine, and Bengio, Yoshua. Deep sparse rectifier neural networks. In Gordon, Geoffrey; Dunson, David, and Dudík, Miroslav, editors, *PMLR*, 2011. (Cited on page 42.)
- Gong, Tao; Chen, Kai; Wang, Xinjiang; Chu, Q.; Zhu, Feng; Lin, Dahua; Yu, Nenghai, and Feng, Huamin. Temporal roi align for video object recognition. In *AAAI*, 2021. (Cited on page 32.)
- Goodfellow, Ian; Bengio, Yoshua, and Courville, Aaron. *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>. (Cited on pages 45, 47 and 48.)
- Gray, Douglas; Brennan, Shane, and Tao, Hai. Evaluating appearance models for recognition, reacquisition, and tracking. In *PETS*, 2007. (Cited on page 61.)
- Gu, Kerui; Yang, Linlin, and Yao, Angela. Removing the Bias of Integral Pose Regression. In *ICCV*, 2021. (Cited on pages 10, 11 and 12.)
- Guo, Hengkai; Tang, Tang; Luo, Guozhong; Chen, Riwei; Lu, Yongchen, and Wen, Linfu. Multi-Domain Pose Network for Multi-Person Pose Estimation and Tracking. In *CVPR*, 2018. (Cited on pages 21, 22, 23, 58, 68 and 82.)

- Han, Byeong-Ju; Ko, Kuhyeun, and Sim, Jae-Young. End-to-end trainable trident person search network using adaptive gradient propagation. In *ICCV*, 2021a. (Cited on pages 35 and 36.)
- Han, Chuchu; Ye, Jiacheng; Zhong, Yunshan; Tan, Xin; Zhang, Chi; Gao, Changxin, and Sang, Nong. Re-id driven localization refinement for person search. *ICCV*, 2019. (Cited on page 35.)
- Han, Chuchu; Su, Kai; Yu, Dongdong; Yuan, Zehuan; Gao, Changxin; Sang, Nong; Yang, Yi, and Wang, Changhu. Weakly supervised person search with region siamese networks. *ICCV*, 2021b. (Cited on page 36.)
- Han, Shoudong; Huang, Piao; Wang, Hongwei; Yu, En; Liu, Donghaisheng, and Pan, Xiaofeng. MAT: Motion-Aware Multi-Object Tracking. *Neurocomput.*, 2022. (Cited on page 27.)
- Haoshu Fang, Shuqin Xie and Lu, Cewu. RMPE: Regional Multi-person Pose Estimation. *ICCV*, 2016. (Cited on pages 10 and 24.)
- He, Kaiming and Sun, Jian. Convolutional neural networks at constrained time cost. *CVPR*, 2015. (Cited on page 54.)
- He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing, and Sun, Jian. Deep Residual Learning for Image Recognition. In *CVPR*, 2016. (Cited on pages 10, 14, 28, 32, 33, 36, 48, 53, 54, 58, 99, 100, 114 and 121.)
- He, Kaiming; Gkioxari, Georgia; Dollár, Piotr, and Girshick, Ross. Mask R-CNN. In *ICCV*, 2017. (Cited on pages 21 and 25.)
- He, Kaiming; Girshick, Ross B., and Dollár, Piotr. Rethinking imagenet pre-training. *ICCV*, 2019a. (Cited on page 53.)
- He, Shuting; Luo, Hao; Wang, Pichao; Wang, Fan; Li, Hao, and Jiang, Wei. Transreid: Transformer-based object re-identification. In *CVPR*, 2021. (Cited on pages 34 and 119.)
- He, Zhen; Li, Jian; Liu, Daxue; He, Hangen, and Barber, David. Tracking by Animation: Unsupervised Learning of Multi-Object Attentive Trackers. In *CVPR*, 2019b. (Cited on page 30.)
- Hejazi, Seyed Mostafa and Abhayaratne, Charith. Handcrafted localized phase features for human action recognition. *Image and Vision Computing*, 2022. (Cited on page 7.)
- Hendrycks, Dan and Gimpel, Kevin. Gaussian Error Linear Units (GELUs). *arXiv-Preprint*, 2016. (Cited on pages 113 and 115.)
- Hermans, Alexander; Beyer, Lucas, and Leibe, Bastian. In Defense of the Triplet Loss for Person Re-Identification. *arXiv-Preprint*, 2017. (Cited on pages 33, 58, 88, 96, 99 and 115.)
- Hirzer, Martin; Beleznaï, Csaba; Roth, Peter M., and Bischof, Horst. Person Re-identification by Descriptive and Discriminative Classification. In *Image Analysis*, 2011. (Cited on page 24.)
- Hochreiter, Sepp and Schmidhuber, Jürgen. Long Short-Term Memory. *Neural Comput.*, 1997. (Cited on pages 29, 31 and 42.)

- Hogg, David. Model-based vision: a program to see a walking person. *Image and Vision Computing*, 1983. (Cited on pages 7 and 10.)
- Hornáková, Andrea; Henschel, Roberto; Rosenhahn, Bodo, and Swoboda, Paul. Lifted Disjoint Paths with Application in Multiple Object Tracking. In *ICML*, 2020. (Cited on page 32.)
- Hou, Ruibing; Chang, Hong; Ma, Bingpeng; Shan, Shiguang, and Chen, Xilin. Temporal Complementary Learning for Video Person Re-Identification. In *ECCV*, 2020a. (Cited on page 88.)
- Hou, Ruibing; Ma, Bingpeng; Chang, Hong; Gu, Xinqian; Shan, Shiguang, and Chen, Xilin. Feature Completion for Occluded Person Re-Identification. *TPAMI*, 2020b. (Cited on pages 34 and 119.)
- Hu, Jie; Shen, Li, and Sun, Gang. Squeeze-and-Excitation Networks. In *CVPR*, 2018. (Cited on page 15.)
- Huang, Gao; Liu, Zhuang; van der Maaten, Laurens, and Weinberger, Kilian Q. Densely Connected Convolutional Networks. In *CVPR*, 2017. (Cited on page 11.)
- Huang, Junjie; Zhu, Zheng; Guo, Feng, and Huang, Guan. The Devil Is in the Details: Delving Into Unbiased Data Processing for Human Pose Estimation. In *CVPR*, 2020. (Cited on page 12.)
- Hwang, Jihye; Lee, Jieun; Park, Sungheon, and Kwak, Nojun. Pose estimator and tracker using temporal flow maps for limbs. In *IJCNN*, 2019. (Cited on pages 21, 22, 83, 108, 109 and 118.)
- Insafutdinov, Eldar; Pishchulin, Leonid; Andres, Bjoern; Andriluka, Mykhaylo, and Schiele, Bernt. DeeperCut: A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model. In *ECCV*, 2016. (Cited on pages 10, 14, 20 and 61.)
- Insafutdinov, Eldar; Andriluka, Mykhaylo; Pishchulin, Leonid; Tang, Siyu; Levinkov, Evgeny; Andres, Bjoern, and Schiele, Bernt. ArtTrack: Articulated Multi-person Tracking in the Wild. In *CVPR*, 2017. (Cited on pages 2, 20, 21, 57, 68 and 129.)
- Ioffe, Sergey and Szegedy, Christian. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *ICML*, 2015. (Cited on pages 46 and 115.)
- Iqbal, Umar; Garbade, Martin, and Gall, Juergen. Pose for Action - Action for Pose. *FG 2017*, 2016. (Cited on pages 7, 10, 19 and 20.)
- Iqbal, Umar; Milan, Anton, and Gall, Juergen. PoseTrack: Joint Multi-Person Pose Estimation and Tracking. In *CVPR*, 2017. (Cited on pages 2, 3, 20, 21, 57, 68, 88, 89, 90 and 129.)
- Izadinia, Hamid; Saleemi, Imran; Li, Wenhui, and Shah, Mubarak. (MP)2T: Multiple People Multiple Parts Tracker. In *ECCV*, 2012. (Cited on page 20.)
- Jaffe, Lucas and Zakhov, Avideh. Gallery filter network for person search. In *WACV*, 2023. (Cited on page 36.)
- Jain, Arjun; Tompson, Jonathan; LeCun, Yann, and Bregler, Christoph. MoDeep: A deep learning framework using motion features for human pose estimation. In *ACCV*, 2015. (Cited on pages 17 and 18.)

- Jin, Sheng; Ma, Xujie; Han, Zhipeng; Wu, Yue; Yang, Wei; Liu, Wentao; Qian, Chen, and Ouyang, Wanli. Towards Multi-Person Pose Tracking : Bottom-up and Top-down Methods. In *ICCVW*, 2017. (Cited on page 20.)
- Jin, Sheng; Liu, Wentao; Ouyang, Wanli, and Qian, Chen. Multi-person Articulated Tracking with Spatial and Temporal Embeddings. *CVPR*, 2019a. (Cited on page 57.)
- Jin, Sheng; Liu, Wentao; Ouyang, Wanli, and Qian, Chen. Multi-person Articulated Tracking with Spatial and Temporal Embeddings. In *CVPR*, 2019b. (Cited on pages 20, 21, 22, 57 and 83.)
- Jin, Sheng; Liu, Wentao; Xie, Enze; Wang, Wenhai; Qian, Chen; Ouyang, Wanli, and Luo, Ping. Differentiable Hierarchical Graph Grouping for Multi-Person Pose Estimation. In *ECCV*, 2020. (Cited on page 15.)
- Johansson, Gunnar. Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics*, 1973. (Cited on pages 1, 3, 7 and 19.)
- Johnson, S. and Everingham, M. Learning Effective Human Pose Estimation from Inaccurate Annotation. In *CVPR*, 2011. (Cited on pages 8 and 9.)
- Johnson, Sam and Everingham, Mark. Clustered Pose and Nonlinear Appearance Models for Human Pose Estimation. In *BMVC*, 2010. (Cited on page 8.)
- Kalman, Rudolph Emil and Others, . A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 1960. (Cited on page 26.)
- Kearns, M. Thoughts on Hypothesis Boosting. Unpublished, 1988. (Cited on page 10.)
- Khiredkar, Rawal; Chari, Visesh; Agrawal, Amit, and Tyagi, Ambrish. Multi-instance pose networks: Rethinking top down pose estimation. In *ICCV*, 2021. (Cited on page 12.)
- Kim, Chanho; Li, Fuxin, and Rehg, James M. Multi-object Tracking with Neural Gating Using Bilinear LSTM. In *ECCV*, 2018. (Cited on pages 28 and 29.)
- Kim, Yoon; Denton, Carl; Hoang, Luong, and Rush, Alexander M. Structured attention networks. *ICLR*, 2017. (Cited on page 49.)
- Kingma, Diederik and Ba, Jimmy. Adam: A Method for Stochastic Optimization. *ICLR*, 12 2014. (Cited on page 45.)
- Kocabas, Muhammed; Karagoz, Salih, and Akbas, Emre. MultiPoseNet: Fast Multi-Person Pose Estimation using Pose Residual Network. In *ECCV*, 2018. (Cited on pages 15, 68 and 73.)
- Kong, Tao; Yao, Anbang; Chen, Yurong, and Sun, Fuchun. HyperNet: Towards Accurate Region Proposal Generation and Joint Object Detection. *CVPR*, 2016. (Cited on page 11.)
- Kreiss, Sven; Bertoni, Lorenzo, and Alahi, Alexandre. PifPaf: Composite Fields for Human Pose Estimation. In *CVPR*, 2019. (Cited on pages 14 and 53.)

- Kriesel, David. *A brief introduction to Neural Networks*. 2007. Available at <http://www.dkriesel.com>, version Zeta2. (Cited on pages 40 and 41.)
- Krizhevsky, Alex; Sutskever, Ilya, and Hinton, Geoffrey E. ImageNet Classification with Deep Convolutional Neural Networks. In *NIPS*, 2012. (Cited on pages 47 and 53.)
- Kuhn, Harold W. The Hungarian Method for the Assignment Problem. *NRL*, 1955. (Cited on pages 2, 20, 26, 57, 95 and 130.)
- Kumar, A.; Yaghoubi, E.; Das, A.; Harish, B., and Proença, H. The P-DESTRE: A Fully Annotated Dataset for Pedestrian Detection, Tracking and Short/Long-term Re-Identification from Aerial Devices. *IEEE Transactions on Information Forensics and Security*, 2020. (Cited on pages 89, 90 and 91.)
- Ladicky, Lubor; Torr, Philip H. S., and Zisserman, Andrew. Human Pose Estimation Using a Joint Pixel-wise and Part-wise Formulation. In *CVPR*, 2013. (Cited on page 10.)
- Lan, Xu; Zhu, Xiatian, and Gong, Shaogang. Person Search by Multi-Scale Matching. *ECCV*, 2018. (Cited on pages 34 and 36.)
- Law, Hei and Deng, Jia. Cornernet: Detecting objects as paired keypoints. In *ECCV*, 2018. (Cited on pages 25 and 26.)
- Le Cun, Y.; Jackel, L.D.; Boser, B.; Denker, J.S.; Graf, H.P.; Guyon, I.; Henderson, D.; Howard, R.E., and Hubbard, W. Handwritten digit recognition: applications of neural network chips and automatic learning. *IEEE Communications Magazine*, 1989. (Cited on page 47.)
- Leal-Taixé, L.; Milan, A.; Reid, I.; Roth, S., and Schindler, K. MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking. *arXiv-Preprint*, 2015. (Cited on pages 88, 89, 90, 93, 96 and 98.)
- Leal-Taixé, L.; Canton-Ferrer, C., and Schindler, K. Learning by tracking: Siamese CNN for robust target association. *CVPR Workshops*, 2016. (Cited on page 33.)
- Lecun, Y.; Bottou, L.; Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998. (Cited on pages 40, 45 and 47.)
- Lee, Sanghoon; Oh, Youngmin; Baek, Donghyeon; Lee, Junghyup, and Ham, Bumsub. Oimnet++: Prototypical normalization and localization-aware learning for person search. In *ECCV*, 2022. (Cited on page 35.)
- Li, Dengjie; Chen, Siyu; Zhong, Yujie; Liang, Fan, and Ma, Lin. DiP: Learning Discriminative Implicit Parts for Person Re-Identification. In *arXiv-Preprint*, 2022a. (Cited on pages 33, 34 and 119.)
- Li, Jia; Su, Wen, and Wang, Zengfu. Simple Pose: Rethinking and Improving a Bottom-up Approach for Multi-Person Pose Estimation. *AAAI*, 2020a. (Cited on page 14.)

- Li, Jiahe; Gao, Xu, and Jiang, Tingting. Graph networks for multiple object tracking. *WACV*, 2020b. (Cited on page 33.)
- Li, Jiefeng; Wang, Can; Zhu, Hao; Mao, Yihuan; Fang, Hao-Shu, and Lu, Cewu. CrowdPose: Efficient Crowded Scenes Pose Estimation and A New Benchmark. *CVPR*, 2019a. (Cited on page 12.)
- Li, Jiefeng; Bian, Siyuan; Zeng, Ailing; Wang, Can; Pang, Bo; Liu, Wentao, and Lu, Cewu. Human Pose Regression with Residual Log-likelihood Estimation. In *ICCV*, 2021a. (Cited on pages 10, 11, 12 and 13.)
- Li, Junjie; Yan, Yichao; Wang, Guanshuo; Yu, Fufu; Jia, Qiong, and Ding, Shouhong. Domain adaptive person search. In *ECCV*, 2022b. (Cited on page 36.)
- Li, Ke; Wang, Shijie; Zhang, Xiang; Xu, Yifan; Xu, Weijian, and Tu, Zhuowen. Pose recognition with cascade transformers. In *CVPR*, 2021b. (Cited on page 13.)
- Li, Wei and Wang, Xiaogang. Locally Aligned Feature Transforms across Views. In *CVPR*, 2013. (Cited on page 24.)
- Li, Wen; Zou, Cheng; Wang, Meng; Xu, Furong; Zhao, Jianan; Zheng, Ruobing; Cheng, Yuan, and Chu, Wei. DC-Former: Diverse and Compact Transformer for Person Re-Identification. *AAAI*, 2023. (Cited on pages 34 and 119.)
- Li, Wenbo; Wang, Zhicheng; Yin, Binyi; Peng, Qixiang; Du, Yuming; Xiao, Tianzi; Yu, Gang; Lu, Hongtao; Wei, Yichen, and Sun, Jian. Rethinking on Multi-Stage Networks for Human Pose Estimation. *arXiv preprint*, 2019b. (Cited on pages 11 and 68.)
- Li, Yanjie; Zhang, Shoukui; Wang, Zhicheng; Yang, Sen; Yang, Wankou; Xia, Shu-Tao, and Zhou, Erjin. TokenPose: Learning Keypoint Tokens for Human Pose Estimation. In *iccv*, 2021c. (Cited on page 13.)
- Li, Yulin; He, Jianfeng; Zhang, Tianzhu; Liu, Xiang; Zhang, Yongdong, and Wu, Feng. Diverse part discovery: Occluded person re-identification with part-aware transformer. In *CVPR*, 2021d. (Cited on pages 34 and 119.)
- Li, Zhengjia and Miao, Duoqian. Sequential End-to-end Network for Efficient Person Search. In *AAAI*, 2021. (Cited on pages vii, 35, 36, 60, 88, 95, 100 and 103.)
- Liang, Chao; Zhang, Zhipeng; Zhou, Xue; Li, Bing; Zhu, Shuyuan, and Hu, Weiming. Rethinking the Competition Between Detection and ReID in Multiobject Tracking. *IEEE Transactions on Image Processing*, 2022. (Cited on page 29.)
- Lin, Tsung-Yi; Maire, Michael; Belongie, Serge; Hays, James; Perona, Pietro; Ramanan, Deva; Dollár, Piotr, and Zitnick, C. Microsoft COCO: Common Objects in Context. In *ECCV*, 2014. (Cited on pages 2, 13, 61, 68, 69, 70, 76, 98 and 119.)

- Lin, Tsung-Yi; Dollár, Piotr; Girshick, Ross B.; He, Kaiming; Hariharan, Bharath, and Belongie, Serge J. Feature Pyramid Networks for Object Detection. In *CVPR*, 2017a. (Cited on pages 11, 25, 98, 102 and 119.)
- Lin, Tsung-Yi; Goyal, Priya; Girshick, Ross; He, Kaiming, and Dollár, Piotr. Focal Loss for Dense Object Detection. In *ICCV*, 2017b. (Cited on pages 15 and 25.)
- Lin, Xiangtan; Ren, Pengzhen; Xiao, Yun; Chang, Xiaojun, and Hauptmann, Alexander G. Person search challenges and solutions: A survey. *ArXiv-Preprint*, 2021. (Cited on page 35.)
- Liu, Hao; Feng, Jiashi; Jie, Zequn; Karlekar, Jayashree; Zhao, Bo; Qi, Meibin; Jiang, Jianguo, and Yan, Shuicheng. Neural person search machines. *ICCV*, 2017. (Cited on page 35.)
- Liu, Wei; Anguelov, Dragomir; Erhan, Dumitru; Szegedy, Christian; Reed, Scott; Fu, Cheng-Yang, and Berg, Alexander C. SSD: Single Shot MultiBox Detector. In *ECCV*, 2016. (Cited on page 25.)
- Liu, Ze; Lin, Yutong; Cao, Yue; Hu, Han; Wei, Yixuan; Zhang, Zheng; Lin, Stephen, and Guo, Baining. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In *ICCV*, 2021a. (Cited on pages 13 and 32.)
- Liu, Zhenguang; Chen, Haoming; Feng, Runyang; Wu, Shuang; Ji, Shouling; Yang, Bailin, and Wang, Xun. Deep Dual Consecutive Network for Human Pose Estimation. *CVPR*, 2021b. (Cited on pages 17 and 18.)
- Liu, Zhenguang; Feng, Runyang; Chen, Haoming; Wu, Shuang; Gao, Yixing; Gao, Yunjun, and Wang, Xiang. Temporal Feature Alignment and Mutual Information Maximization for Video-Based Human Pose Estimation. In *CVPR*, 2022. (Cited on pages 18 and 55.)
- Long, Chen; Haizhou, Ai; Zijie, Zhuang, and Chong, Shang. Real-time Multiple People Tracking with Deeply Learned Candidate Selection and Person Re-identification. In *ICME*, 2018. (Cited on pages 28, 29 and 30.)
- Long, Jonathan; Shelhamer, Evan, and Darrell, Trevor. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. (Cited on page 56.)
- Loshchilov, Ilya and Hutter, Frank. Decoupled Weight Decay Regularization. In *ICCV*, 2019. (Cited on pages 45 and 120.)
- Lowe, David G. Object Recognition from Local Scale-Invariant Features. In *ICCV*, 1999. (Cited on page 8.)
- Lu, Y.; Hong, Z.; Liu, B.; Li, W., and Yu, N. Dhff: Robust Multi-Scale Person Search by Dynamic Hierarchical Feature Fusion. *ICIP*, 2019. (Cited on page 36.)
- Lu, Zhichao; Rathod, Vivek; Votel, Ronny, and Huang, Jonathan. RetinaTrack: Online Single Stage Joint Detection and Tracking. *CVPR*, 2020. (Cited on page 29.)

- Luiten, Jonathon; Osep, Aljosa; Dendorfer, Patrick; Torr, Philip; Geiger, Andreas; Leal-Taixé, Laura, and Leibe, Bastian. HOTA: A Higher Order Metric for Evaluating Multi-Object Tracking. *IJCV*, 2020. (Cited on pages ix, 64, 65, 93, 94, 96, 98, 99, 118 and 128.)
- Luo, Chuanchen; Chen, Yuntao; Wang, Naiyan, and Zhang, Zhaoxiang. Spectral Feature Transformation for Person Re-identification. *ICCV*, 2019a. (Cited on page 88.)
- Luo, Hao; Gu, Youzhi; Liao, Xingyu; Lai, Shenqi, and Jiang, Wei. Bag of Tricks and a Strong Baseline for Deep Person Re-Identification. In *CVPRW*, 2019b. (Cited on pages vii, 28, 33, 53, 58, 59, 77, 115, 119, 121 and 122.)
- Luo, Zhengxiong; Wang, Zhicheng; Huang, Yan; Wang, Liang; Tan, Tieniu, and Zhou, Erjin. Rethinking the Heatmap Regression for Bottom-up Human Pose Estimation. In *CVPR*, 2021. (Cited on page 15.)
- Maas, Andrew L. Rectifier nonlinearities improve neural network acoustic models. In *ICML*, 2013. (Cited on page 42.)
- Mahmoudi, Nima; Ahadi, Seyed Mohammad, and Rahmati, Mohammad. Multi-target tracking using CNN-based features: CNNMTT. *Multimedia Tools and Applications*, 2019. (Cited on page 28.)
- Manen, S.; Gygli, M.; Dai, D., and Gool, L. V. PathTrack: Fast Trajectory Annotation with Path Supervision. In *ICCV*, 2017. (Cited on pages 90 and 91.)
- Mao, Weian; Tian, Zhi; Wang, Xinlong, and Shen, Chunhua. FCPose: Fully Convolutional Multi-Person Pose Estimation with Dynamic Instance-Aware Convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. (Cited on page 16.)
- Mao, Weian; Ge, Yongtao; Shen, Chunhua; Tian, Zhi; Wang, Xinlong; Wang, Zhibin, and Hengel, Anton van den. Poseur: Direct human pose regression with transformers. In *ECCV*, 2022. (Cited on pages 10 and 13.)
- Meinhardt, Tim; Kirillov, Alexander; Leal-Taixe, Laura, and Feichtenhofer, Christoph. TrackFormer: Multi-Object Tracking with Transformers. In *CVPR*, 2022. (Cited on pages 31 and 32.)
- Miao, Jiaxu; Wu, Yu; Liu, Ping; Ding, Yuhang, and Yang, Yi. Pose-Guided Feature Alignment for Occluded Person Re-Identification. In *ICCV*, 2019. (Cited on pages 33, 53 and 119.)
- Milan, A.; Roth, S., and Schindler, K. Continuous Energy Minimization for Multitarget Tracking. *TPAMI*, 2014. (Cited on page 32.)
- Milan, A.; Leal-Taixé, L.; Reid, I.; Roth, S., and Schindler, K. MOT16: A Benchmark for Multi-Object Tracking. *arXiv-Preprint*, 2016. (Cited on pages 29, 88, 89 and 90.)
- Munjal, Bharti; Amin, S.; Tombari, Federico, and Galasso, Fabio. Query-guided end-to-end person search. *CVPR*, 2019. (Cited on page 35.)
- Newell, Alejandro; Yang, Kaiyu, and Deng, Jia. Stacked Hourglass Networks for Human Pose Estimation. In *ECCV*, 2016. (Cited on pages 10, 11, 15 and 53.)

- Newell, Alejandro; Huang, Zhiao, and Deng, Jia. Associative Embedding: End-to-End Learning for Joint Detection and Grouping. In *NIPS*, 2017. (Cited on pages 9, 14, 15, 22 and 55.)
- Nguyen, Dat Tien; Hong, Hyung Gil; Kim, Ki Wan, and Park, Kang Ryoung. Person recognition system based on a combination of body images from visible light and thermal cameras. *Sensors*, 2017. (Cited on page 33.)
- Nie, Bruce Xiaohan; Xiong, Caiming, and Zhu, Song-Chun. Joint action recognition and pose estimation from video. In *CVPR*, 2015. (Cited on page 19.)
- Nie, Xuecheng; Feng, Jiashi; Xing, Junliang, and Yan, Shuicheng. Pose Partition Networks for Multi-Person Pose Estimation. In *ECCV*, 2018. (Cited on pages 15 and 68.)
- Nie, Xuecheng; Li, Yuncheng; Luo, Linjie; Zhang, Ning, and Feng, Jiashi. Dynamic Kernel Distillation for Efficient Pose Estimation in Videos. In *ICCV*, 2019a. (Cited on page 19.)
- Nie, Xuecheng; Zhang, Jianfeng; Yan, Shuicheng, and Feng, Jiashi. Single-Stage Multi-Person Pose Machines. *ICCV*, 2019b. (Cited on pages 9, 14, 16 and 55.)
- Nilsson, D. An Efficient Algorithm for Finding the M Most Probable Configurations in Probabilistic Expert Systems. *Statistics and Computing*, 1998. (Cited on page 10.)
- Ning, G. and He, Zhihai. Dual Path Networks for Multi-Person Human Pose Estimation. *ICCVW*, 2017. (Cited on page 14.)
- Ning, Guanghan and Huang, Heng. LightTrack: A Generic Framework for Online Top-Down Human Pose Tracking. *arXiv preprint*, 2019. (Cited on pages 21, 22, 23, 24, 83 and 128.)
- Ning, Guanghan; Liu, Ping; Fan, Xiaochuan, and Zhang, Chi. A Top-down Approach to Articulated Human Pose Estimation and Tracking. In *ECCV Workshop*, 2019. (Cited on pages 23 and 83.)
- Niyogi, Sourabh A. and Adelson, Edward H. Analyzing and Recognizing Walking Figures in XYT. In *CVPR*, 1994. (Cited on page 7.)
- OpenAI, . Gpt-4 technical report, 2023. (Cited on page 49.)
- O'Rourke, Joseph and Badler, Norman I. Model-based image analysis of human motion using constraint propagation. *TPAMI*, 1980. (Cited on page 7.)
- Ouyang, Wanli; Chu, Xiao, and Wang, Xiaogang. Multi-source Deep Learning for Human Pose Estimation. In *CVPR*, 2014. (Cited on pages 8 and 9.)
- Pang, Bo; Li, Yizhuo; Zhang, Yifan; Li, Muchen, and Lu, Cewu. TubeTK: Adopting Tubes to Track Multi-Object in a One-Step Training Model. In *CVPR*, 2020. (Cited on pages 29 and 30.)
- Pang, Jiangmiao; Qiu, Linlu; Li, Xia; Chen, Haofeng; Li, Qi; Darrell, Trevor, and Yu, Fisher. Quasi-Dense Similarity Learning for Multiple Object Tracking. In *CVPR*, 2021. (Cited on page 29.)

- Papandreou, George; Zhu, Tyler; Kanazawa, Nori; Toshev, Alexander; Tompson, Jonathan; Bregler, Chris, and Murphy, Kevin. Towards Accurate Multi-person Pose Estimation in the Wild. In *CVPR*, 2017. (Cited on pages 10, 11 and 24.)
- Papandreou, George; Zhu, Tyler; Chen, Liang-Chieh; Gidaris, Spyros; Tompson, Jonathan, and Murphy, Kevin. PersonLab: Person Pose Estimation and Instance Segmentation with a Bottom-Up, Part-Based, Geometric Embedding Model. In *ECCV*, 2018. (Cited on page 14.)
- Parikh, Ankur P.; Täckström, Oscar; Das, Dipanjan, and Uszkoreit, Jakob. A decomposable attention model for natural language inference. *ArXiv-Preprint*, 2016. (Cited on page 49.)
- Park, Dennis and Ramanan, Deva. N-best maximal decoders for part models. In *ICCV*, 2011. (Cited on page 10.)
- Park, Sungchan; You, Eunyi; Lee, Inhoe, and Lee, Joonseok. Towards robust and smooth 3d multi-person pose estimation from monocular videos in the wild. In *ICCV*, 2023. (Cited on page 130.)
- Park, Taesung; Liu, Ming-Yu; Wang, Ting-Chun, and Zhu, Jun-Yan. Semantic Image Synthesis With Spatially-Adaptive Normalization. In *CVPR*, 2019. (Cited on page 114.)
- Peng, Jinlong; Wang, Changan; Wan, Fangbin; Wu, Yang; Wang, Yabiao; Tai, Ying; Wang, Chengjie; Li, Jilin; Huang, Feiyue, and Fu, Yanwei. Chained-Tracker: Chaining Paired Attentive Regression Results for End-to-End Joint Multiple-Object Detection and Tracking. In *ECCV*, 2020. (Cited on pages 29 and 30.)
- Pfister, Tomas; Charles, James, and Zisserman, Andrew. Flowing convnets for human pose estimation in videos. In *ICCV*, 2015. (Cited on pages 17 and 18.)
- Pishchulin, Leonid; Jain, Arjun; Andriluka, Mykhaylo; Thormaehlen, Thorsten, and Schiele, Bernt. Articulated People Detection and Pose Estimation: Reshaping the Future. In *CVPR*, 2012. (Cited on page 10.)
- Pishchulin, Leonid; Andriluka, Micha; Gehler, Peter, and Schiele, Bernt. Poselet conditioned pictorial structures. In *CVPR*, 2013a. (Cited on page 8.)
- Pishchulin, Leonid; Andriluka, Mykhaylo; Gehler, Peter, and Schiele, Bernt. Strong Appearance and Expressive Spatial Models for Human Pose Estimation. In *ICCV*, 2013b. (Cited on pages 8 and 14.)
- Pishchulin, Leonid; Insafutdinov, Eldar; Tang, Siyu; Andres, Björn; Andriluka, Mykhaylo; Gehler, Peter, and Schiele, Bernt. DeepCut: Joint Subset Partition and Labeling for Multi Person Pose Estimation. In *CVPR*, 2016. (Cited on pages 10 and 14.)
- Polana, R. and Nelson, R. Detecting activities. In *CVPR*, 1993. (Cited on page 7.)
- Prilutsky, Boris I and Zatsiorsky, Vladimir M. Optimization-based models of muscle coordination. *Exerc. Sport Sci. Rev.*, 2002. (Cited on page 55.)

- Qi, Lei; Huo, Jing; Wang, Lei; Shi, Yinghuan, and Gao, Yang. MaskReID: A Mask Based Deep Ranking Neural Network for Person Re-identification. *ICME*, 2018. (Cited on page 33.)
- Qian, R.J. and Huang, T.S. Motion analysis of human ambulatory patterns. In *ICPR*, 1992. (Cited on page 7.)
- Qiu, Lingteng; Zhang, Xuanye; Li, Yanran; Li, Guanbin; Wu, Xiaojun; Xiong, Zixiang; Han, Xiaoguang, and Cui, Shuguang. Peeking into occluded joints: A novel framework for crowd pose estimation. In *ECCV*, 2020a. (Cited on pages 12 and 13.)
- Qiu, Zhongwei; Qiu, Kai; Fu, Jianlong, and Fu, Dongmei. DGCN: Dynamic Graph Convolutional Network for Efficient Multi-Person Pose Estimation. *AAAI*, 2020b. (Cited on page 14.)
- Raaj, Yaadhav; Idrees, Haroon; Hidalgo, Gines, and Sheikh, Yaser. Efficient Online Multi-Person 2D Pose Tracking With Recurrent Spatio-Temporal Affinity Fields. In *CVPR*, 2019. (Cited on pages 20, 21, 22, 57, 83, 108, 109, 118 and 128.)
- Rafi, Umer; Doering, Andreas; Leibe, Bastian, and Gall, Juergen. Self-supervised Keypoint Correspondences for Multi-Person Pose Estimation and Tracking in Videos. In *ECCV*, 2020. (Cited on pages 4, 67, 88, 93, 98, 108, 109 and 117.)
- Ramanan, Deva; Forsyth, David A., and Zisserman, Andrew. Strike a Pose: Tracking People by Finding Stylized Poses. In *CVPR*, 2005. (Cited on page 7.)
- Redmon, Joseph and Farhadi, Ali. Yolov3: An incremental improvement. *ArXiv*, 2018. (Cited on page 30.)
- Redmon, Joseph; Divvala, Santosh; Girshick, Ross, and Farhadi, Ali. You Only Look Once: Unified, Real-Time Object Detection. In *CVPR*, 2016. (Cited on pages 25 and 53.)
- Ren, Shaoqing; He, Kaiming; Girshick, Ross B., and Sun, Jian. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *NIPS*, 2015. (Cited on pages vii, 10, 20, 25, 26, 35, 36, 53, 55, 60, 94, 95, 98, 100, 102 and 119.)
- Ren, Xiaofeng; Berg, Alexander C., and Malik, Jitendra. Recovering Human Body Configurations Using Pairwise Constraints between Parts. In *ICCV*, 2005. (Cited on page 8.)
- Ristani, Ergys; Solera, Francesco; Zou, Roger S.; Cucchiara, R., and Tomasi, Carlo. Performance Measures and a Data Set for Multi-target, Multi-camera Tracking. In *ECCVW*, 2016a. (Cited on pages 88, 89, 90 and 91.)
- Ristani, Ergys; Solera, Francesco; Zou, Roger S.; Cucchiara, Rita, and Tomasi, Carlo. Performance Measures and a Data Set for Multi-Target, Multi-Camera Tracking. *ECCVW*, 2016b. (Cited on pages 63, 64 and 99.)
- Rohr, K. Incremental recognition of pedestrians from image sequences. In *CVPR*, 1993. (Cited on page 7.)

- Ronchi, Matteo Ruggero and Perona, Pietro. Benchmarking and error diagnosis in multi-instance pose estimation. *ICCV*, 2017. (Cited on page 61.)
- Ruan, Weijian; Liu, Wu; Bao, Qian; Chen, Jun; Cheng, Yuhao, and Mei, Tao. POINet: Pose-Guided Ovonc Insight Network for Multi-Person Pose Tracking. In *International Conference on Multimedia*, 2019. (Cited on pages 21, 22, 24 and 83.)
- Rumelhart, D. E.; Hinton, G. E., and Williams, R. J. Parallel distributed processing: Explorations in the microstructure of cognition. chapter Learning Internal Representations by Error Propagation. MIT Press, 1986. (Cited on page 45.)
- Sadeghian, Amir; Alahi, Alexandre, and Savarese, Silvio. Tracking the Untrackable: Learning to Track Multiple Cues With Long-Term Dependencies. In *ICCV*, 2017. (Cited on pages 28 and 29.)
- Sandler, Mark; Howard, Andrew; Zhu, Menglong; Zhmoginov, Andrey, and Chen, Liang-Chieh. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *CVPR*, 2018. (Cited on page 17.)
- Sapp, Benjamin; Weiss, David J., and Taskar, Ben. Parsing human motion with stretchable models. *CVPR*, 2011. (Cited on page 60.)
- Sarfraz, M. Saquib; Schumann, Arne; Eberle, Andreas, and Stiefelhagen, Rainer. A Pose-Sensitive Embedding for Person Re-Identification with Expanded Cross Neighborhood Re-Ranking. *CVPR*, 2018. (Cited on pages 33 and 119.)
- Schapire, Robert E. A Brief Introduction to Boosting. In *IJCAI*, 1999. (Cited on page 8.)
- Seidenschwarz, Jenny; Brasó, Guillem; Serrano, Victor Castro; Elezi, Ismail, and Leal-Taixé, Laura. Simple cues lead to a strong multi-object tracker. In *CVPR*, 2023. (Cited on page 28.)
- Shi, Dahu; Wei, Xing; Yu, Xiaodong; Tan, Wenming; Ren, Ye, and Pu, Shiliang. InsPose: Instance-Aware Networks for Single-Stage Multi-Person Pose Estimation. In *MM*, 2021. (Cited on page 16.)
- Shi, Dahu; Wei, Xing; Li, Liangqi; Ren, Ye, and Tan, Wenming. End-to-End Multi-Person Pose Estimation with Transformers. In *CVPR*, 2022. (Cited on page 16.)
- Shuai, Bing; Li, Xinyu; Kundu, Kaustav, and Tighe, Joseph. Id-free person similarity learning. In *CVPR*, 2022. (Cited on pages 36 and 37.)
- Simonyan, Karen and Zisserman, Andrew. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Bengio, Yoshua and LeCun, Yann, editors, *ICLR*, 2015. (Cited on pages 22, 29, 53 and 56.)
- Singh, Bharat; Najibi, Mahyar, and Davis, Larry S. SNIPER: Efficient Multi-Scale Training. *NIPS*, 2018. (Cited on page 118.)
- Sinkhorn, Richard and Knopp, Paul. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 1967. (Cited on page 130.)

- Snower, Michael; Kadav, Asim; Lai, Farley, and Graf, Hans Peter. 15 Keypoints Is All You Need. *arxiv-preprint*, 2019. (Cited on pages 17, 18, 20, 21, 22, 23, 24, 88, 98, 108, 109, 117 and 118.)
- Somers, Vladimir; De Vleeschouwer, Christophe, and Alahi, Alexandre. Body Part-Based Representation Learning for Occluded Person Re-Identification. *WACV23*, 2023. (Cited on pages 33 and 119.)
- Son, Jeany; Baek, Mooyeol; Cho, Minsu, and Han, Bohyung. Multi-Object Tracking With Quadruplet Convolutional Neural Networks. In *CVPR*, 2017. (Cited on page 33.)
- Song, Jie; Wang, Limin; Van Gool, Luc, and Hilliges, Otmar. Thin-slicing network: A deep structured model for pose estimation in videos. In *CVPR*, 2017. (Cited on pages 17 and 18.)
- Srivastav, Vinkle; Chen, Keqi, and Padoy, Nicolas. Selfpose3d: Self-supervised multi-person multi-view 3d pose estimation. In *CVPR*, 2024. (Cited on page 130.)
- Srivastava, Rupesh Kumar; Greff, Klaus, and Schmidhuber, Jürgen. Highway networks. *ArXiv-Preprint*, 2015. (Cited on page 54.)
- Stadler, Daniel and Beyerer, Jürgen. Modelling Ambiguous Assignments for Multi-Person Tracking in Crowds. In *WACVW*, 2022. (Cited on page 27.)
- Steinwart, Ingo and Christmann, Andreas. *Support Vector Machines*. Springer Publishing Company, 2008. ISBN 0387772413. (Cited on page 8.)
- Su, Chi; Zhang, Shiliang; Xing, Junliang; Gao, Wen, and Tian, Qi. Deep attributes driven multi-camera person re-identification. In *ECCV*, 2016. (Cited on page 33.)
- Su, Jiajun; Wang, Chunyu; Ma, Xiaoxuan; Zeng, Wenjun, and Wang, Yizhou. Virtualpose: Learning generalizable 3d human pose models from virtual data. *ECCV*, 2022. (Cited on page 130.)
- Su, Kai; Yu, Dongdong; Xu, Zhenqi; Geng, Xin, and Wang, Changhu. Multi-Person Pose Estimation with Enhanced Channel-wise and Spatial Information. *CVPR*, 2019. (Cited on page 11.)
- Subramaniam, Arulkumar; Nambiar, Athira, and Mittal, Anurag. Co-segmentation inspired attention networks for video-based person re-identification. In *ICCV*, 2019. (Cited on page 33.)
- Sun, Deqing; Yang, Xiaodong; Liu, Ming-Yu, and Kautz, Jan. PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume. In *CVPR*, 2018a. (Cited on page 77.)
- Sun, Ke; Xiao, Bin; Liu, Dong, and Wang, Jingdong. Deep High-Resolution Representation Learning for Human Pose Estimation. In *CVPR*, 2019a. (Cited on pages 7, 9, 10, 11, 13, 15, 21, 24, 48, 53, 68, 75, 82, 83, 128 and 130.)
- Sun, Peize; Cao, Jinkun; Jiang, Yi; Zhang, Rufeng; Xie, Enze; Yuan, Zehuan; Wang, Changhu, and Luo, Ping. TransTrack: Multiple-Object Tracking with Transformer. *ArXiv-Preprint*, 2020. (Cited on page 31.)

- Sun, ShiJie; Akhtar, Naveed; Song, HuanSheng; Mian, Ajmal, and Shah, Mubarak. Deep Affinity Network for Multiple Object Tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021. (Cited on pages 28 and 29.)
- Sun, Xiao; Xiao, Bin; Liang, Shuang, and Wei, Yichen. Integral human pose regression. *ECCV*, 2018b. (Cited on pages 10, 11 and 12.)
- Sun, Yifan; Zheng, Liang; Yang, Yi; Tian, Qi, and Wang, Shengjin. Beyond Part Models: Person Retrieval with Refined Part Pooling (and A Strong Convolutional Baseline). In *ECCV*, 2018c. (Cited on pages 33, 35, 58 and 119.)
- Sun, Yifan; Xu, Qin; Li, Yali; Zhang, Chi; Li, Yikang; Wang, Shengjin, and Sun, Jian. Perceive Where to Focus: Learning Visibility-aware Part-level Features for Partial Person Re-identification. *CVPR*, 2019b. (Cited on page 33.)
- Szegedy, Christian; Liu, Wei; Jia, Yangqing; Sermanet, Pierre; Reed, Scott; Anguelov, Dragomir; Erhan, Dumitru; Vanhoucke, Vincent, and Rabinovich, Andrew. Going Deeper with Convolutions. In *CVPR*, 2015. (Cited on pages viii, 28, 48, 53, 54, 70, 71, 72 and 95.)
- Szegedy, Christian; Vanhoucke, Vincent; Ioffe, Sergey; Shlens, Jon, and Wojna, Zbigniew. Rethinking the Inception Architecture for Computer Vision. In *CVPR*, 2016. (Cited on pages 53, 54 and 115.)
- Tang, Siyu; Andres, Bjoern; Andriluka, Mykhaylo, and Schiele, Bernt. Subgraph Decomposition for Multi-Target Tracking. In *CVPR*, 2015. (Cited on page 32.)
- Tang, Siyu; Andriluka, Mykhaylo; Andres, Bjoern, and Schiele, Bernt. Multiple People Tracking by Lifted Multicut and Person Re-Identification. In *CVPR*, 2017. (Cited on pages 28 and 32.)
- Tian, Kun; Huang, Houjing; Ye, Yun; Li, Shiyu; Lin, Jinbin, and Huang, Guan. End-to-end thorough body perception for person search. *AAAI*, 2020. (Cited on pages 35 and 36.)
- Tian, Zhi; Shen, Chunhua; Chen, Hao, and He, Tong. Fcos: Fully convolutional one-stage object detection. In *ICCV*, 2019. (Cited on pages 25 and 26.)
- Tokmakov, Pavel; Li, Jie; Burgard, Wolfram, and Gaidon, Adrien. Learning to track with object permanence. *ICCV*, 2021. (Cited on page 30.)
- Tompson, Jonathan J; Jain, Arjun; LeCun, Yann, and Bregler, Christoph. Joint Training of a Convolutional Network and a Graphical Model for Human Pose Estimation. In *NIPS*, 2014. (Cited on page 8.)
- Toshev, Alexander and Szegedy, Christian. DeepPose: Human Pose Estimation via Deep Neural Networks. In *CVPR*, 2014. (Cited on pages 8 and 9.)
- Tripathi, Shashank; Müller, Lea; Huang, Chun-Hao P.; Taheri, Omid; Black, Michael, and Tzionas, Dimitrios. 3D human pose estimation via intuitive physics. In *CVPR*, 2023. (Cited on page 3.)

- Tu, Hanyue; Wang, Chunyu, and Zeng, Wenjun. Voxelpose: Towards multi-camera 3d human pose estimation in wild environment. In *ECCV*, 2020. (Cited on page 130.)
- Ukita, Norimichi. Iterative Action and Pose Recognition Using Global-and-Pose Features and Action-Specific Models. In *ICCVW*, 2013. (Cited on page 19.)
- Ullah, Mohib and Cheikh, Faouzi Alaya. Deep Feature Based End-to-End Transportation Network for Multi-Target Tracking. *ICIP*, 2018. (Cited on page 28.)
- Ulyanov, Dmitry; Vedaldi, Andrea, and Lempitsky, Victor S. Instance normalization: The missing ingredient for fast stylization. *arXiv-Preprint*, 2016. (Cited on page 46.)
- Vaswani, Ashish; Shazeer, Noam; Parmar, Niki; Uszkoreit, Jakob; Jones, Llion; Gomez, Aidan N; Kaiser, Łukasz, and Polosukhin, Illia. Attention is All you Need. In *NIPS*, 2017. (Cited on pages vii, 13, 31, 49, 50, 51, 111, 112 and 115.)
- Voigtlaender, Paul; Krause, Michael; Osep, Aljosa; Luiten, Jonathon; Sekar, Berin Balachandrar Gnana; Geiger, Andreas, and Leibe, Bastian. MOTs: Multi-Object Tracking and Segmentation. In *CVPR*, 2019. (Cited on pages 29 and 30.)
- Wang, Benzhi; Yang, Yang; Wu, Jinlin; Qi, Guo-jun, and Lei, Zhen. Self-similarity driven scale-invariant learning for weakly supervised person search. In *ICCV*, 2023a. (Cited on pages 36 and 37.)
- Wang, Cheng; Ma, Bingpeng; Chang, Hong; Shan, S., and Chen, Xilin. Tcts: A task-consistent two-stage framework for person search. *CVPR*, 2020a. (Cited on page 35.)
- Wang, Chien-Yao; Bochkovskiy, Alexey, and Liao, Hong-Yuan Mark. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *CVPR*, 2023b. (Cited on page 25.)
- Wang, Dongkai; Zhang, Shiliang, and Hua, Gang. Robust Pose Estimation in Crowded Scenes with Direct Pose-Level Inference. In *NeurIPS*, 2021a. (Cited on pages 16 and 17.)
- Wang, Fang and Li, Yi. Beyond physical connections: Tree models in human pose estimation. In *CVPR*, 2013. (Cited on page 9.)
- Wang, Gaoang; Gu, Renshu; Liu, Zuozhu; Hu, Weijie; Song, Mingli, and Hwang, Jenq-Neng. Track without Appearance: Learn Box and Tracklet Embedding with Local and Global Motion Patterns for Vehicle Tracking. *ICCV*, 2021b. (Cited on page 27.)
- Wang, Guan'an; Yang, Shuo; Liu, Huanyu; Wang, Zhicheng; Yang, Yang; Wang, Shuliang; Yu, Gang; Zhou, Erjin, and Sun, Jian. High-order information matters: Learning relation and topology for occluded person re-identification. In *CVPR*, 2020b. (Cited on pages 33, 34 and 119.)
- Wang, Manchen; Tighe, Joseph, and Modolo, Davide. Combining detection and tracking for human pose estimation in videos. In *CVPR*, 2020c. (Cited on pages 7, 21, 22, 24, 55, 88, 98, 108, 109, 117 and 118.)

- Wang, Pengfei; Ding, Changxing; Shao, Zhiyin; Hong, Zhibin; Zhang, Sheng, and Tao, Dacheng. Quality-aware part models for occluded person re-identification. *IEEE Transactions on Multimedia*, 2022a. (Cited on page 34.)
- Wang, Qiang; Zheng, Yun; Pan, Pan, and Xu, Yinghui. Multiple object tracking with correlation learning. *CVPR*, 2021c. (Cited on page 30.)
- Wang, Shuai; Sheng, Hao; Zhang, Yang; Wu, Yubin, and Xiong, Zhang. A general recurrent tracking framework without real data. In *ICCV*, 2021d. (Cited on page 31.)
- Wang, Tao; Zhang, Jianfeng; Cai, Yujun; Yan, Shuicheng, and Feng, Jiashi. Direct multi-view multi-person 3d human pose estimation. *NeurIPS*, 2021e. (Cited on page 130.)
- Wang, Tao; Liu, Hong; Song, Pinhao; Guo, Tianyu, and Shi, Wei. Pose-guided feature disentangling for occluded person re-identification based on transformer. In *AAAI*, 2022b. (Cited on pages 34 and 119.)
- Wang, Yihan; Li, Muiyang; Cai, Han; Chen, Wei-Ming, and Han, Song. Lite pose: Efficient architecture design for 2d human pose estimation. *CVPR*, 2022c. (Cited on page 17.)
- Wang, Yongxin; Kitani, Kris, and Weng, Xinshuo. Joint Object Detection and Multi-Object Tracking with Graph Neural Networks. *ICRA*, 2021f. (Cited on pages 31 and 32.)
- Wang, Zhikang; Zhu, Feng; Tang, Shixiang; Zhao, Rui; He, Lihuo, and Song, Jiangning. Feature Erasing and Diffusion Network for Occluded Person Re-Identification. In *CVPR*, 2022d. (Cited on pages 34 and 119.)
- Wang, Zhongdao; Zheng, Liang; Liu, Yixuan, and Wang, Shengjin. Towards Real-Time Multi-Object Tracking. *ECCV*, 2020d. (Cited on pages 29, 30, 88, 99, 102 and 117.)
- Wei, Fangyun; Sun, Xiao; Li, Hongyang; Wang, Jingdong, and Lin, Stephen. Point-Set Anchors for Object Detection, Instance Segmentation and Pose Estimation. *ECCV*, 2020. (Cited on page 15.)
- Wei, Longhui; Zhang, Shiliang; Yao, Hantao; Gao, Wen, and Tian, Qi. GLAD: Global-Local-Alignment Descriptor for Pedestrian Retrieval. *ICM*, 2017. (Cited on pages 33, 115 and 119.)
- Wei, Shih-En; Ramakrishna, Varun; Kanade, Takeo, and Sheikh, Yaser. Convolutional pose machines. In *CVPR*, 2016. (Cited on pages 10 and 14.)
- Weinzaepfel, Philippe; Revaud, Jerome; Harchaoui, Zaid, and Schmid, Cordelia. DeepFlow: Large Displacement Optical Flow with Deep Matching. In *ICCV*, 2013. (Cited on pages 18, 20 and 23.)
- Wen, Longyin; Li, Wenbo; Yan, Junjie; Lei, Zhen; Yi, Dong, and Li, Stan Z. Multiple target tracking based on undirected hierarchical relation hypergraph. In *CVPR*, 2014. (Cited on page 32.)
- Wen, Longyin; Du, Dawei; Li, Shengkun; Bian, Xiao, and Lyu, Siwei. Learning Non-Uniform Hypergraph for Multi-Object Tracking. *AAAI*, 2019. (Cited on page 28.)

- Wen, Yandong; Zhang, Kaipeng; Li, Zhifeng, and Qiao, Yu. A discriminative feature learning approach for deep face recognition. In *ECCV*, 2016. (Cited on page 59.)
- Wieczorek, Mikolaj; Rychalska, Barbara, and Dabrowski, Jacek. On the unreasonable effectiveness of centroids in image retrieval. In *ICNIP*, 2021. (Cited on page 33.)
- Williams, Christopher and Rasmussen, Carl. Gaussian processes for regression. In Touretzky, D.; Mozer, M.C., and Hasselmo, M., editors, *NIPS*, 1995. (Cited on page 29.)
- Wojke, Nicolai; Bewley, Alex, and Paulus, Dietrich. Simple Online and Realtime Tracking with a Deep Association Metric. In *ICIP*, 2017. (Cited on pages 26, 27 and 28.)
- Wu, Bo and Nevatia, R. Tracking of multiple, partially occluded humans based on static body part detection. In *CVPR*, 2006. (Cited on page 63.)
- Wu, Jiahong; Zheng, He; Zhao, Bo; Li, Yixin; Yan, Baoming; Liang, Rui; Wang, Wenjia; Zhou, Shipai; Lin, Guosen; Fu, Yanwei; Wang, Yizhou, and Wang, Yonggang. AI Challenger: A Large-scale Dataset for Going Deeper in Image Understanding. *arXiv preprint*, 2017. (Cited on page 68.)
- Wu, Yu; Lin, Yutian; Dong, Xuanyi; Yan, Yan; Ouyang, Wanli, and Yang, Yi. Exploit the Unknown Gradually: One-Shot Video-Based Person Re-Identification by Stepwise Learning. In *CVPR*, 2018. (Cited on page 91.)
- Wu, Yuxin and He, Kaiming. Group normalization. *IJCV*, 2018. (Cited on page 46.)
- Xia, Fangting; Wang, Peng; Chen, Xianjie, and Yuille, Alan L. Joint Multi-Person Pose Estimation and Semantic Part Segmentation. *CVPR*, 2017. (Cited on page 10.)
- Xiang, Yu; Alahi, Alexandre, and Savarese, Silvio. Learning to track: Online multi-object tracking by decision making. In *ICCV*, 2015. (Cited on page 29.)
- Xiao, Bin; Wu, Haiping, and Wei, Yichen. Simple Baselines for Human Pose Estimation and Tracking. In *ECCV*, 2018. (Cited on pages 2, 10, 11, 13, 17, 18, 19, 20, 21, 22, 23, 24, 53, 55, 68, 70, 74, 75, 76, 77, 82, 83, 88, 96, 98, 108, 109, 117 and 128.)
- Xiao, Tong; Li, Shuang; Wang, Bochao; Lin, Liang, and Wang, Xiaogang. End-to-End Deep Learning for Person Search. *CVPR*, 2017a. (Cited on pages 89, 90, 91 and 100.)
- Xiao, Tong; Li, Shuang; Wang, Bochao; Lin, Liang, and Wang, Xiaogang. Joint Detection and Identification Feature Learning for Person Search. *CVPR*, 2017b. (Cited on pages 35, 88 and 100.)
- Xiao, Yabo; Wang, Xiao Juan; Yu, Dongdong; Wang, Guoli; Zhang, Qian, and Mingshu, HE. Adaptivepose: Human parts as adaptive points. In *AAAI*, 2022. (Cited on page 16.)
- Xiu, Yuliang; Li, Jiefeng; Wang, Haoyu; Fang, Yinghong, and Lu, Cewu. Pose Flow: Efficient Online Pose Tracking. In *BMVC*, 2018. (Cited on pages 2, 17, 18, 19, 23, 57 and 82.)
- Xu, Chuanzhi and Zhou, Yue. Hierarchical Online Multi-person Pose Tracking with Multiple Cues. In Cheng, Long; Leung, Andrew Chi Sing, and Ozawa, Seiichi, editors, *ICONIP*, 2018. (Cited on pages 24 and 58.)

- Xu, Jiarui; Cao, Yue; Zhang, Zheng, and Hu, Han. Spatial-Temporal Relation Networks for Multi-Object Tracking. *ICCV*, 2019. (Cited on pages 28 and 29.)
- Xu, Yihong; Ban, Yutong; Delorme, Guillaume; Gan, Chuang; Rus, Daniela, and Alameda-Pineda, Xavier. TransCenter: Transformers with Dense Representations for Multiple-Object Tracking. In *TPAMI*, 2021. (Cited on pages 31 and 32.)
- Xu, Yufei; Zhang, Jing; Zhang, Qiming, and Tao, Dacheng. ViTPose: Simple Vision Transformer Baselines for Human Pose Estimation. In *NeurIPS*, 2022. (Cited on pages 7 and 13.)
- Xue, Hui; Chen, Songcan, and Yang, Qiang. Structural Support Vector Machine. In *ISNN*, 2008. (Cited on page 10.)
- Xue, Nan; Wu, Tianfu; Xia, Gui-Song, and Zhang, Liangpei. Learning Local-Global Contextual Adaptation for Multi-Person Pose Estimation. In *CVPR*, 2022. (Cited on page 16.)
- Yan, B.; Jiang, Yi; Sun, Pei; Wang, D.; Yuan, Zehuan; Luo, Ping, and Lu, Huchuan. Towards Grand Unification of Object Tracking. In *ECCV*, 2022. (Cited on page 31.)
- Yan, Yichao; Zhang, Qiang; Ni, Bingbing; Zhang, Wendong; Xu, Minghao, and Yang, Xiaokang. Learning context graph for person search. *CVPR*, 2019. (Cited on pages 35 and 36.)
- Yan, Yichao; Li, Jingpeng; Qin, Jie; Bai, Song; Liao, Shengcai; Liu, Li; Zhu, Fan, and Shao, Ling. Anchor-Free Person Search. *CVPR*, 2021. (Cited on pages 35 and 88.)
- Yang, Fan; Odashima, Shigeyuki; Masui, Shoichi, and Jiang, Shan. Hard to Track Objects with Irregular Motions and Similar Appearances? Make It Easier by Buffering the Matching Space. *WACV*, 2023. (Cited on page 27.)
- Yang, Sen; Quan, Zhibin; Nie, Mu, and Yang, Wankou. TransPose: Keypoint Localization via Transformer. In *ICCV*, 2021a. (Cited on page 13.)
- Yang, Yi and Ramanan, D. Articulated Pose Estimation with Flexible Mixtures-of-parts. In *CVPR*, 2011. (Cited on pages 8 and 9.)
- Yang, Yi and Ramanan, Deva. Articulated human detection with flexible mixtures of parts. *TPAMI*, 2013. (Cited on pages 10 and 60.)
- Yang, Yiding; Ren, Zhou; Li, Haoxiang; Zhou, Chunlun; Wang, Xinchao, and Hua, Gang. Learning Dynamics via Graph Neural Networks for Human Pose Estimation and Tracking. In *CVPR*, 2021b. (Cited on pages 13, 17, 21, 22, 23, 24, 88, 98, 108, 109, 117 and 118.)
- Yao, Bangpeng and Fei-Fei, Li. Recognizing Human-Object Interactions in Still Images by Modeling the Mutual Context of Objects and Human Poses. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2012. (Cited on page 19.)
- Yao, Hantao; Zhang, Shiliang; Hong, Richang; Zhang, Yongdong; Xu, Changsheng, and Tian, Qi. Deep representation learning with part loss for person re-identification. *TIP*, 2019. (Cited on page 33.)

- Ye, Yuteng; Zhou, Hang; Yu, Junqing; Hu, Qiang, and Yang, Wei. Dynamic Feature Pruning and Consolidation for Occluded Person Re-Identification. *ArXiv-Preprint*, 2022. (Cited on pages 34 and 119.)
- Yu, Dongdong; Su, Kai; Sun, Jia, and Wang, Changhu. Multi-Person Pose Estimation for Pose Tracking with Enhanced Cascaded Pyramid Network. In *ECCV Workshop*, 2018. (Cited on pages 83 and 118.)
- Yu, Fengwei; Li, Wenbo; Li, Quanquan; Liu, Yu; Shi, Xiaohua, and Yan, Junjie. POI: Multiple Object Tracking with High Performance Detection and Appearance Feature. In *ECCV Workshops*, 2016. (Cited on pages 26 and 28.)
- Yu, Fisher; Koltun, Vladlen, and Funkhouser, Thomas A. Dilated residual networks. *CVPR*, 2017. (Cited on page 56.)
- Yu, Rui; Du, Dawei; LaLonde, Rodney; Davila, Daniel; Funk, Christopher; Hoogs, Anthony, and Clipp, Brian. Cascade transformers for end-to-end person search. In *CVPR*, 2022. (Cited on pages 35 and 36.)
- Yu, Shijie; Chen, Dapeng; Zhao, Rui; Chen, Haobin, and Qiao, Yu. Neighbourhood-guided Feature Reconstruction for Occluded Person Re-Identification. *ArXiv-Preprint*, 2021. (Cited on pages 34 and 119.)
- Yuan, Yuhui; Fu, Rao; Huang, Lang; Lin, Weihong; Zhang, Chao; Chen, Xilin, and Wang, Jingdong. HRFormer: High-Resolution Transformer for Dense Prediction. In *NeurIPS*, 2021. (Cited on page 13.)
- Yue, Luo; Jimmy, Ren; Zhouxia, Wang; Wenxiu, Sun; Jinshan, Pan; Jianbo, Liu; Jiahao, Pang, and Liang, Lin. LSTM Pose Machines. In *CVPR*, 2018. (Cited on pages 18 and 19.)
- Zamir, Amir Roshan; Dehghan, Afshin, and Shah, Mubarak. GMCP-tracker: Global multi-object tracking using generalized minimum clique graphs. In *ECCV*, 2012. (Cited on pages 32 and 33.)
- Zand, Mohsen; Etemad, Ali, and Greenspan, Michael. Objectbox: From centers to boxes for anchor-free object detection. *ECCV*, 2022. (Cited on pages 25 and 26.)
- Zeng, Fangao; Dong, Bin; Zhang, Yuang; Wang, Tiancai; Zhang, Xiangyu, and Wei, Yichen. MOTR: End-to-End Multiple-Object Tracking with TRansformer. In *ECCV*, 2022. (Cited on pages 31 and 32.)
- Zhang, Aston; Lipton, Zachary C.; Li, Mu, and Smola, Alexander J. *Dive into Deep Learning*. Cambridge University Press, 2023a. (Cited on page 47.)
- Zhang, Feng; Zhu, Xiatian; Dai, Hanbin; Ye, Mao, and Zhu, Ce. Distribution-Aware Coordinate Representation for Human Pose Estimation. In *CVPR*, 2020a. (Cited on page 12.)
- Zhang, Jimuyang; Zhou, Sanping; Chang, Xin; Wan, Fangbin; Wang, Jinjun; Wu, Yang, and Huang, Dong. Multiple object tracking by flowing and fusing. *ArXiv*, 2020b. (Cited on page 30.)

- Zhang, Li; Li, Yuan, and Nevatia, Ramakant. Global data association for multi-object tracking using network flows. In *CVPR*, 2008. (Cited on pages 32 and 33.)
- Zhang, Rui; Zhu, Zheng; Li, Peng; Wu, Rui; Guo, Chaoxu; Huang, Guan, and Xia, Hailun. Exploiting Offset-guided Network for Pose Estimation and Tracking. In *CVPR*, 2019. (Cited on pages 2, 11, 22, 24, 68, 71, 76, 83, 88, 98 and 109.)
- Zhang, Xiangyu; Zhou, Xinyu; Lin, Mengxiao, and Sun, Jian. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. *CVPR*, 2017. (Cited on page 11.)
- Zhang, Yang; Sheng, Hao; Wu, Yubin; Wang, Shuai; Lyu, Weifeng; Ke, W., and Xiong, Zhang. Long-Term Tracking With Deep Tracklet Association. *IEEE Transactions on Image Processing*, 2020c. (Cited on pages 28 and 29.)
- Zhang, Yifu; Wang, Chunyu; Wang, Xinggang; Zeng, Wenjun, and Liu, Wenyu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *IJCV*, 2021. (Cited on pages 29, 30, 88, 99, 102 and 117.)
- Zhang, Yifu; Sun, Peize; Jiang, Yi; Yu, Dongdong; Weng, Fucheng; Yuan, Zehuan; Luo, Ping; Liu, Wenyu, and Wang, Xinggang. ByteTrack: Multi-Object Tracking by Associating Every Detection Box. *ECCV*, 2022. (Cited on pages 26, 27, 28 and 130.)
- Zhang, Yuang; Wang, Tiancai, and Zhang, Xiangyu. MOTRv2: Bootstrapping End-to-End Multi-Object Tracking by Pretrained Object Detectors. *CVPR*, 2023b. (Cited on pages 31 and 32.)
- Zhang, Yuexi; Wang, Yin; Camps, Octavia, and Sznaiier, Mario. Key Frame Proposal Network for Efficient Pose Estimation in Videos. In *ECCV*, 2020d. (Cited on page 19.)
- Zhao, Liming; Li, Xi; Zhuang, Yueting, and Wang, Jingdong. Deeply-learned part-aligned representations for person re-identification. In *CVPR*, 2017. (Cited on page 33.)
- Zhao, Zelin; Wu, Ze; Zhuang, Yueqing; Li, Boxun, and Jia, Jiaya. Tracking Objects as Pixel-wise Distributions. *ECCV*, 2022. (Cited on pages 31 and 32.)
- Zheng, Ce; Zhu, Sijie; Mendieta, Matias; Yang, Taojiannan; Chen, Chen, and Ding, Zhengming. 3d human pose estimation with spatial and temporal transformers. *ICCV*, 2021. (Cited on page 130.)
- Zheng, Dingyuan; Xiao, Jimin; Huang, Kaizhu, and Zhao, Yao. Segmentation mask guided end-to-end person search. *Signal Process. Image Commun.*, 2020. (Cited on page 35.)
- Zheng, L.; Shen, L.; Tian, L.; Wang, S.; Wang, J., and Tian, Q. Scalable Person Re-identification: A Benchmark. In *ICCV*, 2015. (Cited on pages xii, 24, 118 and 119.)
- Zheng, Liang; Bie, Zhi; Sun, Yifan; Wang, Jingdong; Su, Chi; Wang, Shengjin, and Tian, Qi. MARS: A Video Benchmark for Large-Scale Person Re-identification. In *ECCV*, 2016. (Cited on page 91.)
- Zheng, Liang; Zhang, Hengheng; Sun, Shaoyan; Chandraker, Manmohan; Yang, Yi, and Tian, Qi. Person Re-identification in the Wild. In *CVPR*, 2017a. (Cited on pages 33, 53, 89, 90, 91 and 100.)

- Zheng, Liang; Huang, Yujia; Lu, Huchuan, and Yang, Yi. Pose-Invariant Embedding for Deep Person Re-Identification. *IEEE Transactions on Image Processing*, 2019. (Cited on page 33.)
- Zheng, Zhedong; Zheng, Liang, and Yang, Yi. A Discriminatively Learned CNN Embedding for Person Re-identification. *ACM Transactions on Multimedia Computing Communications and Applications*, 2017b. (Cited on page 28.)
- Zhong, Yingji; Wang, Xiaoyu, and Zhang, Shiliang. Robust Partial Matching for Person Search in the Wild. *CVPR*, 2020. (Cited on page 35.)
- Zhou, and Chellappa, . Computation of optical flow using a neural network. In *IEEE 1988 International Conference on Neural Networks*, 1988. (Cited on page 48.)
- Zhou, Chunluan; Ren, Zhou, and Hua, Gang. Temporal keypoint matching and refinement network for pose estimation and tracking. In *ECCV*, 2020a. (Cited on pages 17, 21, 22, 23, 24, 109, 118 and 129.)
- Zhou, K.; Yang, Y.; Cavallaro, A., and Xiang, T. Omni-Scale Feature Learning for Person Re-Identification. In *ICCV*, 2019. (Cited on page 88.)
- Zhou, Kaiyang; Yang, Yongxin; Cavallaro, Andrea, and Xiang, Tao. Learning Generalisable Omni-Scale Representations for Person Re-Identification. *TPAMI*, 2021. (Cited on page 88.)
- Zhou, Xingyi; Koltun, Vladlen, and Krähenbühl, Philipp. Tracking Objects as Points. *ECCV*, 2020b. (Cited on pages 14, 16, 29, 30, 32, 55 and 88.)
- Zhou, Xingyi; Yin, Tianwei; Koltun, Vladlen, and Krähenbühl, Philipp. Global tracking transformers. In *CVPR*, 2022. (Cited on pages 32 and 33.)
- Zhou, Zongwei; Xing, Junliang; Zhang, Mengdan, and Hu, Weiming. Online Multi-Target Tracking with Tensor-Based High-Order Graph Matching. *ICPR*, 2018. (Cited on page 28.)
- Zhu, Xizhou; Hu, Han; Lin, Stephen, and Dai, Jifeng. Deformable ConvNets v2: More Deformable, Better Results. *ArXiv-Preprint*, 2018. (Cited on page 118.)
- Zhu, Xizhou; Hu, Han; Lin, Stephen, and Dai, Jifeng. Deformable ConvNets V2: More Deformable, Better Results. *CVPR*, 2019. (Cited on pages 18 and 118.)
- Zhu, Xizhou; Su, Weijie; Lu, Lewei; Li, Bin; Wang, Xiaogang, and Dai, Jifeng. Deformable DETR: Deformable Transformers for End-to-End Object Detection. In *ICLR*, 2021. (Cited on pages 26 and 31.)
- Zigang, Geng; Ke, Sun; Bin, Xiao; Zhaoxiang, Zhang, and Jingdong, Wang. Bottom-Up Human Pose Estimation Via Disentangled Keypoint Regression. In *CVPR*, 2021. (Cited on pages 9, 14, 16 and 55.)
- Zou, Shihao; Xu, Yuanlu; Li, Chao; Ma, Lingni; Cheng, Li, and Vo, Minh. Snipper: A spatiotemporal transformer for simultaneous multi-person 3d pose estimation tracking and forecasting on a video snippet. *TCSVT*, 2022. (Cited on page 130.)