Real-time 3D Human Body Pose Estimation from Monocular RGB Input

---

A dissertation submitted towards the degree
Doctor of Engineering
of the
Faculty of Mathematics and Computer Science
of
Saarland University

by
M.Sc. Dushyant Mehta

Saarbrücken, 2020

UNIVERSITÄT
DES
SAARLANDES

Dean of the Faculty:

Univ.-Prof. Dr. Thomas Schuster

Defense:

September 14, 2020, in Saarbrücken

Chair of the Committee:

Prof. Dr. Philipp Slusallek

Examiners:

Prof. Dr. Christian Theobalt
Dr. Gerard Pons-Moll
Prof. Dr. Adrian Hilton

Academic Assistant:

Dr. Vladislav Golyanik

Acknowledgments

This work was made possible through the direct and indirect contribution of many. I am grateful to my supervisor Christian Theobalt for his advice, guidance, patience, and encouragement throughout these past four years. I also want to give special thanks to my closest collaborators at MPI for Informatics, Oleksandr Sotnychenko and Franziska Mueller, without whom the extensive data capture and the final system implementation would not be possible.

I shall forever be indebted to my collaborators and colleagues Helge Rhodin, Dan Casas, Franziska Mueller, Oleksandr Sotnychenko, Weipeng Xu, Kwang-In Kim, Srinath Sridhar, Ayush Tewari, Florian Bernard, Gerard Pons-Moll, Michael Zollhöfer, Abhimitra Meka, Oliver Nalbach, Thomas Leimkuhler, Elena Arabadzhiyska-Koleva, Michal Piovarci, Bertram Somieski, Pablo Garrido, Nadia Robertini, Christian Richardt, Hyeongwoo Kim, Ikhsanul Habibie, Marc Habermann, Gereon Fox, Jiayi Wang, Jozef Hladky, Lingjie Liu, Kripasindhu Sarkar, and Vladislav Golyanik for their continued help, guidance, and friendship.

I have been incredibly fortunate to have been a part of the Computer Graphics Department for six years, and the GVV group for the past four, and thank all the current and past members of the department for enriching my stay here with their expertise and cordiality. I thank the administrative staff members of the Computer Graphics Department, Sabine Budde and Ellen Fries, and the hardware admins of the Department, Hyeongwoo Kim, Jozef Hladky, and Gereon Fox, for always being very responsive and patient with my requests.

I thank Abhimitra Meka, Vladislav Golyanik, Srinath Sridhar, Helge Rhodin, Dan Casas, Mohamed Elgharib, and Weipeng Xu for their feedback on early drafts of the thesis. Although I may whinge and moan incessantly about reviewer comments on the manuscripts of the papers that this thesis is comprised of, the subsequent revisions of the manuscripts did get better as a result of the extensive reviewer feedback, and I thank all the anonymous reviewers who volunteered their time and expertise. I am also grateful to the many many people who volunteered their time and likeness to enable creation of the various datasets proposed in the thesis, and who featured in the demo videos to showcase the capabilities of the developed systems.

I am grateful to the various funding agencies that supported this thesis: ERC Starting Grant CapReal (335545) and ERC Consolidator Grant 4DRepLy (770784). I would also like to thank the thesis committee and the panel for the Promotionskolloquium for their time and expertise.

I would also like to thank my family, and my friends from outside the Computer Graphics Department: Tinqi Fan, Eldar Insafutdinov, Philipp Flotho, Michael Hedderich, Nikita Vedeneev, Junaid Ali, Lakshaya Agnani, and Divam Narula, for their support and camaraderie.

Abstract

Human motion capture finds extensive application in movies, games, sports and biomechanical analysis. However, existing motion capture solutions require cumbersome external and/or on-body instrumentation, or use active sensors with limits on the possible capture volume dictated by power consumption. The ubiquity and ease of deployment of RGB cameras makes monocular RGB based human motion capture an extremely useful problem to solve, which would lower the barrier-to-entry for content creators to employ motion capture tools, and enable newer applications of human motion capture. This thesis demonstrates the first real-time monocular RGB based motion-capture solutions that work in general scene settings. They are based on developing neural network based approaches to address the ill-posed problem of estimating 3D human pose from a single RGB image, in combination with model based fitting.

In particular, the contributions of this work make advances towards three key aspects of real-time monocular RGB based motion capture, namely speed, accuracy, and the ability to work for general scenes. New training datasets are proposed, for single-person and multi-person scenarios, which, together with the proposed transfer learning based training pipeline, allow learning based approaches to be appearance invariant. The training datasets are accompanied by evaluation benchmarks with multiple avenues of fine-grained evaluation. The evaluation benchmarks differ visually from the training datasets, so as to promote efforts towards solutions that generalize to in-the-wild scenes. The proposed task formulations for the single-person and multi-person case allow higher accuracy, and incorporate additional qualities such as occlusion robustness, that are helpful in the context of a full motion capture solution. The multi-person formulations are designed to have a nearly constant inference time regardless of the number of subjects in the scene, and combined with contributions towards fast neural network inference, enable real-time 3D pose estimation for multiple subjects. Combining the proposed learning-based approaches with a model-based kinematic skeleton fitting step provides temporally stable joint angle estimates, which can be readily employed for driving virtual characters.

Zusammenfassung

Menschlicher Motion Capture findet umfangreiche Anwendung in Filmen, Spielen, Sport und biomechanischen Analysen. Bestehende Motion-Capture-Lösungen erfordern jedoch umständliche externe Instrumentierung und / oder Instrumentierung am Körper, oder verwenden aktive Sensoren deren begrenztes Erfassungsvolumen durch den Stromverbrauch begrenzt wird. Die Allgegenwart und einfache Bereitstellung von RGB-Kameras macht die monokulare RGB-basierte Motion Capture zu einem äußerst nützlichen Problem. Dies würde die Eintrittsbarriere für Inhaltsersteller für die Verwendung der Motion Capture verringern und neuere Anwendungen dieser Tools zur Analyse menschlicher Bewegungen ermöglichen.

Diese Arbeit zeigt die ersten monokularen RGB-basierten Motion-Capture-Lösungen in Echtzeit, die in allgemeinen Szeneneinstellungen funktionieren. Sie basieren auf der Entwicklung neuronaler netzwerkbasierter Ansätze, um das schlecht gestellte Problem der Schätzung der menschlichen 3D-Pose aus einem einzelnen RGB-Bild in Kombination mit einer modellbasierten Anpassung anzugehen. Insbesondere machen die Beiträge dieser Arbeit Fortschritte in Richtung drei Schlüsselaspekte der monokularen RGB-basierten Echtzeit-Bewegungserfassung, nämlich Geschwindigkeit, Genauigkeit und die Fähigkeit, für allgemeine Szenen zu arbeiten. Es werden neue Trainingsdatensätze für Einzel- und Mehrpersonen-Szenarien vorgeschlagen, die zusammen mit der vorgeschlagenen Trainingspipeline, die auf Transferlernen basiert, ermöglichen, dass lernbasierte Ansätze nicht von Unterschieden im Erscheinungsbild des Bildes beeinflusst werden. Die Trainingsdatensätze werden von Bewertungsbenchmarks mit mehreren Möglichkeiten einer feinkörnigen Bewertung begleitet. Die angegebenen Benchmarks unterscheiden sich visuell von den Trainingsaufzeichnungen, um die Entwicklung von Lösungen zu fördern, die sich auf verschiedene Szenen verallgemeinern lassen. Die vorgeschlagenen Aufgabenformulierungen für den Einzel- und Mehrpersonenfall ermöglichen eine höhere Genauigkeit und enthalten zusätzliche Eigenschaften wie die Robustheit der Okklusion, die im Kontext einer vollständigen Bewegungserfassungslösung hilfreich sind. Die Mehrpersonenformulierungen sind so konzipiert, dass sie unabhängig von der Anzahl der Subjekte in der Szene eine nahezu konstante Inferenzzeit haben. In Kombination mit Beiträgen zur schnellen Inferenz neuronaler Netze ermöglichen sie eine 3D-Posenschätzung in Echtzeit für mehrere Subjekte. Die Kombination der vorgeschlagenen lernbasierten Ansätze mit einem modellbasierten kinematischen Skelettanpassungsschritt liefert zeitlich stabile Gelenkwinkelschätzungen, die leicht zum Ansteuern virtueller Charaktere verwendet werden können.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis proposes neural network based methods and processes for enabling real-time 3D human body pose estimation in general scenes captured with a single RGB camera. The proposed methods and processes are meant to serve as the backbone for monocular marker-less motion capture solutions, with applications in diverse areas, ranging from entertainment to sports analysis, where much more cumbersome and expensive setups are used typically.

In particular, this work makes key contributions to three aspects of monocular RGB based 3D pose estimation that are necessary for a deployable solution, namely speed, accuracy, and the ability to work for in-the-wild scenes. These contributions advance the state of the art in the highly under-constrained monocular RGB based body pose estimation setups. Combining the proposed learning-based approach with a model-based kinematic skeleton fitting step provides the fourth key aspect – temporally stable joint angle estimates, which can be readily employed for driving virtual characters.

The thesis shall serve as a step-by-step guide to building an accurate and real-time monocular RGB based motion capture solution, with contributions ranging from data capture and processing, to training formulations and methodologies, to insights about fast neural network architecture design, and bringing these aspects together into a complete system.

In This Chapter

- Introduction to the topic in detail, and discussion of the challenges and application opportunities of easily deployable low-cost motion capture solutions
- Elaboration upon the particular problem of monocular RGB input based body pose estimation, and the associated scientific challenges
- Laying out the structure of the thesis, and touching upon the contributions made towards the key aspects of a deployable monocular RGB based motion capture system identified earlier

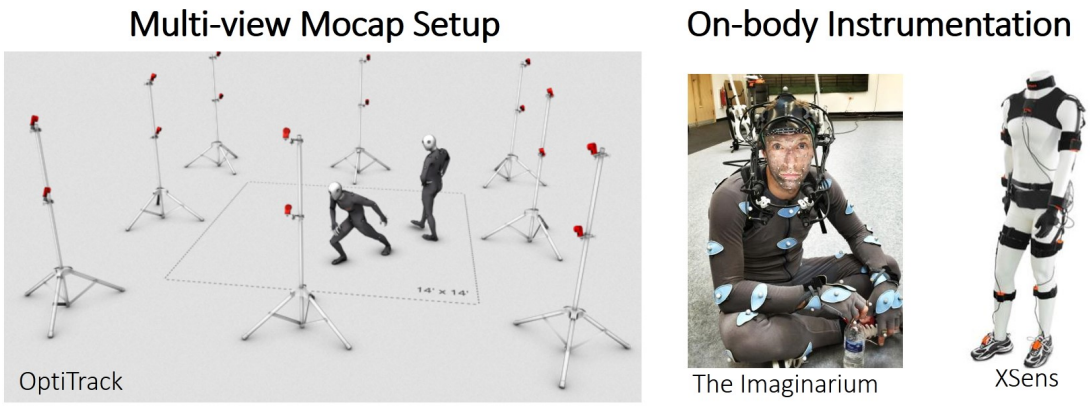

Figure 1.1: Typical Motion Capture Equipment: Multi-view systems are not portable, and may require extensive on-body instrumentation in addition. Inertial sensing based systems (shown on the right) are portable, but still require extensive instrumentation.

1.1 Low Cost and Easy To Deploy Motion Capture: Challenges and Opportunities

Understanding humans in general scenes has been one of the long standing goals of computer graphics and computer vision, with applications as far ranging as social sciences, communication, security, and entertainment. For entertainment, readily deployable motion capture finds direct application in interactive games, content creation for virtual worlds, educational and art installations. The low associated deployment cost of the systems proposed in the course of this research makes motion capture available as a tool to independent content creators, such as those on online video platforms. Beyond entertainment, knowing how humans behave and interact with elements of the scene also carries information regarding the semantic content of the scene, which can help computers better parse human centric scenes for applications such as cooperative / assistive robotics.

For such applications, the articulation and location of a simplified kinematic model of the human body is estimated, typically using extensive external- and/or on-body-instrumentation as shown in Figure 1.1. The extensive and often intrusive instrumentation limits the applicability of such solutions to restricted and carefully setup scenarios, and is not only cumbersome, but expensive as well. Multi-view video based systems which track body parts based on color and reprojection consistency across views, and monocular pose estimation system utilizing depth sensing, both do away with the need for on-body instrumentation. However, these systems still present several restrictions. Multi-view video camera setups are not portable, while depth sensing based systems have a limited capture volume, higher power requirement, and are prone to interference from sunlight.

1.1.1 Monocular RGB Base 3D Human Body Pose Capture

Given the ubiquity of RGB cameras, and consequently the vast amounts of pre-recorded data available through the internet, a monocular RGB based motion capture solution would be ideal, both in terms of ease of deployment owing to simple sensor equipment, and as a step towards understanding humans in general scenes by leveraging pre-recorded data. A monocular RGB based solution democratizes motion capture technology, making it available to everyone for use in entertainment or content creation.

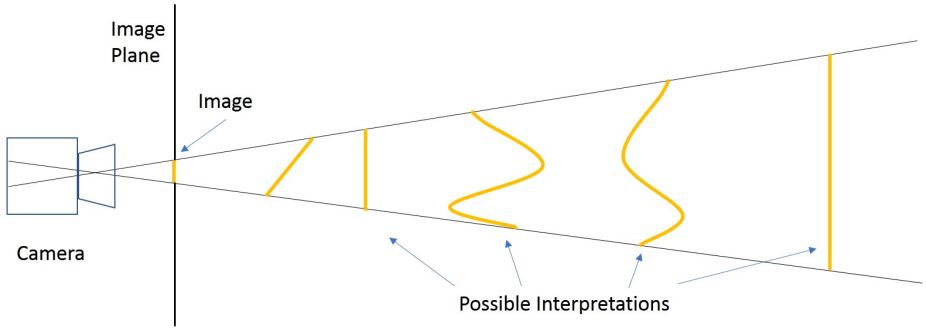

Figure 1.2: Perspective Projection Ambiguity: Multiple scene structures of different shapes, sizes, and orientations can produce the same projective image. Deciphering properties of the structure, such as its orientation and distance from the camera, using only the projective image consequently requires some prior knowledge of the structures being observed.

However, 3D pose estimation from just a single RGB camera is ill-posed. Due to the perspective projection ambiguity it is not possible to know the true size of the kinematic structure being observed, and its distance from the camera, as shown in Figure 1.2. Through explicit incorporation of priors on body proportions (bone lengths), or by learning such priors from data, the problem can be made more tractable.

Many applications of motion capture, such as driving virtual characters, primarily require estimates of the articulation of the kinematic 3D structure, and the scale of the relative localization of the kinematic structure with respect to the camera can be manually calibrated. Thus we make the simplifying assumption that all subjects observed have the same height, and the relative body proportions are learned from data. When the subject's true height is known, it allows calibration of the scale of the camera relative location estimates.

Even though these simplifications provide a starting point for learning based approaches, there are several additional challenges that need to be overcome to create a monocular RGB based motion capture system.

Formally, given a monocular RGB image I of size $W \times H$ as input, and optionally the heights of the subjects in the scene, the solution should output the 3D articulation $\theta_i \in \mathbb{R}^D$ of the kinematic structure of each subject i in the scene, where D is the degrees of freedom of the kinematic model, along with the location $\mathbf{d}_i \in \mathbb{R}^3$ of a reference joint (pelvis) relative to the camera. See Figure 1.3. If subject heights are not known, they can either be estimated through a calibration step, or each subject is assumed to have the same height. In that case, the localization relative to the camera is up to a scale. The 3D articulation θ also includes the orientation of the kinematic structure relative to the camera. The camera relative pose can also be expressed in terms of joint positions $\mathbf{P}^{\mathrm{G}_i} = FK(\theta_i, \mathbf{d}_i)$, obtained by applying forward kinematics to the kinematic model.

The approaches developed in this thesis leverage learning based methods, for which the 3D articulation is expressed in terms of joint positions obtained by applying forward kinematics to a kinematic structure with a known height. The 3D pose $\mathcal{P}^{3\mathcal{D}} = \{\mathbf{P}_\mathbf{i}^{\mathbf{3D}}\}_{i=1}^{m}$ for each of the m persons in the image, where, $\mathbf{P}_\mathbf{i}^{\mathbf{3D}} \in \mathbb{R}^{3\times J}$ describes the 3D locations of the J body joints of person i relative to a reference joint on the body, usually the pelvis. The joint locations can equivalently be expressed relative to the kinematic parent joints and trivially converted to pelvis-relative locations. Per detected subject, the articulation $\mathbf{P}_\mathbf{i}^{\mathbf{3D}} \in \mathbb{R}^{3\times J}$ predicted from learning based methods, together with a localization of the body joints $\mathbf{P}_\mathbf{i}^{\mathbf{2D}} \in \mathbb{R}^{2\times J}$ in the image plane, are subsequently passed to a kinematic model fitting step along with information about the skeleton scale. Kinematic model fitting incorporates

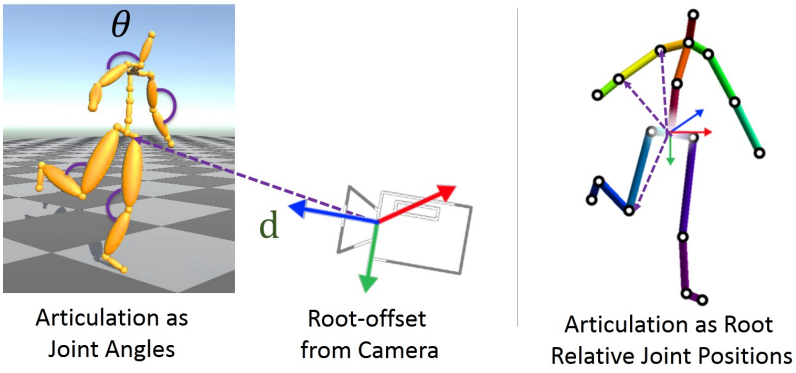

Figure 1.3: 3D Pose Representation: The output sought from the solution is the body skeleton articulation of the subjects in the image, expressed in terms of joint angles, and localization of the subject relative to the camera (left). The learning based component of the proposed solutions expresses body skeleton articulation in terms of root (pelvis) relative joint positions (right), which are converted to temporally smooth joint angles by the kinematic fitting step. See Section 1.1.1.

information across frames to produce temporally smooth joint angle estimates, and localizes the skeletal structures relative to the camera.

1.1.2 Limitations of Prior Work

Marker-less 3D motion capture methods that *track* articulated human poses from *multi-view* video sequences, often are restricted to controlled scenes (Balan et al. 2007; Chai et al. 2005; Gall et al. 2010; Sminchisescu et al. 2001; Starck et al. 2003; Stoll et al. 2011; Urtasun et al. 2005; Wren et al. 1997). Special RGB-D cameras enable real-time monocular pose estimation (Shotton et al. 2013) or motion tracking (Baak et al. 2011; Microsoft Corporation 2015), but often do not work in general scenes with clutter, as well as outdoors in bright sunlight.

The availability of large amounts of annotated in-the-wild 2D pose datasets in conjunction with data-driven approaches using Convolutional Neural Networks (CNNs) have shown impressive results for 2D body pose estimation (Belagiannis et al. 2016; Bulat et al. 2016; Carreira et al. 2016; X. Chen et al. 2014; Gkioxari et al. 2016; P. Hu et al. 2016; Insafutdinov et al. 2016; Lifshitz et al. 2016; Newell et al. 2016; Pishchulin et al. 2016; Tompson et al. 2014; Toshev et al. 2014; S.-E. Wei et al. 2016), outperforming previous hand crafted and model-based methods by a large margin (A. Agarwal et al. 2006; Mykhaylo Andriluka et al. 2009; Felzenszwalb et al. 2005).

Direct 3D pose regression from images, however, remains challenging on multiple accounts. A common approach is to lift 2D keypoints to 3D (Bogo et al. 2016; S. Li et al. 2015; Simo-Serra et al. 2013, 2012; C. Wang et al. 2014; Yasin et al. 2016; F. Zhou et al. 2014; Xiaowei Zhou et al. 2015a,c), but suffers from depth ambiguities. Though recent advances in direct CNN-based 3D regression show promise, utilizing different prediction space formulations (S. Li et al. 2014; Bugra Tekin et al. 2016a; Xingyi Zhou et al. 2016) and incorporating additional constraints (Bugra Tekin et al. 2016b; Y. Yu et al. 2016; Xiaowei Zhou et al. 2015c; Xingyi Zhou et al. 2016), the achieved results are far from the accuracy levels seen for 2D pose prediction.

The difficult nature of the problem aside, 3D pose prediction is further stymied by the lack of suitably large and diverse annotated 3D pose corpora. The internet provides virtually limitless images of

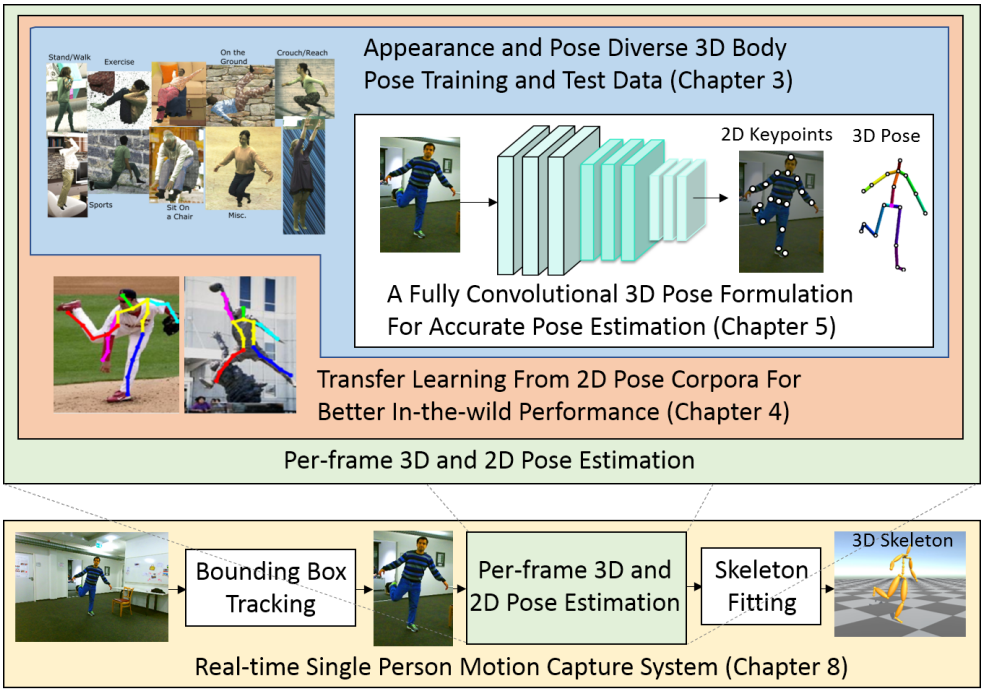

Figure 1.4: Overview of the general framework developed in this thesis for real-time 3D pose estimation from a single RGB camera in single person scenarios. Given a stream of input images, the 2D and 3D pose of the subject are predicted per frame using proposed neural network based approaches. These are passed on to a model based skeleton fitting step, which reconciles the 2D and 3D poses across time, while also removing outliers, to produce temporally coherent joint angle estimates, and also localize the subject relative to the camera. To speed up computation, the predicted 2D pose is used to bootstrap a bounding box tracker to crop out the appropriate region from the next frame, so as to not run the expensive neural network stage on regions not containing the subject.

humans, but unlike manual 2D pose annotation (Mykhaylo Andriluka et al. 2014; Johnson et al. 2010; Sapp et al. 2013), annotating with 3D poses is infeasible owing to the inherent ambiguities. Existing largest datasets either use marker-based motion capture for 3D annotation (Ionescu et al. 2014b; Sigal et al. 2010), which restricts recording to skin-tight clothing, or marker-less systems (Joo et al. 2015) with diverse clothing, but still restrictive in terms of scene appearance. In general, real recordings are hardly scalable to tens of thousands of examples required to explain the vast variability of human pose and appearance. Synthesizing example images is an alternative, however it may limit generalization to real scenes (Wenzheng Chen et al. 2016) due to over-simplified animation and rendering. This systemic lack of large visually diverse 3D data corpora notably constrains accuracy and generalization of prior methods, and strongly motivates the dataset and training schema contributions of this work.

In the recent years, neural network based approached have come to dominate the field of computer vision (K. He et al. 2016a; Krizhevsky et al. 2012; Simonyan et al. 2014). Though the narrative surrounding the development of these data driven approaches has been 'end-to-end' learning, hand engineering hasn't entirely been replaced. In place of hand engineered features, such as HoG (Dalal et al. 2005), it is problem formulations that are the focus of domain knowledge guided design. A significant part of the thesis would be devoted to designing appropriate formulations for human pose estimation, while trying to incorporate several desirable properties such as occlusion robustness.

Additionally, the shift towards neural network based approaches comes with a significant computa-

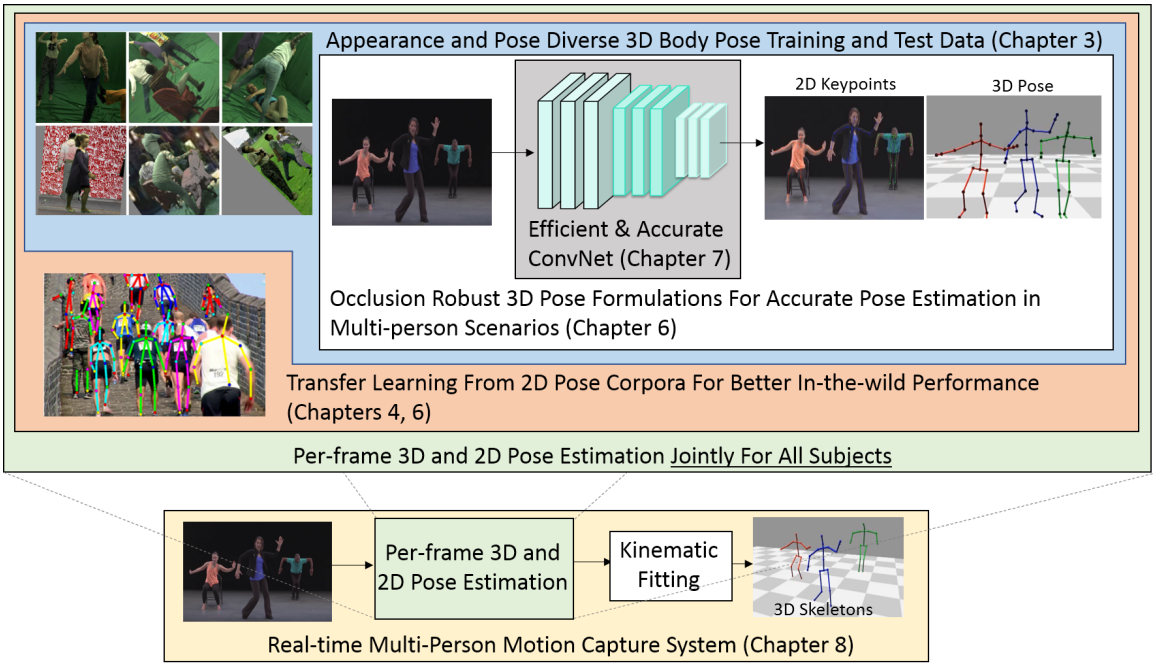

Figure 1.5: Overview of the general framework developed in this thesis for real-time multi-person 3D pose estimation from a single RGB camera. As with the single person setting, given a stream of input images, the 2D and 3D poses of all subjects are predicted per frame using the proposed neural network based approaches. Importantly, all subjects are handled jointly, which mitigates a linear dependency of the computational cost on the number of subjects. Subsequently, per subject, the 2D and 3D pose are passed on to a model based kinematic fitting step, which reconciles the 2D and 3D poses across time, to produce temporally coherent joint angle estimates per subject, while also localizing the subjects in the scene relative to the camera.

tional cost at inference time, proving a hindrance to real-time deployment of neural network based motion capture solutions on typical consumer hardware. This thesis also makes several contributions towards speeding up inference of convolutional neural networks.

1.2 Scope and Overview

This thesis proposes tools and techniques towards reliable capture of 3D human body pose from a single RGB camera in general scenes, as well as benchmarks and metrics for the evaluation of the performance of such systems. The focus is on enabling in-the-wild motion capture in a wide variety of scenes, which runs in real-time on typical consumer desktop computers, and yields temporally smooth estimates. The techniques investigated are primarily based on convolutional neural networks, with contributions towards dataset capture, training objective formulation, training schema, neural network architecture, and system design, to overcome the various challenges discussed previously. An overview of the pipeline for single person pose estimation is shown in Figure 1.4, and the pipeline for multi-person pose estimation is shown in Figure 1.5.

1.3 Structure and Technical Contributions

The thesis is organized into 9 chapters, with Chapters 3-8 covering the main technical contributions ranging from dataset capture to training schema to design of training objectives for accurate and occlusion-robust single- and multi-person pose estimation to designing real-time systems to handle single and multi-person scenarios.

- Chapter 2 discusses prior art in 2D and 3D pose estimation, as well as work related to efficient inference with convolutional neural networks. This chapter, both contextualizes the contributions of the thesis with regards to prior and concurrent work, as well as provides the relevant background knowledge.

- Chapter 3 describes in detail the motivation, methodology, and process behind acquisition of image data with 3D pose annotations, for the purposes of training and evaluation. The contributions are in designing the capture setting to increase the appearance, viewpoint, and pose diversity compared to existing motion capture datasets for the single-person case, and scaling up the single-person dataset through image composition to provide training data for the multi-person case. Further, challenging evaluation benchmarks for the single-person and multi-person case are proposed, comprised of real video sequences with ground-truth captured with multi-view marker-less motion capture in scene settings different from the training datasets, to test the generalizability of proposed methods to diverse scene settings. The chapter also proposes various avenues of fine-grained evaluation on the proposed benchmarks.

- Chapter 4 describes the first attempts at in-the-wild pose estimation in the slightly simpler setting of unoccluded single person pose estimation, formulated as a direct image to pose regression problem. The chapter discusses the need to go beyond the avenues of appearance augmentation introduced in the captured dataset, by transfer learning based training schema that leverage features learned on in-the-wild annotated 2D pose data to close the generalization gap to real in-the-wild images.

- Chapter 5 describes a novel 3D pose encoding using a fully-convolutional formulation that tightly couples pose inference for each body part to its direct image evidence. This chapter develops the key insight of focusing the receptive fields of the network on the appropriate body part while making an inference for that particular body part. This insight would be a recurring feature in the pose formulations developed for the multi-person case. The formulation is shown to be more accurate and more robust to various perturbations than direct image to pose regression. Further, the strong coupling of pose inference to image evidence leads to pose estimates that overlay well on the input image, a feature that is important for various AR/VR applications.

- Chapter 6 further builds on the insights developed in the previous chapter to develop pose formulations for the multi-person case, which are robust to occlusions. Since the objective is real-time pose estimation, it is desired that the output size and inference time should not scale arbitrarily with the number of subjects in the scene. Two pose formulations are proposed, the first of which incorporates occlusion robustness through a selective use of redundancy in the pose encoding. While such a pose formulation allows for pose estimates under occlusion, it suffers from several shortfalls under inter-personal occlusion. The second pose formulation improves upon this by limiting itself to inference of body parts for which direct evidence is available. A separate small network then infers the occluded body parts, in the context of the

pose estimates for the visible body parts. This chapter also introduces fine-grained evaluation metrics for the multi-person evaluation dataset proposed in Chapter 3.

- Chapter 7 discusses aspects of real-time inference with convolutional neural networks on consumer hardware. The key contributions are in terms of efficient neural network architecture design, in part inspired by some surprising insights on implicit pruning convolutional neural networks. The proposed convolutional network architecture is faster and more memory efficient than the typical state-of-the-art baseline architectures, and more amenable to feature pruning, leading to further speed-up.
- Chapter 8 focuses on the design of the complete system, combining per-frame estimates coming from the techniques developed thus far, with kinematic model based fitting that is cognizant of previous frames' estimates, to produce temporally smooth joint angle estimates which can be readily employed to drive rigged characters.
- Chapter 9 concludes the thesis with a discussion of the avenues of future work. This chapter proposes possible ways to address the shortcomings of the presented work, and also presents further applications and extensions of the developed techniques.

1.4 List of Publications

The contributions mentioned above are encompassed by the following publications:

- Monocular 3D Human Pose Estimation In The Wild Using Improved CNN Supervision, D. Mehta; H. Rhodin; D. Casas; P. Fua; O. Sotnychenko; W. Xu; C. Theobalt, International Conference on 3D Vision (3DV) 2017 (*Covered by Chapters 3 & 4, and Appendices A & B*)
- VNect: Real-time 3D Human Pose Estimation with a Single RGB Camera, D. Mehta; S. Sridhar; O. Sotnychenko; H. Rhodin; M. Shafiei; H.P. Seidel; W. Xu; D. Casas; C. Theobalt, ACM Transactions on Graphics (SIGGRAPH) 2017 (*Covered by Chapters 5 & 8*)
- Single-Shot Multi-Person 3D Body Pose Estimation From Monocular RGB Input, D. Mehta; O. Sotnychenko; F. Mueller; W. Xu; S. Sridhar; G. Pons-Moll; C. Theobalt, International Conference on 3D Vision (3DV) 2018 (*Covered by Chapters 3 & 6, and Appendix C*)
- On Implicit Filter Level Sparsity in Convolutional Neural Networks, D. Mehta; K.I. Kim; C. Theobalt Computer Vision and Pattern Recognition (CVPR) 2019 (*Covered by Chapter 7 and Appendix D*)
- XNect: Real-time Multi-person 3D Human Pose Estimation with a Single RGB Camera, D. Mehta; O. Sotnychenko; F. Mueller; W. Xu; M. Elgharib; P. Fua; H.P. Seidel; H. Rhodin; G. Pons-Moll; C. Theobalt, ACM Transactions on Graphics (SIGGRAPH) 2020 (*Covered by Chapters 6 & 8*)

Chapter 2

Background and Related Work

Towards the goal of real-time and accurate capture of 3D human body pose from monocular RGB images in general scene settings, the thesis contributes new training datasets and evaluation metrics, problem formulations for reliable singe- and multi-person pose estimation, training schema for leveraging disparate annotations across datasets, insights into fast convolutional neural network design, and the design of systems based on these contributions to produce temporally smooth joint angle estimates that can be readily employed in various applications.

Given the broad scope of contributions, this chapter only covers the background and related work immediately relevant to the particular contributions of the thesis. The thesis assumes prior knowledge of Deep Learning and a familiarity with the associated terminology. One may refer to the many excellent resources on the topic, such as the book on *Deep Learning* by Goodfellow et al. 2016, for the relevant prior knowledge.

In This Chapter

- Discussion of seminal and contemporary work on 2D body pose estimation and tracking from RGB images/videos (Section 2.1)
- Discussion of prior and contemporary work on 3D body pose estimation using various sensing modalities and multi-view RGB images (Section 2.2)
- Discussion of contemporary 3D body pose estimation approaches from monocular RGB input, and comparison to the contributions of the thesis (Section 2.2.3)
- Discussion of Neural Network weight sparsification methods, and the interplay of sparsity with generalization and accuracy (Section 2.3.1)
- Discussion of Convolutional Neural Network architectures, with a focus on designs for fast inference (Section 2.3.2)

2.1 Monocular RGB Based 2D Body Pose Estimation

Discriminative 2D human pose estimation is often an intermediate step to monocular 3D pose estimation. Monocular 2D pose estimation has been extensively studied in the past. Approaches

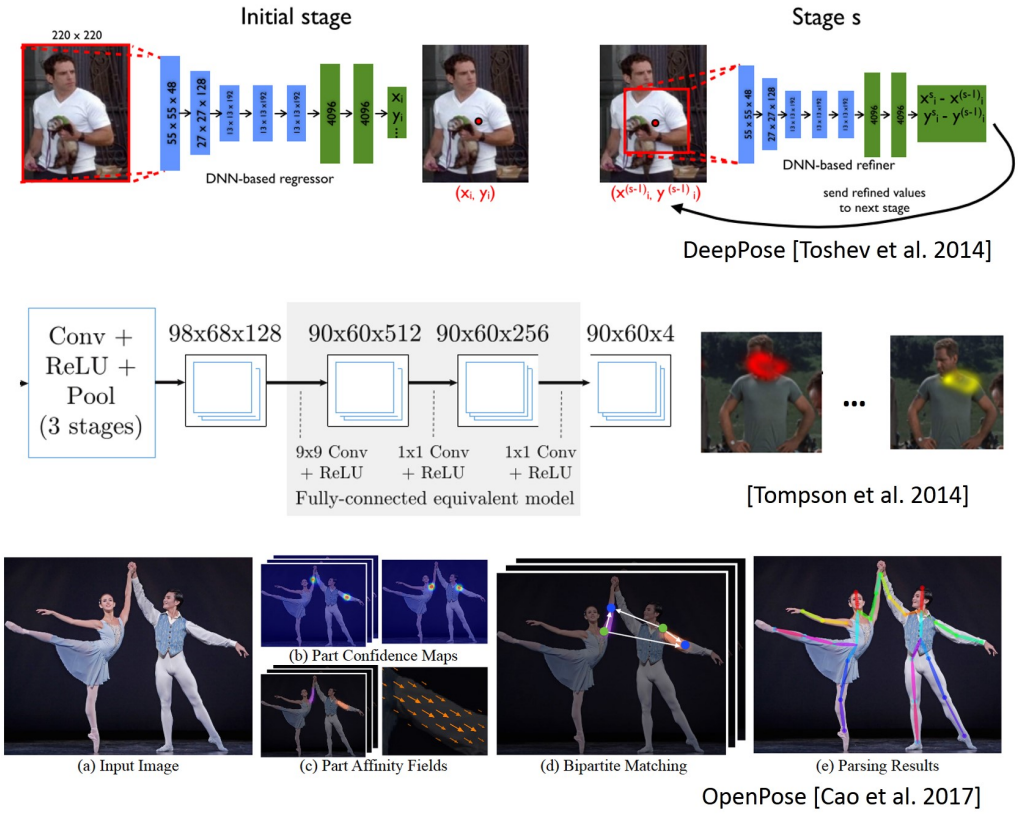

Figure 2.1: 2D body pose estimation approaches using Convolutional Neural Networks have matured from vectorized body keypoint coordinate prediction (Toshev et al. 2014), to keypoint heatmap prediction (Tompson et al. 2014), to additionally predicting other convolutional feature maps (Cao et al. 2017; Newell et al. 2017) to facilitate association of keypoints to person identities in multi-person scenarios.

using part detectors and graphical models (A. Agarwal et al. 2006; Mykhaylo Andriluka et al. 2009; Bourdev et al. 2009; Fastovets et al. 2013; Felzenszwalb et al. 2005; Ferrari et al. 2009) were successively outperformed by CNN-based approaches (Jain et al. 2013, 2014; Pishchulin et al. 2016; Toshev et al. 2014; S.-E. Wei et al. 2016). Here, the discussion focuses on aspects of CNN based 2D pose estimation.

The key development in CNN-based 2D pose estimation has been the shift from vectorized prediction of body keypoint coordinates (Toshev et al. 2014), to prediction of convolutional feature maps which encode the detection confidence of the keypoints in the image plane in the form of heatmaps (Tompson et al. 2014). The pixel locations of the maxima of the heatmaps indicate the coordinates of the respective keypoints, as shown in Figure 2.2, and the predictions are much more accurate than vectorized coordinate predictions on account of being strongly linked to the image evidence. The 3D pose formulations developed in this thesis are motivated by this development, and propose ways to similarly shift 3D pose inference away from a vectorized formulation to convolutional feature maps that strongly link the pose inference of each body part to its direct image evidence.

Multi-Person 2D Pose Estimation Estimating the coordinates suffices for single person scenarios, however for multi-person scenarios additionally require the detected parts to be grouped into identities.

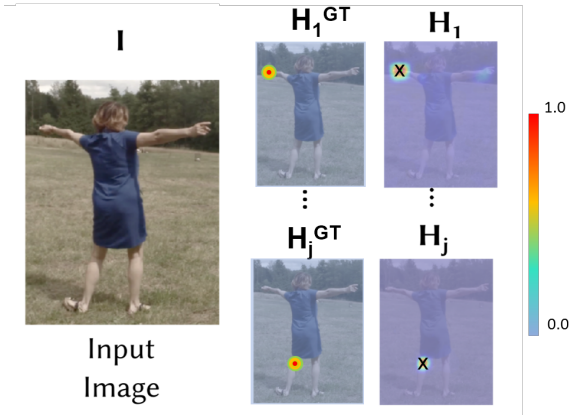

Figure 2.2: Heatmap based formulation for 2D body pose estimation. The network is trained to predict per-pixel confidence $\in [0, 1]$ that the pixel overlays body joint j. The heatmaps are typically $\frac{1}{8}th$ the spatial size of the input image. The network is trained either using cross-entropy loss which treats the heatmap prediction as a per-pixel classification problem, or with a euclidean loss between the predicted heatmaps H_j and the ground truth heatmaps H_j^{GT}, created by putting a gaussian peak with a limited spatial support at the annotated keypoint location. The location of the maximum of the predicted heatmap is used as the predicted keypoint location.

Figure 2.3: Representative frames from various 2D body pose datasets. All datasets provide body keypoint annotations, while some additionally provide instance and/or part segmentation (Liang et al. 2018; T.-Y. Lin et al. 2014). Some datasets provide temporal sequences of annotations (M. Andriluka et al. 2018), and others also provide sparse correspondences to a body mesh (Güler et al. 2018).

Multi-person 2D pose estimation methods can be divided into bottom-up and top-down approaches. Top-down approaches first detect individuals in a scene and fall back to single-person 2D pose approaches or variants for pose estimation (Gkioxari et al. 2014; Iqbal et al. 2016; Papandreou et al. 2017; Pishchulin et al. 2012; Sun et al. 2011). Reliable detection of individuals under significant occlusion, and tracking of people through occlusions remains challenging.

Bottom-up approaches instead first localize the body parts of all subjects and associate them to individuals in a second step. Associations can be obtained by predicting joint locations and their identity embeddings together (Newell et al. 2017), or by solving a graph cut problem (Insafutdinov et al. 2017; Pishchulin et al. 2016). This involves solving an NP-hard integer linear program which easily takes hours per image. The work of Insafutdinov et al. 2017 improves over Pishchulin et al. 2016 by including image-based pairwise terms and stronger detectors based on ResNet (K. He et al. 2016a). This way reconstruction time reduces to several minutes per frame. Cao et al. 2017 predict joint locations and part affinities (PAFs), which are 2D vectors linking each joint to its parent. PAFs allow quick and greedy part association, enabling real time mutli-person 2D pose estimation.

Güler et al. 2018 compute dense correspondences from pixels to the surface of SMPL (M. Loper et al. 2015), but they do not estimate 3D pose.

Recent work has also looked into tracking person identities (and hence body parts) across frames (Girdhar et al. 2018; Insafutdinov et al. 2017; Iqbal et al. 2017; Xiu et al. 2018).

2.1.1 2D Body Pose Datasets

The success of CNN-based approaches has been in a large part due to large image databases in diverse scene settings with 2D joint location annotations. LSP (Johnson et al. 2010) and LSP-Extended (Johnson et al. 2011) provide keypoint annotations in single-person scenes in sports poses. LIP (Liang et al. 2018) provides keypoint annotations, as well as part segmentations in single-person scenarios. MPI Human Pose (Mykhaylo Andriluka et al. 2014) and MSCOCO (T.-Y. Lin et al. 2014) represent general multi-person scenes. MSCOCO additionally provides instance segmentation for multi-person scenes. PoseTrack (M. Andriluka et al. 2018) provides 2D pose annotation on temporal sequences. DensePose (Güler et al. 2018) also provides sparse correpondences of pixels in the image to body mesh.

Annotations on in-the-wild images in these 2D pose datasets are manually created. However, it is not possible to create 3D pose annotations on in-the-wild images manually at a scale large enough to be usable for learning by CNN-based methods. Some datasets such as MarCOnI (Elhayek et al. 2016) have 3D pose annotations generated manually through triangulation in a multi-view setup, but such an approach only scales to a lower number of scenes and subjects. Chapter 3 discusses the capture approach proposed in this thesis to increase the appearance diversity of 3D pose datasets.

2.2 3D Body Pose Estimation

2.2.1 Multi-view 3D Pose Estimation Methods

In multi-view setups marker-less motion-captue solutions attain high accuracy. Tracking of a manually initialized actor model from frame to frame with a generative image formation is common. See Thomas B. Moeslund et al. 2006 for a complete overview. Most methods target high quality

with offline computation (Bregler et al. 1998; Howe et al. 1999; M. M. Loper et al. 2014; Sidenbladh et al. 2000; Starck et al. 2003). Real-time performance could be attained by representing the actor with Gaussians (Rhodin et al. 2015; Stoll et al. 2011; Wren et al. 1997), skeletonizing visual hulls (Bakken et al. 2012), and other approximations (Ma et al. 2014), in addition to clever formulations of model-to-image fitting. Per-frame approaches (Bakken et al. 2012) avoid some of the issues faced by tracking based approaches, which often lose track in local minima of the non-convex fitting functions they optimize and require separate initialization, e.g. using Bogo et al. 2016; Rhodin et al. 2016b; Sminchisescu et al. 2001. Robustness could be increased with a combination of generative and discriminative estimation (Elhayek et al. 2016), even from a single input view (Rosales et al. 2006; Sminchisescu et al. 2006), and egocentric perspective (Rhodin et al. 2016a). Recent approaches leveraging convolutional neural networks (Joo et al. 2018; Trumble et al. 2016), together with statistical and geometric (Bridgeman et al. 2019; Iskakov et al. 2019; Simon et al. 2017), insights and various body modeling (Joo et al. 2018; Trumble et al. 2018) approaches, have further improved the fidelity of capture. Other work (Gilbert et al. 2019; Malleson et al. 2019, 2017) has combined inertial sensing with multi-viewpoint video for improved 3D pose estimates.

2.2.2 Depth Sensing Based 3D Pose Methods

The additional depth channel provided by RGB-D sensors led to very robust real-time pose estimation solutions (Baak et al. 2011; Ganapathi et al. 2012; Ma et al. 2014; Shotton et al. 2013; X. Wei et al. 2012; M. Ye et al. 2014), but at the cost of special sensors that do not work under all conditions. The availability of these sensors as consumer grade devices enabled a range of new applications. Q. Liu et al. 2015 further augmented the depth modality with binaural audio to ease tracking of subjects under occlusion. Even real-time tracking of general deforming objects (Zollhöfer et al. 2014) and template-free reconstruction (Dou et al. 2016; Innmann et al. 2016; Newcombe et al. 2015; Orts-Escolano et al. 2016) has been demonstrated.

Mutli-view depth-based reconstruction techniques obtain even higher accuracy and robustness (Dou et al. 2016; Orts-Escolano et al. 2016). RGB-D information overcomes forward-backwards ambiguities in monocular pose estimation, however, active IR-based sensors preclude applications in direct sunlight. Further, mobile applications would suffer from their high energy consumption, and they are not yet as widely available as RGB cameras build in every smartphone. Some of these limitations can be overcome with stereo cameras, but it requires textured objects and a sufficiently wide camera baseline, which may be impractical for consumer-level applications.

Hence, in this thesis, the focus is on monocular RGB based solutions for 3D body pose estimation.

2.2.3 Monocular RGB Based 3D Pose Estimation

Analysis-by-synthesis approaches to monocular motion capture employ a parametric model of the human body, which can be rendered into an image that is compared to the input image to update the parameters in an iterative fashion. The model can be an abstract representation, such as a stick figure, or a detailed pose and shape model, and the synthesized image can be a photorealistic image that gets compared directly to the input image, or more abstract image which gets compared to an equivalent abstraction of the input image. Common abstractions include edges, silhouettes and contours, stick figure representations etc. Refer to Thomas B Moeslund 1999 for a survey of analysis-by-synthesis approaches to monocular motion capture. Monocular generative motion capture has only been shown for short clips and when paired with strong motion priors (Urtasun et al.

2006) or in combination with discriminative re-initialization (Rosales et al. 2006; Sminchisescu et al. 2006), since generative reconstruction is fundamentally underconstrained. Using photo-realistic template models for model fitting enables more robust monocular tracking of simple motions, but requires more expensive offline computation (La Gorce et al. 2008). Sampling-based methods avoid local minima (Balan et al. 2005; Bo et al. 2010; Deutscher et al. 2005; Gall et al. 2010). However, real-time variants can not guarantee global convergence due to a limited number of samples, such as particle swarm optimization techniques (Oikonomidis et al. 2011). Structure-from-motion techniques exploit motion cues in a batch of frames (Garg et al. 2013), and have also been applied to human motion estimation (Gotardo et al. 2011; Lee et al. 2013; H. S. Park et al. 2011; Zhu et al. 2011). However, batch optimization does not apply to the real-time setting where frames are streamed sequentially. For some applications manual annotation and correction of frames is suitable, for instance to enable movie actor reshaping (Jain et al. 2010) and garment replacement in video (Rogge et al. 2014). In combination with physical constraints, highly accurate reconstructions are possible from monocular video (X. Wei et al. 2010). Vondrak et al. 2012 succeed without manual annotation by simulating biped-controllers, but require batch-optimization. While these methods can yield high-quality reconstructions, interaction and expensive optimization preclude live applications.

Notably, only few methods target real-time monocular reconstruction. Exceptions are the regression of 3D pose from Haar features by Bissacco et al. [2007] and detection of a set of discrete poses from edge direction histograms in the vicinity of the previous frame pose (Taycher et al. 2006). Both only obtain temporally instable, coarse pose, not directly usable in our applications. Chai and Hodgins obtain sufficient quality to drive virtual avatars in real-time, but require visual markers (Chai et al. 2005).

Buoyed by the success of CNN-based methods for 2D pose estimation, several CNN-based approaches for 3D pose estimation have come to the fore before and in-parallel with the approaches proposed in this thesis. The related approaches (classic and CNN-based) can be broadly categorized as follows:

2D-to-3D 'Lifting' A common strategy for 3D articulated pose estimation is lifting 2D pose or joint position predictions to 3D, e.g. Mori et al. 2006; C. J. Taylor 2000. Here, model-based optimization is often required in addition to (iteratively) optimize the projection of a 3D human model to explain the 2D predictions. This is computationally expensive, but allows incorporation of pose priors or inter-penetration constraints (Bogo et al. 2016), sparsity assumptions (C. Wang et al. 2014; Xiaowei Zhou et al. 2015a,b), joint limits (Akhter et al. 2015; Elhayek et al. 2016; Rhodin et al. 2016a), and temporal constraints (Rhodin et al. 2016b). An alternative to iterative optimization is sampling. Simo-Serra et al. 2012 sample 3D pose from 2D predictions and improve discriminative 2D detection from likely 3D samples (Simo-Serra et al. 2013). S. Li et al. 2015 classify the nearest training examples. 2D-3D lifting is related to non-rigid and articulated structure-from-motion from image sequences (Paladini et al. 2012; H. S. Park et al. 2011) where sub-space motion or pose models, such as bilinear models (F. Zhou et al. 2014), are assumed.

Regression from 2D detections to 3D pose alleviates expensive optimization and sampling (Martinez et al. 2017; Yasin et al. 2016) and works on individual images. Some methods treat 3D pose as a hidden variable, integrate the 3D to 2D projection model into the regression function, and enforce a prior on the hidden variable (Brau et al. 2016). While 2D joint locations may reveal information about the subject shape (Bogo et al. 2016), this abstraction to keypoints loses vital image information.

Other work has proposed to augment the 2D pose with relative depth ordering of body joints as additional context to disambiguate 2D to 3D lifting (Pavlakos et al. 2018a; Pons-Moll et al. 2014).

Image to 3D Pose Estimation To overcome the depth ambiguity associated with 2D-to-3D lifting, discriminative methods have been proposed that learn implicit depth features for 3D pose directly from more expressive image representations. Rosales et al. 2000 regress 3D pose from silhouette images using a probabilistic formulation that captures the multi-modality of pose-to-silhouette mapping, A. Agarwal et al. 2006 use linear regression , and Elgammal et al. 2004 employ a joint embedding of images and 3D pose. Sminchisescu et al. 2007 further utilized temporal consistency to propagate pose probabilities with a Bayesian mixture of experts Markov model. With the recent advancement of machine learning techniques and compute capabilities, 3D pose has been directly regressed from the image, through structured learning of latent pose (S. Li et al. 2015; Bugra Tekin et al. 2016a), joint prediction of 2D and 3D pose (S. Li et al. 2014; B. Tekin et al. 2017; Yasin et al. 2016), novel pose space formulations (Pavlakos et al. 2017) and classification over example poses (Pons-Moll et al. 2014; Grégory Rogez et al. 2016). Relative per-bone predictions (S. Li et al. 2014), kinematic skeleton models (Xingyi Zhou et al. 2016), or root centered joint positions (Ionescu et al. 2014a) are used as the eventual output space. Such direct 3D pose regression methods capture depth relations well, but 3D estimates usually do not accurately match the true 2D location when re-projected to the image, because estimations are done in cropped images that lose camera perspective effects, using a canonical height, and minimize 3D loss instead of projection to 2D. Furthermore, they only deliver joint positions, are temporally unstable, and none has shown real-time performance.

Methods that directly regress 3D pose (Ionescu et al. 2014a; S. Li et al. 2014; Bugra Tekin et al. 2016a; Xingyi Zhou et al. 2016) commonly crop the input image to the bounding box of the subject, use the 3D joint position relative to the pelvis as output, and normalize subject height (Ionescu et al. 2014a). In controlled conditions, actor silhouette and fixed camera placement provides additional height cues (Y. Yu et al. 2016).

Some recent methods integrate a 3D body model (M. Loper et al. 2015) within a network to predict 3D pose and shape from single images (Kanazawa et al. 2018; Omran et al. 2018; Pavlakos et al. 2018b; Tung et al. 2017). Other approaches optimize a body model or a template (Habermann et al. 2019; Xu et al. 2018) to fit 2D poses or/and silhouettes (Alldieck et al. 2019, 2018a,b; Bogo et al. 2016; Lassner et al. 2017). None of them handles multiple people.

Multi-Person 3D Pose Earlier work on monocular multi-person 3D pose capture often followed a generative formulation, e. g. estimating 3D body and camera pose from 2D landmarks using a learned pose space (Ramakrishna et al. 2012).

Gregory Rogez et al. 2017 use a detection-based approach and first find representative poses of discrete pose clusters that are subsequently refined. Predicting multiple proposals per individual and fusing them afterwards is time consuming and may incorrectly merge nearby individuals with similar poses. The LCRNet++ implementation of this algorithm uses a ResNet-50 [2016] base network and achieves non-real-time interactive $10-12$fps on consumer hardware even with the faster but less accurate 'demo' version that uses fewer anchor poses. The approach for multi-person pose estimation proposed in the thesis jointly encodes the 2D and 3D pose of all subjects in the scene in the same feature volume, to ensure that the inference time does not scale linearly with the number of subjects in the scene. The formulation ameliorates inevitable encoding conflicts and provides robustness to occlusions through the incorporation of selective redundancy. Similarly, Zanfir et al. 2018b jointly encode the 2D and 3D pose of all subjects in the scene using a fixed number of feature maps. Different from the proposed approach they encode the full 3D pose vector at all the projected pixels of the skeleton, and not just at the body joint locations, which makes the 3D feature space rife

with potential encoding conflicts. For association, they learn a function to evaluate limb grouping proposals. A 3D pose decoding stage extracts 3D pose features per limb and uses an attention mechanism to combine these into a 3D pose prediction for the limb. The approach of Zanfir et al. 2018a also combines learning and optimization, but their space-time optimization over all frames is not real-time.

Temporal Inference Temporal information in videos gives additional cues and increases accuracy (Bugra Tekin et al. 2016b; Xiaowei Zhou et al. 2015c), but conditioning on motion increases the input dimension and requires motion databases with sufficient motion variation, which may be even harder to capture than pose data sets. Some approaches reduce ambiguity in 2D-to-3D lifting through the use of temporal information (Pavllo et al. 2019).

The approaches proposed in the thesis rely on a range of novel insights to enable the first real-time monocular RGB based motion capture systems for single-person and multi-person scenarios. Contributions towards novel training datasets and transfer learning insights described in Chapter 3 and Chapter 4 allow the systems to operate in a diverse range of scene settings. Contributions towards a novel pose formulation which links 3D pose inference per body part to direct image evidence, resulting in more accurate predictions as compared to naïve image-to-pose-vector prediction, is described in Chapter 5. These insights are further extended to occlusion robust pose formulations, which incorporate selective redundancy to allow pose inference under occlusion. These formulations, described in Chapter 6, also jointly encode the 3D pose of all subjects in the scene without needing to scale the output dimensions, allowing fast inference even with a large number of subjects in the scene. Real-time inference is achieved through contributions towards the design of faster CNN architectures, as described in Chapter 7, and system design insights, as described in Chapter 8. The combination of learning-based approaches for per-frame inference, and model-based kinematic fitting approaches, as described in Chapter 8, allows reliable motion capture in general scenes even under occlusions, resulting in temporally smooth joint angle estimates which can be readily employed in various applications.

2.3 Fast Inference With Neural Networks

Neural network solutions to image based problems primarily employ convolutional layers as building blocks, which are parameter efficient but make up most of the compute cost in these networks.

Several network pruning and compression solutions have targeted the dense fully connected layers (Z. Yang et al. 2015) at the output of classification networks, achieving massive parameter reduction, which does not translate into equivalent compute speed gains (Wen et al. 2016). Non structured pruning of weights can also achieve significant compression rates without an equivalent speedup on off-the-shelf hardware (S. Han et al. 2016, 2015).

Prior art on on speeding up neural network execution has hence focused on seven broad fronts, which are to some degree complementary to each other. For a detailed survey on neural network acceleration approaches, refer to Cheng et al. 2017. Here, the seven are briefly listed: The first one concerns feature/filter level pruning or sparsification, either while training the weights (Hanson et al. 1989; Z. Liu et al. 2017) or in a post-hoc manner (Molchanov et al. 2017; Mozer et al. 1989; Theis et al. 2017). The second concerns network architecture search approaches for determining the network structure for a particular problem (Elsken et al. 2018; C. Liu et al. 2017; Pham et al. 2018; Zoph et al. 2016), using learned combinations of pre-determined computational blocks. The third

concerns reduction in the cost of the elemental operations through quantization (Hubara et al. 2016; D. Lin et al. 2016; Wu et al. 2016) or binarization of the weights (Hubara et al. 2016; Rastegari et al. 2016), and through algorithms such as Winograd (Lavin et al. 2016) and FFT (Vasilache et al. 2014) for faster convolutions. The fourth pertains to low-rank decomposition of the learned weight tensors (Wenlin Chen et al. 2015; Denton et al. 2014; Jaderberg et al. 2014; Y.-D. Kim et al. 2015; Lebedev et al. 2014; Wen et al. 2017). The fifth concerns learned group sparsity in the input weights of nodes of a particular layer (G. Huang et al. 2018; Lebedev et al. 2016; Scardapane et al. 2017; Wen et al. 2016), which can be exploited on off-the-shelf hardware. The sixth concerns techniques such as knowledge distillation (Hinton et al. 2015) which instead focus on better training skinnier and shallower networks, guided by the internal representations of a more expressive model trained on the same task (BuciluǍ et al. 2006; Romero et al. 2014; Zagoruyko et al. 2016). The last one falls in the domain of empirical determination of representational structure, with techniques like bottleneck modules, depth-wise convolutions, pre-determined structured sparsity etc. (Bagherinezhad et al. 2017; Howard et al. 2018; Iandola et al. 2016; S. Mehta et al. 2018; Szegedy et al. 2017; M. Wang et al. 2017) used for computational savings.

Feature pruning/sparsification, and building blocks for fast neural network inference are examined in detail here because these are the most pertinent to the findings on implicit sparsification and fast Convolutional Neural Network design proposal in Chapter 7.

2.3.1 Heuristics for Feature Pruning/Sparsification

Neural network sparsification can be broadly grouped into synaptic sparsification and neuronal sparsification. The former entails removal of a subset of weights associated with a particular neuron, and the latter sees the removal of all weights associated with a particular neuron.

Determining synaptic importance requires estimating individual parameter saliency to identify the least useful parameters. The estimates used range from crude measures such as parameter magnitudes (S. Han et al. 2016) to more sophisticated and expensive to compute second order estimates (Hassibi et al. 1993; LeCun et al. 1990). These methods iteratively prune and finetune the network. The unstructured sparsity pattern resulting from synaptic sparsification does not lend itself to faster inference on existing CPU and GPU hardware.

On the other hand neuronal importance approaches for post-hoc pruning remove whole features instead of individual parameters, and are more amenable to acceleration on existing hardware. Filter/feature level pruning is based on the observation that overparameterized neural networks, particularly wider neural networks, are easier to optimize in practice than skinny neural networks (Arora et al. 2018; Haeffele et al. 2015; Mozer et al. 1989), and the redundant features can be identified and removed during or after training.

The filter importance heuristics employed include weight norm of a filter (H. Li et al. 2017; Srinivas et al. 2016), the average percentage of zero outputs of a node over the dataset (H. Hu et al. 2016), feature saliency based on first (Molchanov et al. 2017; Mozer et al. 1989) and second order (Theis et al. 2017) Taylor expansion of the loss or a surrogate (Mozer et al. 1989). The feature computation cost can also be included in the pruning objective (Molchanov et al. 2017; Theis et al. 2017).

Some post-hoc pruning approaches are based on preserving the outputs of the subsequent layer (Luo et al. 2017), merging of neurons with similar weights (Srinivas et al. 2016), non structured sparsity guided by awareness of potential speedup (J. Park et al. 2017) and feature pruning based on the learned scale values in the scale layer after Batch Norm (J. Ye et al. 2018).

Methods for sparsity while training include the approach of Hanson et al. 1989 which uses all the weights of a feature to dictate the weight decay of each individual weight of the feature, coupled with non-uniform penalization of weight magnitudes to learn sparse features. Chauvin 1989 achieve feature sparsity through penalization of activations of hidden nodes. Guo et al. 2016 learn a binary mask with the weights (non structured) which allows weight threshold based pruning while the weight value itself is retained for possible revival in future iterations. Z. Liu et al. 2017 use lasso regularization on the scale layer after Batch Norm to induce feature sparsity. Wen et al. 2017 propose to nudge correlated filters to collapse during training to induce low rank feature representations. Wen et al. 2016 use predefined low rank decompositions with group lasso regularization inducing additional sparsity. The formulation of G. Huang et al. 2018 forces the same connectivity pattern to be learned for all features in predefined groups. Scardapane et al. 2017 use group lasso penalty to learn group sparse features. Lebedev et al. 2016 learn the same non-structured sparsity for each group of features in a particular layer, to facilitate speed up through faster matrix multiplication based convolution.

Chapter 7.1 presents some surprising results on extensive filter level sparsity emerging implicitly in common Convolutional Neural Network training setups, and ties the mechanisms causing this phenomenon to various explicit sparsification heuristics.

2.3.2 The Design of Fast Neural Networks

Various building blocks have been proposed for Convolutional Neural Network design to enable efficient inference while promoting information flow through the network, and achieving a large receptive field. Some examples are shown in Figure 2.4.

Bottleneck blocks (K. He et al. 2016a) are designed to reduce the number of channels for costly 3×3 convolutions, and variants of it have been used in other network architectures as well (Szegedy et al. 2017). Xception (Chollet 2017) and MobileNetV1/2 (Howard et al. 2018; Sandler et al. 2018) utilize depth-wise 3×3 convolutions in conjunction with point-wise (1×1) convolutions to reduce inference cost. Other architectures employ a compromise between depth-wise convolutions and vanilla convolutions, known as grouped convolutions (Szegedy et al. 2017) for a better tradeoff between computation cost and accuracy. Other approaches (Romera et al. 2018) use separable convolutions to reduce the computation cost of 3×3 convolutions. Beyond the building blocks, the connectivity patterns of the building blocks dictate information flow through the network. Residual connectivity (K. He et al. 2016a; Srivastava et al. 2015) has enabled the training of very deep network architectures, an order of magnitude or more deeper than non-residual architectures (Simonyan et al. 2014). Other connectivity patterns such as concatenative skip connections in DenseNets (G. Huang et al. 2017) show better information flow through the network, achieving better accuracy in practice with lower network depth. However, the dense connectivity pattern is slow in inference due to its large memory footprint. Other architectures (S. Mehta et al. 2018) use a combination of such connectivity patterns, in conjunction with dilated convolutions, in order to achieve faster inference with large receptive fields. However, dilated convolutions suffer from gridding artefacts in the outputs.

The network architecture proposed in this thesis (Chapter 7.2.3) takes inspiration from the success of various building blocks and connectivity patterns and pays heed to their limitations, in order to achieve faster inference speeds with a lower memory footprint, at a given level of accuracy.

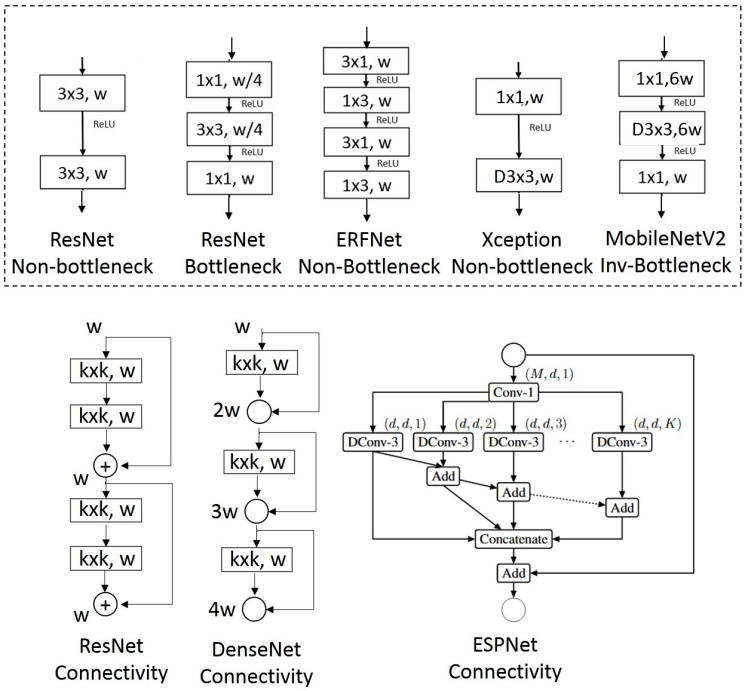

Figure 2.4: Examples of common building blocks (top) employed in various CNN architectures, as well as typical connectivity patterns (bottom) these building blocks are combined with. Shown here are variants of ResNet (K. He et al. 2016a) building blocks, ERFNet (Romera et al. 2018) block with spatially separable convolutions, Xception (Chollet 2017) block with a combination of pointwise convolution and depthwise separable convolution, and MobileNetV2 (Sandler et al. 2018) with inverted-bottleneck block using a combination of depthwise separable and pointwise convolutions. Convolutional layers or blocks combining various convolutional layers, as above, can be connected together in various different ways, some examples of which are: additive residual skip-connectivity as in ResNet (K. He et al. 2016a), dense concatenative skip-connectivity as in DenseNet (G. Huang et al. 2017), or combinations of the two as in the hierarchical feature fusion module of ESPNet (S. Mehta et al. 2018). The design goal behind the building blocks and connectivity patterns is to promote information flow through the network and achieve a large receptive field, while minimizing the compute cost.

Chapter 3

Capturing Annotated 3D Body Pose Data

As discussed in Chapter 1 and Chapter 2, 3D body pose estimation from monocular RGB input is a difficult and ill posed problem. Learning based approaches leveraging large annotated data corpora have been very successful at the related problem of 2D pose estimation on in-the-wild scenes, and have shown promising results for 3D pose estimation in constrained scene settings. However, their application to 3D pose estimation in general scene settings has not been possible largely due to limitations of the available data corpora. Manually annotating 2D body joint locations on in-the-wild human images is possible, and considerable effort has been put in to organize collection of such annotations in a scalable way (Mykhaylo Andriluka et al. 2014). However, manual annotation of 3D body poses at scale is cumbersome (Elhayek et al. 2016), and obtaining such annotations typically requires a motion-capture setup with external and/or on-body instrumentation. Thus datasets with 3D pose annotations are starkly limited in terms of scene diversity, and are primarily restricted to indoor settings with a single unoccluded subject.

This chapter proposes new training datasets to close the gap to real world scene settings by providing the means to increase appearance diversity, and simulate multi-person scene settings. Further, to evaluate the generalization performance of methods trained on variants of the proposed training dataset, this chapter proposes single-person and multi-person test datasets which comprise of real scenes in a variety of indoor and outdoor settings, and are notably different in appearance from the training dataset. These benchmarks support fine-grained evaluation by activity, by body joint type, and by body joint visibility.

In This Chapter

- Discussion of the limitations of the existing 3D body pose datasets (Section 3.1)
- Description of the capture setting and features of the proposed single-person 3D pose dataset, MPI-INF-3DHP (Section 3.2.1)
- Description of the process of creating annotated multi-person composite scenes (MuCo-3DHP) from the proposed single-person dataset (Section 3.2.2)
- Description of the single-person 3D pose test dataset (MPI-INF-3DHP) and the proposed 3D Percentage of Correct Keypoint (3DPCK) metric (Section 3.3.1)

- Description of the multi-person 3D pose test dataset, MuPoTS-3D (Section 3.3.2)

The content of this chapter is based on D. Mehta et al. 2017a and D. Mehta et al. 2018.

3.1 Related 3D Body Pose Datasets: Overview and Shortcomings

Existing large scale datasets with images of humans annotated with 3D body pose, such as Human3.6m (Ionescu et al. 2014b), HumanEva (Sigal et al. 2010) and Total Capture (Trumble et al. 2017), utilize marker-based motion capture for 3D annotation. This restricts the type of clothing worn by the subjects to skin-tight clothing, and increases the pre-capture setup time per subject because the markers need to be put on carefully. The datasets are limited to a single subject, and the scene appearance is restricted to the studio setup, and hence is starkly limited. Alternatively, large datasets with markerless capture using a dome of hundreds of cameras, such as Panoptic (Joo et al. 2015), enable more diverse clothing, and include multi-person scenarios. However, the scene setting is still starkly limited, and the captured multi-person scenarios, while diverse, do not adequately cover combinatorially many possible multi-person scenarios. Recent work (Pavlakos et al. 2018a) has proposed manual labeling of body joint depth ordering constraints on in-the-wild images as an amenable alternative to 3D pose annotations, however this only serves as a weak supervision signal, and the resulting accuracy is not at par with methods trained with 3D pose annotations.

Manual annotation of multi-view data to obtain 3D annotations through triangulation was only scaled to small corpora, such as MarCOnI (Elhayek et al. 2016). In general, real recordings are hardly scalable to tens of thousands of examples required to explain the vast variability of human pose and appearance. The 3D Poses in the Wild (3DPW) dataset (Marcard et al. 2018) features multiple people outdoors recorded with a moving camera and includes ground truth 3D pose. The number of subjects is however limited. Synthesizing example images is an alternative, however it may limit generalization to real scenes (Wenzheng Chen et al. 2016; Ionescu et al. 2014b; Grégory Rogez et al. 2016; Varol et al. 2017) due to over-simplified animation and rendering. To obtain 3D annotations for in-the-wild 2D pose datasets, Gregory Rogez et al. 2017 use 2D to 3D lifting to create pseudo annotations on the MPII 2D pose dataset (Mykhaylo Andriluka et al. 2014). However, these pseudo annotations suffer from a loss of accurancy on account of depth ambiguity associated with 2D-to-3D lifting approaches, as discussed in Chapter 2.

This systemic lack of large and diverse in appearence 3D human pose data corpora notably constrains accuracy and generalization of prior methods, and strongly motivates the dataset (Chapter 3) and training schema (Chapter 4) contributions of this work.

3.2 Training Datasets for 3D Body Pose Estimation

First, the data capture approach for single-person scenarios is described in Section 3.2.1. The data capture approach is designed for diverse pose and clothing capture, and the setup eases augmentation to extend the captured appearance variability of clothing and background.

Then the captured data for the single-person case is scaled to combinatorially many multi-person scenarios, as described in Section 3.2.2.

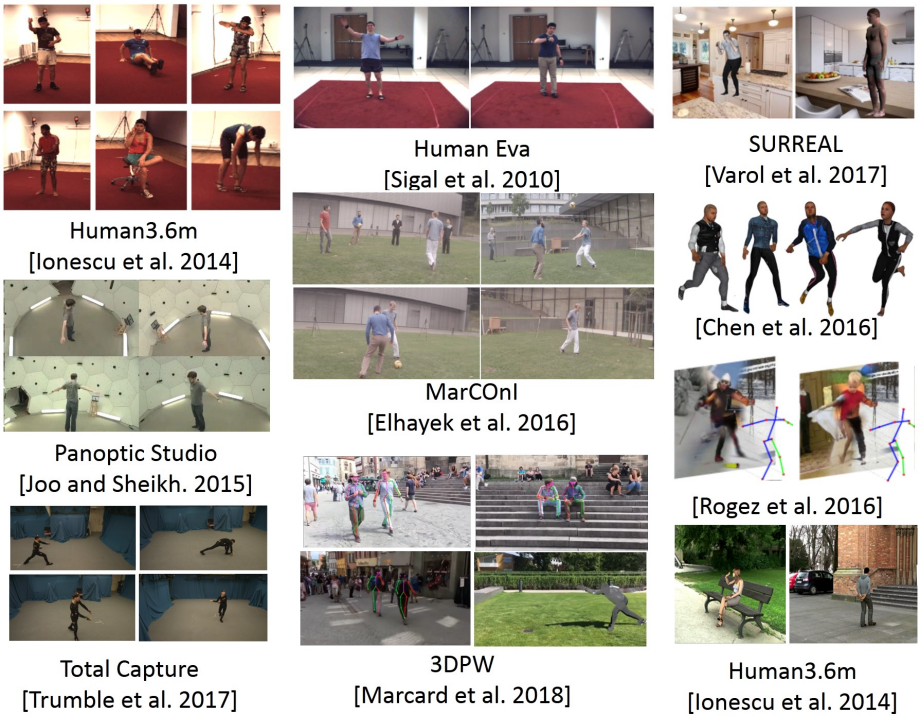

Figure 3.1: Representative frames from several datasets with person images annotated with 3D body pose information. Datasets captured with multi-view setups indoors (Ionescu et al. 2014b; Joo et al. 2015) are starkly limited in terms of appearance diversity, and multi-view setups outdoors (Elhayek et al. 2016) are starkly limited both in terms of number of subjects as well as scene appearance diversity on account of manual annotation. Synthetic approaches (Wenzheng Chen et al. 2016; Grégory Rogez et al. 2016; Varol et al. 2017) create large diversity in scene appearance, but the renderings have a significant domain gap from real scenes.

3.2.1 MPI-INF-3DHP: Single-Person 3D Pose Dataset

Capture Configuration: The dataset is captured in a multi-camera studio with ground truth from a commercial marker-less motion capture (*The Captury* 2016). No special suits and markers are needed, allowing the capture of motions wearing everyday apparel, including relatively loose clothing. In contrast to existing datasets, the dataset is captured with a green-screen background to allow automatic segmentation and augmentation. For detailed considerations concerning the design and setup of a multi-view markerless body shape and appearance capture studio, refer to Starck et al. 2009. The 14 cameras of the dataset cover a wide range of viewpoints, with five cameras mounted at chest height with a roughly 15° elevation variation similar to the camera orientation jitter in other datasets (Wenzheng Chen et al. 2016). Another five cameras are mounted higher and angled down 45°, three more have a top down view, and one camera is at knee height angled up. Figure 3.3 shows the distribution of camera viewpoints in the capture setup, as well as representative images from a subset of the cameras. The cameras record 4 MP 1:1 aspect ratio images at up to 50 FPS.

Actors and Activities 8 actors (4 male + 4 female) are recorded as part of the training set, performing 8 activity sets each, ranging from walking and sitting to complex exercise poses and dynamic actions. The activity types and prompts are designed to span more diverse *pose* classes than existing pose corpora such as Human3.6m. Each activity set spans roughly one minute. Each

Figure 3.2: The MPI-INF-3DHP training set is comprised of 8 actors. Here, each actor is visualized in both sets of clothing in which the actor was recorded. One set is normal street wear, while the other set is purposefully chosen to have uniformly colored upper and lower body clothing such that they can be independently chroma-keyed for augmentation.

actor features 2 sets of clothing split across the activity sets. One clothing set is *casual everyday apparel*, and the other is *plain-colored* to allow augmentation as shown in Figure 3.2. See Table 3.1 for a detailed breakdown. Figure 3.5 shows example poses from various activity sets. Refer to Appendix A for the prompts used to guide the actors through the activities.

Overall, from all 14 cameras, roughly 1.5M frames are captured, 500k of which are from the five chest high cameras. In addition to the true 3D and 2D pose annotations, 3D poses of a height-normalized skeleton compatible with the 'universal' skeleton of Human3.6M are also made available. The normalization scales the skeleton such that the knee-to-neck height (sum of the lengths of the thigh and spine) is 930mm.

Dataset Augmentation Although the captured dataset out of the box has more clothing variation than other single-person datasets such as Human3.6m [2014], the appearance variation is still not comparable to in-the-wild images. The augmentation approach proposed in this work is inspired by prior work which uses images to augment the background of recorded footage (Wenzheng Chen et al. 2016; Ionescu et al. 2014b; Rhodin et al. 2016a), and recolors plain-color shirts (Rhodin et al. 2016a) while keeping the shading details, using intrinsic image decomposition to separate reflectance and shading (Meka et al. 2016). Chroma-key masks are extracted for the background, a chair in the scene, as well as upper and lower body segmentation for the plain-colored clothing sets. This provides an increased scope for foreground and background augmentation, in contrast to the marker-less recordings of Joo et al. 2015. The chroma-key masks are obtained with the Nuke (Nuke 2015) visual effects software. See Figure 3.4 for example masks as well as augmentation results. The background is simply replaced with images representing indoor and outdoor scenes. For clothing and chair augmentation, the luminance of the masked regions is used as a proxy for shading, and is blended with images of cloth patterns and textures.

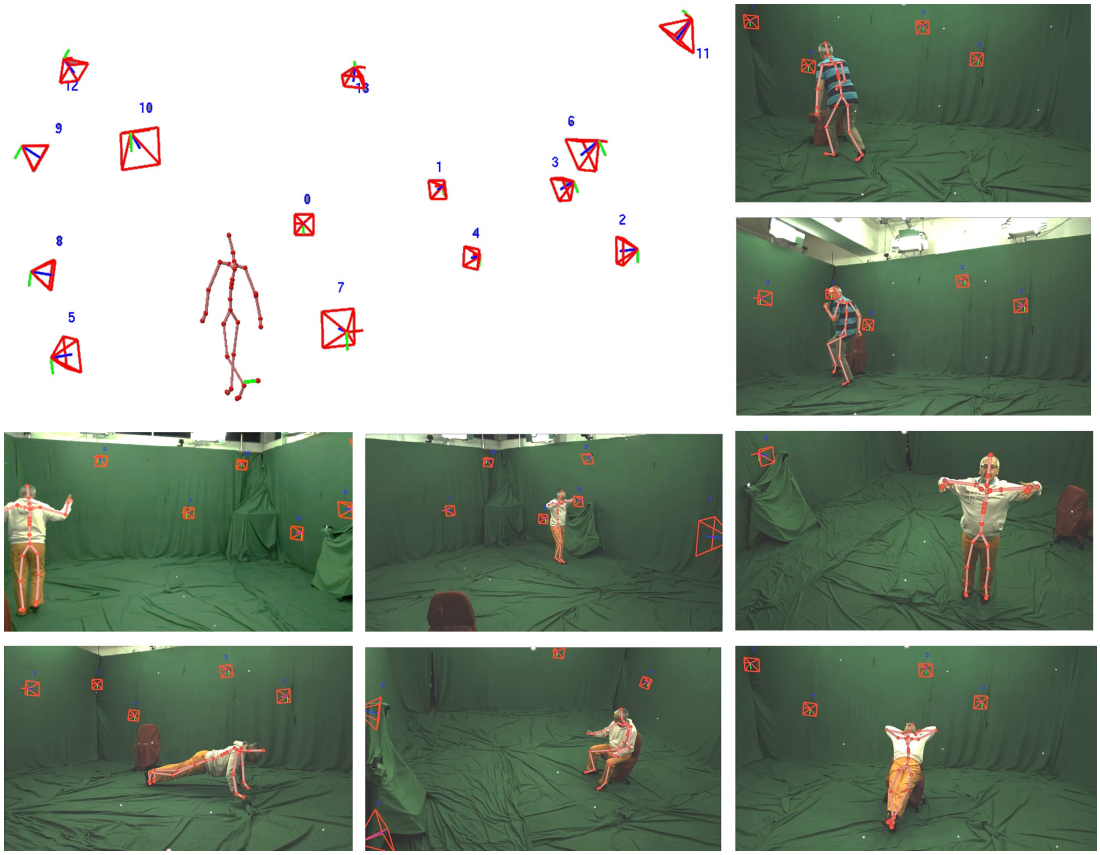

Figure 3.3: Visualization of the camera viewpoints available for the proposed MPI-INF-3DHP single person dataset. Also shown are images from a subset of the viewpoints with the orientations of the visible cameras overlaid. The dataset is captured with a green-screen background such that it can be chroma-keyed and augmented with various images. The chair is covered with a red cloth such that it can be independently chroma-keyed and augmented.

Table 3.1: MPI-INF-3DHP training dataset is comprised of 8 actors recorded from 14 camera viewpoints, performing 8 activity sets each. The activities are each 1 minute long, and grouped into 2 sets of 4 minutes each. The actors wear casual everyday apparel (Street) and plain-colored clothes (Plain) to allow clothing appearance augmentation. Overall, 1.5M frames from a diverse range of viewpoints are available, capturing a diverse range of poses and activities. Through the extensive avenues of background and clothing appearance augmentation made available, the number of effective frames available for training can be increased combinatorially. All cameras record at a 2048×2048 pixel resolution.

Actor ID	Actor Gender	# Frames		Clothing		FPS		Total Frames
		Seq1	Seq2	Seq1	Seq2	Seq1	Seq2	
S1	F	6416	12430	Street	Plain	25	50	260k
S2	M	6502	6081	Street	Plain	25	25	175k
S3	M	12489	12283	Street	Plain	50	50	345k
S4	F	6171	6675	Street	Plain	25	25	175k
S5	F	12820	12312	Street	Plain	50	50	350k
S6	F	6188	6145	Street	Plain	25	25	170k
S7	M	6239	6320	Plain	Street	25	25	170k
S8	M	6468	6054	Plain	Street	25	25	170k

Figure 3.4: Avenues of appearance augmentation in MPI-INF-3DHP dataset. Actors are captured using a markerless multi-camera setup in a green-screen studio (left), and segmentation masks computed for different regions (center left). The captured footage can be augmented by compositing different textures to the background, chair, upper body and lower body areas, independently (center right and right).

Figure 3.5: Representative frames from MPI-INF-3DHP training set, showing different subjects in different clothing sets and poses from different activity sets as well as the scope of appearance augmentation made possible by the dataset.

The efficacy of this increased scope of augmentation at enabling in-the-wild performance of learning based 3D pose estimation approaches is demonstrated throughout the thesis, starting with Chapter 4.

Additional Data Captured: In addition to the 14 synchronized camera viewpoints, another synchronized chest high camera records with a fish-eye lens. An RGB-D camera (ASUS 2011), also placed at chest height and pointed forward records RGB (1280×1024 px) and depth images (640×480 px) at 30 FPS. The RGB-D camera, however, is not frame synchronized with the remaining cameras. 3D body surface scans of each subject are also captured with a laser scanner. See Appendix A for details.

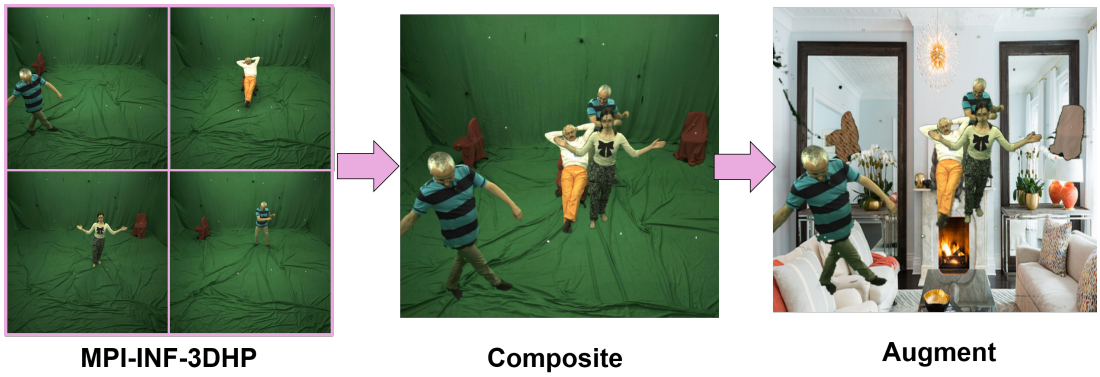

Figure 3.6: The process of creation of the multi-person composited 3D human pose dataset MuCo-3DHP from per-camera image samples from the single-person MPI-INF-3DHP dataset. The images are composited in a depth-aware manner using the 3D pose annotations made available in MPI-INF-3DHP. Appearance diversity can be greatly amplified by augmenting the background as well as clothing appearance.

3.2.2 MuCo-3DHP: Multi-person Composited 3D Human Pose Dataset

Multi-person 3D motion capture under strong occlusions and interactions is challenging even for commercial systems, often requiring manual pose correction constraining 3D accuracy. This severely limits the scale at which real multi-person data can be captured and processed. Thus, to enable a much larger training set, the thesis proposes a new compositing and augmentation scheme that leverages the single-person image data of real people in MPI-INF-3DHP to composite an arbitrary number of multi-person interaction images under user control, with 3D pose annotations.

Figure 3.7: Examples from the proposed multi-person composited training dataset MuCo-3DHP. Ground truth 3D pose reference as well as the full scope of appearance augmentation offered by the single-person MPI-INF-3DHP dataset are brought to bear on multi-person scenarios.

Creating Multi-person Composites As discussed in the previous section, the captured MPI-INF-3DHP single-person 3D pose dataset provides marker-less motion capture based annotations for real

Table 3.2: MPI-INF-3DHP test dataset is comprised of 6 sequences in different scene settings, 4 of which are markedly different from the training dataset. The intention is to encourage the development of approaches which generalize beyond the scene settings present in the training dataset.

Test Seq.	Total Frames	Evaluated Frames	Scene Setting	Image Resolution
TS1	6151	603	Studio GS	2048x2048
TS2	6080	540	Studio GS	2048x2048
TS3	5838	506	Studio No GS	2048x2048
TS4	6007	558	Studio No GS	2048x2048
TS5	320	276	Outdoors	1920x1080
TS6	492	392	Outdoors	1920x1080

images of 8 actors, each captured in a green-screen studio with 2 clothing sets in the same physical space with the same 14 camera setup. Chroma-keying the green-screen background provides person segmentation masks, which, together with the ground-truth 3D skeleton pose annotations can be leveraged to create per-camera composites with 1 to 4 subjects, with frames randomly selected from the 8×2 sequences available per camera. The ground-truth absolute pose information is used to create 3D-aware composites, resulting in correct depth ordering and overlap of subjects. See Figure 3.6. The compositing process results in plausible images covering a range of simulated inter-person overlap and activity scenarios. Furthermore, user-control over the desired pose and occlusion distribution, and foreground/background augmentation using the masks provided with MPI-INF-3DHP is possible. See Figure 3.7 for examples of augmented multi-person composites.

Even though the synthesized composites may not simulate all the nuances of human-human interaction fully, Chapter 6 shows that the various proposed approaches trained on this data generalize well to real world scenes.

3.3 3D Pose Test Datasets

3.3.1 MPI-INF-3DHP Single Person 3D Pose Test Set

The existing test sets for (monocular) 3D pose estimation were found to be restricted to limited settings due to the difficulty of obtaining ground truth labels in general scenes. The *HumanEva* (Sigal et al. 2010) and *Human3.6M* (Ionescu et al. 2014b) test sets are recorded indoors and test on similar looking scenes as the training set, the Human3D+ (Wenzheng Chen et al. 2016) test set was recorded with sensor suites that influence appearance and the annotations lack global alignment, and the MARCONI set (Elhayek et al. 2016) is markerless through manual annotation, but is limited in terms of motion, showing mostly walking motions.

This motivates a new test set to accompany MPI-INF-3DHP training set which is visually different from the training set, to promote and test for generalizability to scene settings and clothing.

Similar to the training set, the test set comes with ground truth annotations captured with a multi-view markerless motion capture system. It complements existing test sets with more diverse motions (Standing/Walking, Sitting/Reclining, Exercise, Sports (Dynamic Poses), On The Floor, Dancing/Miscellaneous), camera view-point variation, larger clothing variation (including a dress), and outdoor recordings from Robertini et al. 2016 in unconstrained environments. See Figure 3.8 for a representative sample, and Table 3.2 for details. We use the 'universal' skeleton for evaluation. As

Figure 3.8: Representative frames from MPI-INF-3DHP test set, showing the subjects, scene settings, and activity classes covered. The test set is diverse in clothing, includes both indoor and outdoor settings, and 4 of the 6 sequences are visually markedly different from the training set.

described earlier, the 'universal' skeleton is a uniformly scaled version of the annotated skeleton, such that the knee-to-neck height (sum of the lengths of the thigh and spine) is 930mm. The test set contains 6 sequences, 2 of which are captured in the same green-screen setup as the training set, and 2 sequences are captured in the same studio with the same camera setup as the training set, but without a green-screen background. These use 2048×2048 px footage captured at 25 FPS. The 2 outdoor sequences from Robertini et al. use 1920×1080 px GoPro footage at 60 FPS.

Evaluation Metric: Alternative to Mean Per Joint Position Error In addition to the commonly used Mean Per Joint Position Error (MPJPE) widely employed in 3D pose estimation, in agreement with the discussion in Ionescu et al. 2014b, this thesis suggests an extension to 3D of the 'Percentage of Correct Keypoints (PCK)' (Tompson et al. 2014; Toshev et al. 2014) metric used for 2D Pose evaluation, as well as the 'Area Under the Curve (AUC)' (Insafutdinov et al. 2016) computed for a range of PCK thresholds. MPJPE expresses error as the mean distance of each joint's 3D position prediction from the annotated 3D position of the joint. This has the disadvantage that the case of a few large mispredictions is difficult to tell apart from the case of mostly small mispredictions. The proposed metrics are intended to be more expressive and robust than MPJPE, revealing individual joint mispredictions more strongly while preventing large mispredictions in a few joints from overwhelming the possibly correct predictions for the rest of the joints. 3DPCK defines a threshold of 150mm, corresponding to roughly half of head size, similar to that chosen for the MPII 2D Pose dataset (Mykhaylo Andriluka et al. 2014). If the predicted 3D joint position lies within this threshold, the joint is indicated as correctly predicted. This metric is however a discrete metric, and meant to distinguish gross mispredictions from small mispredictions. An additional finer scaled metric to gauge the accuracy of the mostly correct predictions is the Area Under the Curve (AUC), which is the average percentage of correct keypoints obtained by varying the threshold from 0 to 150mm in 5mm increments. The common minimum set consisting of 14 joints across 2D and 3D

Table 3.3: The Multi-person 3D Pose Test Set (MuPoTS-3D) is comprised of 20 sequences with a diverse range of activities, in diverse scene settings.

	# Annotated Subjects	# Frames	Scene Setting	Image Resolution
TS1	2	201	Studio No GS	2048x2048
TS2	2	251	Studio No GS	2048x2048
TS3	2	401	Studio No GS	2048x2048
TS4	2	301	Studio No GS	2048x2048
TS5	2	261	Studio No GS	2048x2048
TS6	2	541	Outdoors	1920x1080
TS7	3	431	Outdoors	1920x1080
TS8	2	551	Outdoors	1920x1080
TS9	2	501	Outdoors	1920x1080
TS10	2	251	Outdoors	1920x1080
TS11	2	701	Outdoors	1920x1080
TS12	2	251	Outdoors	1920x1080
TS13	3	341	Outdoors	1920x1080
TS14	3	626	Outdoors	1920x1080
TS15	3	801	Outdoors	1920x1080
TS16	3	501	Outdoors	1920x1080
TS17	3	401	Outdoors	1920x1080
TS18	3	126	Outdoors	1920x1080
TS19	3	431	Outdoors	1920x1080
TS20	3	501	Outdoors	1920x1080

approaches (Head, Neck, Shoulders, Elbows, Wrists, Hips, Knees, Ankles) is chosen, to ensure compatibility of evaluations for various approaches as well as across datasets. The joints are grouped by symmetry (ankles, wrists, shoulders etc), and evaluations are broken down by joint-types, activity types, and sequences. These avenues of fine-grained evaluation are a key differentiator from existing benchmarks such as Human3.6m, and results with these metrics will be shown in Chapters 4-8.

3.3.2 MuPoTS-3D: Diverse Multi-Person 3D Pose Test Set

As discussed before, multi-person motion capture is challenging even for multi-view motion capture systems, and often requires extensive manual intervention and correction, limiting the extent of data that can be captured. Various real (not composited) multi-view scenarios were recorded, and multi-view marker-less motion capture system used, with painstaking manual correction, to create the 20 sequences of the first, expressive, in-the-wild multi-person 3D pose benchmark. The test set is termed **Mu**ltiperson **Po**se **T**est **S**et in **3D** (*MuPoTS-3D*). In addition to the pose annotations, per joint occlusion annotations are available. These were manually annotated by two annotators, and then verified and consolidated by a third annotator. The test set covers 5 indoor and 15 outdoor settings, with trees, office buildings, road, people, vehicles, and other stationary and moving distractors in the background. Some of the outdoor footage also has challenging elements like drastic illumination changes, and lens flare. The indoor sequences use 2048×2048 px footage at 30 FPS, and outdoor sequences use 1920×1080 px GoPro footage at 60 FPS. The test set consists of $>$8000 frames, split among the 20 sequences, with 8 subjects, in a variety of clothing styles, poses, interactions, and activities. See Table 3.3 for a detailed sequencewise breakdown. Notably, the test sequences do not

Figure 3.9: Examples from the proposed MuPoTS-3D evaluation set. Ground-truth 3D pose reference and joint occlusion annotations are available for up to 3 subjects in the scene (shown here for the frame on the top right). The set covers a variety of scene settings, activities and clothing.

resemble the training data, and include real interaction scenarios.

Evaluation Metric: The robust *3DPCK* evaluation metric, as well as *AUC* proposed for MPI-INF-3DHP are also used for the multi-person case, with the same set of 14 joints. The *3DPCK* numbers are reported per sequence, averaged over the subjects for which GT reference is available, along with a joint-wise breakdown. Further fine-grained evaluation is made possible by performance breakdown for occluded and unoccluded joints. The relative robustness of 3DPCK over MPJPE (Ionescu et al. 2014b) is also useful to offset the effect of jitter that arises in all non-synthetic annotations. Any GT annotated subject that does not find a match to a prediction reports all 14 joints being incorrect. Performance is also reported as averaged only over predictions matched to an annotated subject. This allows one to understand whether performance changes are due to an improvement in person detection accuracy or pose estimation accuracy.

3.4 Conclusion

This chapter introduces training datasets for single-person and multi-person scenarios, with ground truth 3D pose captured using a markerless motion-capture system, which additionally provides a large diversity in viewpoints not typically seen in prior datasets. Unlike marker based systems, markerless motion capture requires much less pre-capture setup time, and is not restricted to body-tight clothing. This allows motion capture with a large number of subjects, wearing regular street wear. For the single-person dataset, MPI-INF-3DHP, the subjects are guided through the capture sessions with prompts focused on capturing a diverse range of motion within the realm of the broad activity label. In addition to standing / walking type activities, the dataset covers sitting activities extensively, as well activities which require crouching or reaching out. The sitting activities involve extensive interaction with a chair, which also acts as an occluder from several viewpoints. Further, the dataset covers activities lying down on the floor, as well as sports and exercise activities. Additionally, one activity set is labeled as miscellaneous, to cover activities and pose types such as dancing or others which the subjects themselves may come up with. The annotations on the training and test sets are made compatible with the ‘universal’ annotations of Human3.6m, such that the datasets can be combined for training, and cross evaluations are made possible. Further, the capture setup is designed to provide an increased scope of appearance augmentation, which is crucial for generalizing learning-based approaches trained on these datasets to in-the-wild settings, as would be demonstrated in the subsequent Chapters.

Since the camera setup used for MPI-INF-3DHP is unchanged between subjects, and a green-screen background is employed, plausible multi-person scenes can be generated through image space compositing, with occlusion determined by the known depth of each subject relative to the camera. Even though the synthesized composites may not simulate all the nuances of human-human interaction fully, the approach allows generation of plausible multi-person scenarios at scale, and further has the full scope of appearance augmentation from MPI-INF-3DHP available. Methods trained on this dataset, termed MuCo-3DHP, generalize well to real world scenes, as would be demonstrated in subsequent Chapters. This dataset is a key contributor to making monocular RGB based multi-person pose estimation possible because the capture of multi-person scenarios at even a fraction of the scale possible with composites is infeasible. This is particularly so because even multi-view motion capture systems fail often under significant occlusion and the tracked pose requires extensive manual cleanup.

In order to evaluate the performance of learning based approaches trained on the proposed datasets,

this chapter proposes several test datasets comprised of *real world scenes*, captured in both indoor and outdoor settings. The scenarios in the test benchmarks are visually distinct from the training datasets. This promotes approaches that generalize well to in-the-wild scenes. For evaluating the performance on these benchmarks, this chapter proposes outlier robust metrics in place of the commonly used mean per joint error metric (MPJPE). The metric is a 3D extension of the Percentage of Correct Keypoints (PCK) commonly used for 2D body keypoint evaluation. It is coupled with the Area Under the Curve (AUC) obtained by sweeping the 3D PCK threshold, which provides a continuous measure of accuracy for the predictions considered as inliers. The datasets allow a breakdown of performance by body joint type, and by activity. The multi-person evaluation set allows further fine-grained analysis of the performance on visible and occluded joints. These benchmarks, with their extensive avenues for fine-grained evaluation, will be used to analyze the various approaches proposed in the subsequent Chapters.

Chapter 4

Towards In-The-Wild 3D Pose Estimation

The data capture process in the previous chapter (MPI-INF-3DHP) yields 3D pose annotated images with diverse pose classes, allows clothing and background appearance augmentation, and plausible multi-person scenes can be generated at a large scale through image compositing (MuCo-3DHP). This chapter takes the first steps to put the captured data to work, and examines the efficacy and limitations of the image-space augmentation avenues afforded by the captured dataset towards having learning based approaches generalize to scene settings beyond the capture setup.

The primary objective of this chapter is to establish the neural network training approach that will be used throughout the thesis, leveraging various datasets to learn representations that generalize to in-the-wild settings. This chapter studies the relatively constrained setting of unoccluded single-person pose estimation, and is extended in later Chapters (Ch. 6) to general scene settings comprised of multiple people, and containing inter-personal occlusions and occlusions by objects.

In This Chapter

- Demonstration of the poor scene appearance generalizability of methods trained on 3D pose datasets that are limited in appearance diversity (Section 4.1)
- Examination of the improved generalizability resulting from augmented MPI-INF-3DHP data, and a discussion of the domain gap between augmented image data and true in-the-wild images (Section 4.1)
- Presentation of a new method for transfer learning of neural network representations learned on in-the-wild 2D pose annotated datasets for further improving accuracy and scene appearance generalizability of 3D pose estimators, that will also serve as the basis for the training schema adopted in the rest of the thesis (Section 4.2)

The content of this chapter is based on D. Mehta et al. 2017a.

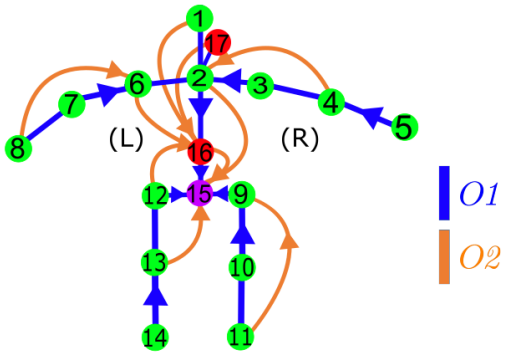

Figure 4.1: 3D pose, represented as a vector of 3D joint positions, is expressed variously as 1) $\mathbf{P^{3D}}$: relative to the root (joint #15), 2) $\mathbf{O}1^{\mathbf{3D}}$ (blue): relative to first order and, 3) $\mathbf{O}2^{\mathbf{3D}}$ (orange): relative to second order parents in the kinematic skeleton hierarchy.

4.1 Image-Crop-to-3D-Pose-Vector Regression: Task Setup

As discussed in Chapter 1, the overall task is the estimation of the 3D articulation θ_i of the kinematic structure of each subject and its camera relative location $\mathbf{d}_i$ in the scene, given a monocular RGB image I of size $W \times H$ as input. The solutions developed in this thesis split the task into two or more distinct parts. The first is the detection of body keypoint locations $\mathcal{P}^{2\mathcal{D}} = \{\mathbf{P_i^{2D}}\}_{i=1}^{m}$, where $\mathbf{P^{3D}}_i \in \mathbb{R}^{2\times J}$, of the m persons in the input image with a Convolutional Neural Network based heatmap prediction approach, as described in Chapter 2. Either jointly with the 2D keypoint detection network or separately, a Convolutional Neural Network predicts the 3D articulation (pose) $\mathcal{P}^{3\mathcal{D}} = \{\mathbf{P_i^{3D}}\}_{i=1}^{m}$ for each of the m persons in the image, where, $\mathbf{P^{3D}}_i \in \mathbb{R}^{3\times J}$ describes the 3D locations of the n body joints of person i. A subsequent kinematic model fitting stage reconciles the 2D and 3D predictions per subject and across time, to localize each subject relative to the camera, and yield the articulated pose expressed as joint angles. This chapter focuses on the prediction of 3D pose $\mathbf{P^{3D}}$ in single-person scenarios.

For single-person scenarios, the input image can be considered to be a tight crop around the subject, obtained either through an object bounding-box detection approach (W. Liu et al. 2016), or by creating a bounding box around the detected keypoints $\mathbf{P^{2D}}$. The task being examined in this chapter, thus, is the prediction of the 3D articulation $\mathbf{P^{3D}}$ of the subject in the cropped image I^C. Such an approach is applicable to multi-person scenarios as (Gkioxari et al. 2014; Iqbal et al. 2016; Moon et al. 2019; Papandreou et al. 2017; Pishchulin et al. 2012; Sun et al. 2011), with an additional mechanism for handling object and inter-personal occlusions required. However, the main disadvantage of applying such an approach to multi-person scenarios is that the computational cost scales linearly with the number of subjects in the scene, which becomes a significant impediment to achieving real-time performance even in scenes with a moderate number of people. Hence, the thesis proposes alternative formulations for the multi-person case in Chapter 6.

The input image is zero centered by subtracting 0.5 from all channels, and the image crops around the subject are resized to 368×368 px. using zero padding to prevent distortion of the crop aspect ratio. The target pose $\mathbf{P^{3D}}$ is expressed as a 1D vector of size $3 * J$.

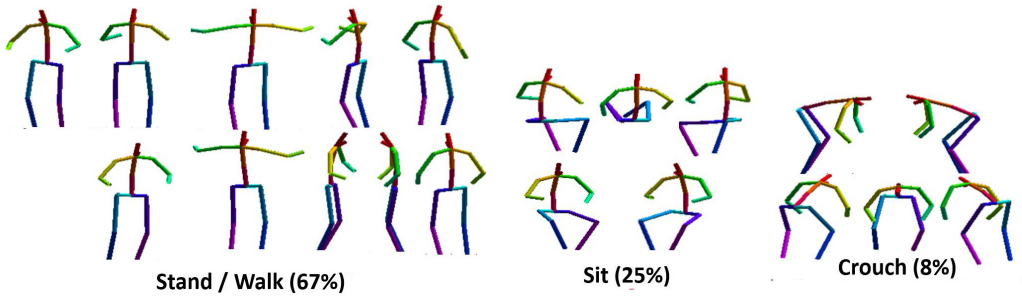

Figure 4.2: Representative poses (centroids) of the 20 K-means pose clusters of the Human3.6m test set (subjects S9,S11), visually grouped into three broad pose classes, which are used also to perform per-class evaluation. Upright poses are dominant, with complex poses such as sitting and crouching only accounting for 25% and 8% of the poses respectively. The proposed Multi-modal fusion scheme significantly improves the latter two, yielding a 3.5mm improvement for Sit and 5.5mm for Crouch pose classes.

4.1.1 Training Data

In the various experiments in this chapter, the following datasets are considered. For each dataset ≈37.5k frames are selected, yielding ≈75k samples per dataset after scale augmentation at 2 scales (0.7 and 1.0).

Human3.6m: The H80k (Ionescu et al. 2014a) subset of Human3.6m is used, with the 'universal' skeleton 3D pose anotations. Subjects S1, S5, S6, S7 and S8 are used for training.

Appearance Unaugmented MPI-INF-3DHP: For the MPI-INF-3DHP dataset proposed in Chapter 3, in order to maintain compatibility of viewpoints with Human3.6m and other datasets, only the 5 chest high camera views are selected for all 8 subjects, giving 500k frames. Frames are subsampled such that at least one joint has moved by more than 200mm between selected frames. From the resulting 100k frames, 37.5k frames are randomly sampled, resulting in 75k frames after scale augmentation. No background and clothing augmentation is applied.

Appearance Augmented MPI-INF-3DHP: The augmented version has the same 37.5k frames, $\approx 10k$ of which are unaugmented, $\approx 15k$ have only the background and chair appearance augmented, and the rest have clothing augmented as well. Scale augmentation results in 75k frames.

4.1.2 Method

For each of the above listed datasets, a Convolutional Neural Network (CNN) is trained to predict the 3D pose. The network will be referred to as 3DPoseNet. Motivated by work on Multi-Task Learning (Caruana 1997), in addition to the final 3D pose vector the 2D keypoint locations $\mathbf{P}^{\mathbf{2D}}$ are also predicted in the form of heatmaps H.

Base Network Architecture: A CNN structure based on Resnet-101 (K. He et al. 2016a) is used as the backbone (up to the filter banks before `res5a`). The number of features in the `res5a` module are halved, and the remaining layers removed from the network. A 3D prediction stub $\mathcal{S}$ comprised of a convolution layer $(k_{5\times5}, s_2)$ with 128 features and a final fully-connected layer that outputs the 3D joint locations is added on top. 2D heatmaps H are predicted as an auxiliary task after `res5a` and, use intermediate supervision with pose $\mathbf{P}^{\mathbf{3D}}$at `res3b3` and `res4b22`. Features learned with Resnet-101 from ImageNet are used to initialize the shared layers of the backbone network. The other layers are initialized using Xavier initialization (Glorot et al. 2010).

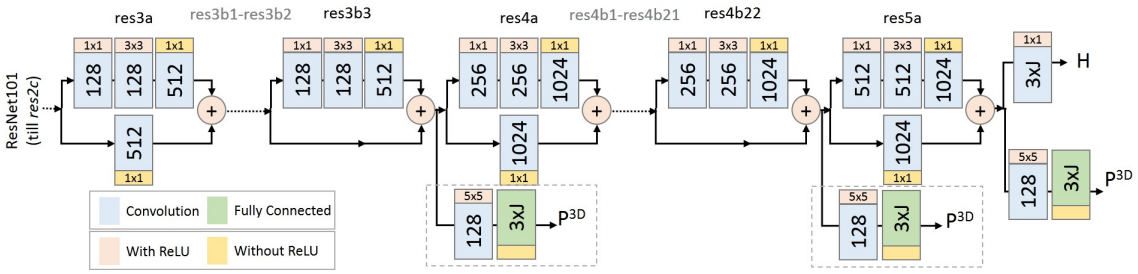

Figure 4.3: The network architecture is based on ResNet-101 (K. He et al. 2016a), as described in Section 4.1.2. The network outputs 2D pose $\mathbf{P^{2D}}$ as heatmaps $\mathbf{H}_j$, and 3D pose $\mathbf{P^{3D}}$ as a $3 * J$ dimensional vector, and uses intermediate supervision, as indicated by dotted boxes.

Multi-modal Pose Fusion: Formulating joint location prediction relative to a single reference joint is not always optimal. Existing literature (S. Li et al. 2014) has observed that predicting joint locations relative to their direct kinematic parents (Order 1 parents) improves performance. However, own experiments reveal that this is not true, and instead it is observed that depending on the pose and the visibility of the joints in the input image, the optimal relative joint for each joint's location prediction differs.

Hence, a multi-modal prediction strategy is proposed which uses joint locations $\mathbf{P^{3D}}$relative to the root, $\mathbf{O}\mathit{1}^{\mathbf{3D}} \in \mathbb{R}^{3*J}$ relative to Order 1 parents and $\mathbf{O}\mathit{2}^{\mathbf{3D}} \in \mathbb{R}^{3*J}$ relative to Order 2 parents along the kinematic tree as the *three modes* of prediction, see Figure 4.1. In place of a single 3D prediction stub used in the base architecture, three identical 3D prediction stubs are attached to `res5a` for predicting the pose as $\mathbf{P^{3D}}$, $\mathbf{O}\mathit{1}^{\mathbf{3D}}$and $\mathbf{O}\mathit{2}^{\mathbf{3D}}$. These predictions are at a later fine-tuning step fed into three fully-connected layers, with 2k, 1k and 51 nodes respectively, to implicitly determine and fuse the better constraints per joint into the final prediction. The network has the flexibility to emphasize different combinations of constraints depending on the pose. This can be viewed as intermediate supervision with auxiliary tasks, yet the separate streams for predicting each mode individually are key to its efficacy.

4.1.3 Baseline Results

Efficacy of Multi-modal Prediction Fusion: The multi-modal fusion scheme yields noticeable improvement across all activity sets for Human3.6m dataset, as shown in Table 4.1. Since upright poses dominate in pose datasets, and the activity classes are often diluted significantly by upright poses, the true extent of improvement by the multi-modal fusion scheme is masked. To show that the fusion scheme indeed improves challenging pose classes, frames from Human3.6m test set are clustered by pose as shown in Figure 4.2, which visualizes the centroid of each cluster. The clusters are then visually grouped into three pose classes, namely Stand/Walk, Sit and Crouch, going by the cluster representatives. For the Stand/Walk class, adding fusion has minimal effect, going from 88.4mm to 88.8mm. However, for Sit class fusion leads to a 3.5mm improvement, from 118.9mm to 115.4mm. Similarly, Crouch class has the highest improvement of 5.5mm, going from 156mm to 150.5mm.

Poor Generalization Performance of Human3.6m and Unaugmented MPI-INF-3DHP: The MPI-INF-3DHP single-person test set proposed in Chapter 3 complements existing benchmarks with additional pose variation and appearance variation, in different settings. This makes it suitable for testing the generalization of various methods. The sequencewise results on MPI-INF-3DHP are

Table 4.1: Activity-wise results (MPJPE in mm) on Human3.6m Ionescu et al. 2014b. Adding the proposed model components one-by-one on top of the *Base* network shows successive improvement of the total accuracy. Models are trained on Human3.6m, with network weights initialized from ImageNet, unless specified otherwise. The version marked with MPI-INF-3DHP is trained with Human3.6m and MPI-INF-3DHP. Evaluation with all 17 joints, on every 64^{th} frame, without rescaling to a person specific skeleton.

	Direct	**Discuss**	**Eating**	**Greet**	**Phone**	**Posing**	**Purch.**	**Sitting**	**Sit Down**	**Smoke**	**Take Photo**	**Wait**	**Walk**	**Walk Dog**	**Walk Pair**	**Total**
Base	99.0	100.1	86.1	101.8	101.3	96.7	94.9	125.3	158.3	100.2	112.5	99.6	83.4	109.6	95.8	104.3
+ Multi-modal Fusion	98.2	99.1	84.8	100.6	99.3	95.3	92.4	122.5	151.6	98.1	110.8	98.6	81.4	107.7	93.9	102.3
+ Transfer *2DPoseNet*	58.6	70.1	62.6	69.7	**77.5**	57.0	76.8	**97.5**	121.9	70.2	86.2	68.5	**53.9**	84.2	**60.1**	74.5
+ MPI-INF-3DHP	**57.5**	**68.9**	**59.6**	**67.3**	78.1	**56.9**	**69.1**	100.0	**117.5**	**69.4**	**82.4**	**68.0**	55.2	**76.5**	61.4	**72.9**

Table 4.2: Evaluation by scene-setting of the proposed method (Base+Fusion) on MPI-INF-3DHP test set, trained with different 3D pose datasets. 3D Percentage of Correct Keypoints metric is reported here, with a threshold of 150mm. The sequences TS1 and TS2 use a green-screen background, while sequences TS3-6 do not, and differ in appearance from the training set. Training on the MPI-INF-3DHP training set improves accuracy significantly, in particular with the augmentation strategy described in Chapter 3.

3D Dataset	3DPCK						
	TS1	TS2	TS3	TS4	TS5	TS6	All
Human3.6m	24.0	16.4	33.6	28.4	10.2	13.0	22.3
MPI-3DHP Unaug	86.0	56.9	39.4	49.6	23.6	17.9	50.0
MPI-3DHP Aug.	84.3	65.3	61.5	63.8	42.3	51.7	64.3

shown in Table 4.2. TS1 and TS2 are with a green-screen background, similar to MPI-INF-3DHP training set, while the remaining 4 sequences TS3-6 are visually different from the training set. It is clear that the avenues of image augmentation afforded by MPI-INF-3DHP significantly improve the the generalization performance of learning based approaches.

4.2 Transfer Learning To Further Improve Scene Generalization

While augmented MPI-INF-3DHP significantly improves generalizability to in-the-wild scenes over prior datasets such as Human3.6m, the image space augmentation approach used for increasing appearance diversity has a domain gap to real world images. This prompts the question whether closing this domain gap would further improve the scene generalization of methods learned on this dataset.

Transfer learning is already utilized in the proposed pipeline, as described in Section 4.1.2, where the features learned with Resnet-101 from ImageNet (Russakovsky et al. 2015) are used to initialize the 3D pose network, as is common for many vision tasks. While this affords a faster convergence while training, there remains room for improved generalization beyond the gains from potential supervision and dataset contributions. Due to the similarity of the tasks, features learned for 2D pose estimation on in-the-wild MPII and LSP training sets can be transferred to 3D pose estimation. A 2D pose estimation network (2DPoseNet) is trained with the same backbone network as 3DPoseNet. See Appendix B for details of 2DPoseNet. Different ways of transferring these representations to the 3D pose network (3DPoseNet) are explored.

Table 4.3: Evaluation of the mechanisms of transfer learning from 2D Pose Network (2DPoseNet) to 3D Pose Network (3DPoseNet) that were explored in the context of the *Base* network. The table compares the effect of various learning rate multiplier combinations for different parts of the network. Human3.6m, Subjects 1,5,6,7,8 used for training, and every 64^{th} frame of 9,11 used for testing. * = Xavier initialization

Learning Rate Multiplier			
up to `res4b22`	`res5a`	3D Stub $\mathcal{S}$	**Total MPJPE (mm)**
1	1	1*	118.7
1/10	1/10	1*	84.6
1/1000	1/1000	1*	89.2
1/10	1	1*	90.7
1/1000	1	1*	**80.7**

4.2.1 Trading off Transferred Representations and New Feature Learning

A naïve initialization of the weights of 3DPoseNet is inadequate, and there is a tradeoff to be made between the preservation of transferred features and learning new pertinent features. Here, an approach is proposed to guide the tradeoff through a learning rate discrepancy between the transferred layers and the new layers. The mechanism for this transfer of features is determined through experimental validation. Table 4.3 shows the evaluated mechanisms for transfer from 2DPoseNet. Based on the experiments, it is chosen to scale down the learning rate of the layers till `res4b22` by a factor of 1000. Through similar experiments for the transfer of ImageNet features, it is chosen to scale down the learning rate of layers till `res4b22` by 10.

The same approach can be applied to other network architectures, and the above experiments on learning rate discrepancy serve as a sound starting point for the determination of the transfer learning mechanism. Unlike jointly training with annotated 2D and 3D pose datasets, this approach has the advantage of not requiring the 2D annotations to be consistent between the two datasets, and one can simply use off-the-shelf trained 2D pose networks.

4.2.2 Results With Effective Transfer Learning

Alternative approaches to improve scene generalization, which examine the problem from a domain adaptation perspective (Ganin et al. 2015), are also be considered. See Appendix B.3 for details. The proposed approach outperforms domain adaptation based approaches, see Table 4.4, first row. Additionally, Table 4.3 validates that the common fine-tuning of the fully-connected layers (third row) and fine-tuning of the complete network (first row) is much less effective then the proposed scheme.

The proposed approach of transferring representations from *2DPoseNet* to *3DPoseNet* yields 66.5% 3DPCK on MPI-INF-3DHP test-set when trained with only Human3.6m data, compared to 64.3% 3DPCK of the model trained on augmented MPI-INF-3DHP dataset without transfer learning. It also shows state of the art performance on Human3.6m test set with an error of ≈74mm, demonstrating the dual advantage of the approach in improving both the accuracy of pose estimation and generalizability to in-the-wild scenes. Combining the proposed dataset and transfer learning leads to the best results at ≈ 75.9% 3DPCK. See Table 4.4.

Table 4.4: Evaluation on MPI-INF-3DHP test set with weight transfer from *2DPoseNet*, by scene setting. 3DPCK is reported for all 6 sequences, for methods trained with different datasets. The sequences TS1 and TS2 use a green screen background, while sequences TS3-6 do not, and differ in appearance from the training set. Training with a combination of MPI-INF-3DHP and Human3.6m gives the best accuracy over all.

3D Dataset	TS1	TS2	TS3	TS4	TS5	TS6	All	
	3DPCK	3DPCK	3DPCK	3DPCK	3DPCK	3DPCK	3DPCK	AUC
Human3.6m (Domain Adaptation)	52.0	35.2	47.2	38.5	23.4	42.5	41.4	17.7
Human3.6m (Base+Fusion)	81.0	61.1	70.3	62.3	47.1	66.2	66.5	33.0
MPI-3DHP Aug. (Base+Fusion)	86.8	72.4	69.7	67.4	46.1	63.1	70.2	35.1
MPI-3DHP UnAug. (Base+Fusion)	88.9	72.6	70.9	69.1	46.2	60.9	70.9	36.1
H36m+MPI-3DHP (Base+Fusion)	89.6	75.7	76.6	73.0	48.7	77.8	75.9	40.6

In contrast to existing approaches countering data scarcity, transfer learning does not require complex dataset synthesis, yet exceeds the performance of Wenzheng Chen et al. 2016 (with synthetic data and domain adaptation, 28.8% 3DPCK, after procrustes alignment) and the method proposed in the previous section trained with the synthetic data of Grégory Rogez et al. 2016 (21.7% 3DPCK). Our approach also performs better than domain adaptation (Ganin et al. 2015) to in-the-wild data (Table 4.4). Specifics of the domain adaptation approach are in Appendix B.

Additionally, qualitative results are shown on LSP (Johnson et al. 2010) and the CMU Panoptic (Joo et al. 2015) datasets, demonstrating robustness to general scenes. Refer to Figure 4.4.

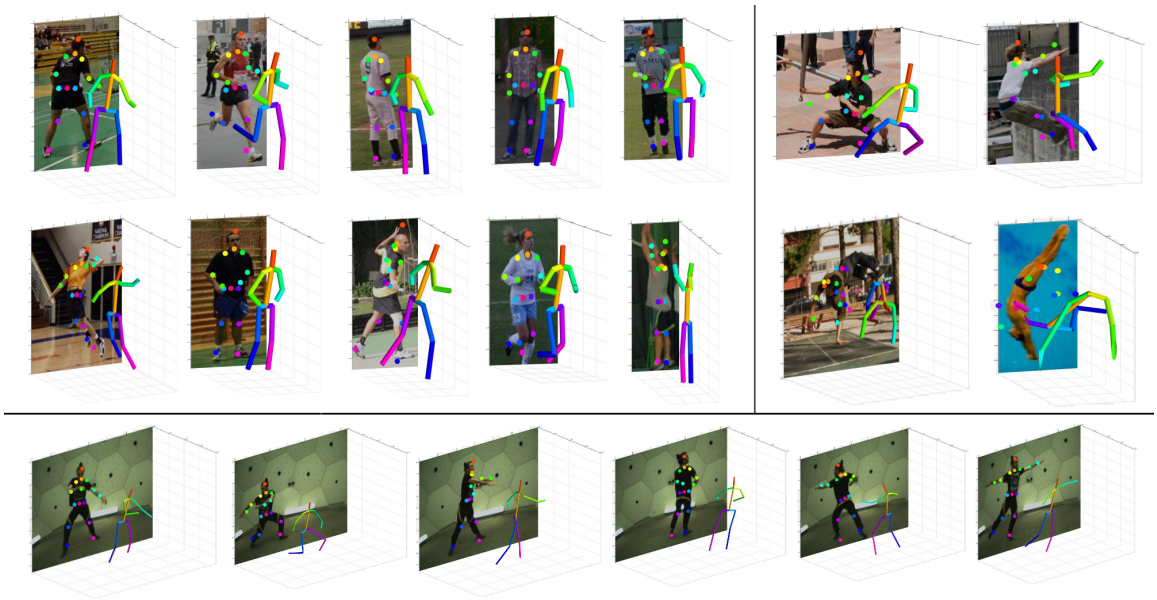

Figure 4.4: Qualitatively evaluation on every 100th frame of the LSP [2010] test set. The proposed approach succeeds in challenging cases (left), with only few failure cases (right). The *Dance1* sequence of the Panoptic Dataset (Joo et al. 2015), is also well reconstructed (bottom).

Comparisons on Human3.6m.: Table 4.5 shows comparison of the proposed approach with existing methods, all trained on Human3.6m. Altogether, with the multi-modal prediction scheme and transfer learning, the proposed method achieves the state of the art (74.11mm, without scaling) at the time, while also generalizing to in-the-wild scenes. Note that the Volumetric coarse-to-fine approach (Pavlakos et al. 2017) requires estimates of the bone lengths to convert their predictions

Table 4.5: Comparison of results on Human3.6m [2014] with the state of the art. Human3.6m, Subjects 1,5,6,7,8 used for training, and 9,11 used for testing. ^{S} = Scaled to test subject specific skeleton, computed from T-pose. ^{T}= Uses Temporal Information, $^{J14/J17}$ = Joint set evaluated, ^{A} = Uses Best Alignment To GT per frame, ^{Act} = Activitywise Training, $^{1/10/64}$ = Test Set Frame Sampling

Method	Total MPJPE (mm)
Deep Kinematic Pose[2016]$^{\mathbf{J17},B}$	107.26
Sparse. Deep. [2015]$^{T,\mathbf{J17},B,10,Act}$	113.01
Motion Comp. Seq. [2016]$^{T,\mathbf{J17},B}$	124.97
LinKDE [2014]$^{\mathbf{J17},B,Act}$	162.14
Du et al. [2016]$^{T,\mathbf{J17},B}$	126.47
Rogez et al. [2016]$^{(\mathbf{J13}),B,64}$	121.20
SMPLify [2016]$^{\mathbf{J14},B,A,(First\ cam.)}$	82.3
3D=2D+Matching [2017]$^{\mathbf{J17},B}$	114.18
Distance Matrix [2017]$^{\mathbf{J17},B}$	87.30
Volumetric Coarse-Fine[2017]$^{\mathbf{J17},B,S*}$	71.90
LCR-Net [2017]$^{\mathbf{J17},B}$	87.7
Full model (w/o MPI-INF-3DHP) $^{\mathbf{J17},B}$	74.11
Full model (w/o MPI-INF-3DHP) $^{\mathbf{J17},B,\mathbf{S}}$	68.61
Full model (w/o MPI-INF-3DHP) $^{\mathbf{J14},B,\mathbf{A}}$	**54.59**

from pixels to 3D space. Complementing Human3.6m with our augmented MPI-INF-3DHP dataset further reduces the error to 72mm.

4.3 Conclusion

This chapter demonstrates the improved generalizability to in-the-wild scenes made possible with the proposed MPI-INF-3DHP dataset, and also proposes a transfer learning based approach to further improve the generalizability to in-the-wild scenes by leveraging representations learned on 2D pose annotated datasets. Despite the demonstrated competitive results, the proposed method is just a first step, and has many limitations, which are discussed in Chapter 5. The formulation proposed in Chapter 5 addresses some of the shortcomings, and takes a step further towards developing an accurate and real-time 3D pose estimation system.

Chapter 5

Coupling 3D Pose Inference to Direct Image Evidence

As mentioned in Section 1.1.1, the motion capture systems developed in this thesis estimate the 3D articulation θ_i of the kinematic structure of each subject i in the scene, along with the location $\mathbf{d}_i \in \mathbb{R}^3$ of a reference joint (pelvis) relative to the camera. For each frame, per detected subject, the articulation of the kinematic structure expressed in terms of joint position, $\mathbf{P_i^{3D}} \in \mathbb{R}^{3\times J}$, is predicted using learning based methods, together with a localization of the body joints $\mathbf{P_i^{2D}} \in \mathbb{R}^{2\times J}$ in the image plane. These are subsequently passed to a kinematic model fitting step along with information about the skeleton scale. Kinematic model fitting incorporates information across frames to produce temporally smooth joint angle estimates, and localizes the skeletal structures relative to the camera.

The general training schema described in the previous chapter leads to 3D pose estimators ($\mathbf{P_i^{3D}}$) that generalize to in-the-wild (unoccluded, single person) scene appearance. Importantly, improved generalizability to in-the-wild scenes also improves the accuracy on various benchmarks, and for the first time brings the accuracy of the per-frame monocular pose estimation system to a level that can be considered good enough for a motion capture system to be built around.

However, closely examining the image-crop to 3D pose vector regression formulation reveals issues with the way the learning task is formulated, which would impact the accuracy and run-time when incorporated in the full system. Further, there is no clear path to scaling the direct pose vector regression approach to the multi-person case. This chapter proposes an alternative formulation for the single person case, with the key intuition to directly couple pose inference of each body part to its local image evidence. This improves the prediction accuracy and run-time performance when incorporated in the full system, and is extensible to multi-person scenarios.

In This Chapter

- Discussion of drawbacks of direct 3D pose vector regression from input image crops (Section 5.1)
- Introduction of a new fully-convolutional 3D pose formulation to tie pose estimation to image evidence (Section 5.3)

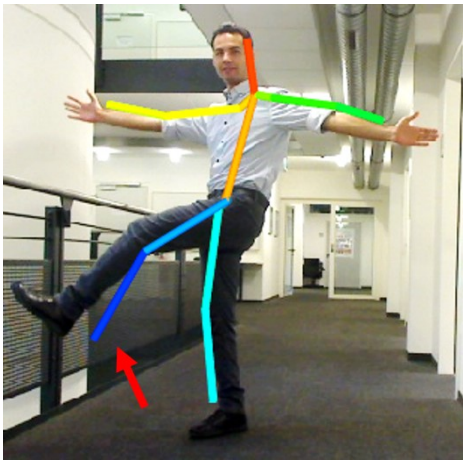

Figure 5.1: The 3D predictions resulting from direct image to 3D pose vector regression do not match the extent of articulation of the input image, and tend towards the average pose of the training corpus.

The content of this chapter is based on D. Mehta et al. 2017b.

5.1 Shortcomings of Image-Crop-to-Pose-Vector Regression

5.1.1 Alignment of 3D Pose Prediction With Input Image

The direct regression approach used in the previous chapter is a commonly employed 3D pose formulation. Given a cropped image $\mathbf{I}$ as input to the network, using the root-relative 3D pose $\mathbf{P}^{\mathbf{3D}}$ as a direct regression target leads to predicted poses whose extent of articulation doesn't match that of the person in the image (See Figures 5.1 and 5.5). Though small misalignments can be addressed by a subsequent inverse kinematics fit to 2D and 3D pose predictions, it takes more iterations (and time) the more the misalignment. It is instructive to examine possible causes of this mode of inaccuracy in the predictions.

One possible cause of the issue is that the pose regression network, which predicts root-relative scale-normalized 3D poses, is forced to learn shift invariance (See Figure 5.2). An alternative would be a formulation that lends itself to shift invariant inference, without the network's output being shift invariant. Any network capacity used up in learning shift invariance potentially comes at the cost of network capacity available towards faithfully capturing the extent of articulation depicted in the input image.

Another possible cause of this issue is that a direct pose regression approach does not ensure that the appropriate parts of the image are considered when inferring the pose of a particular body part. The root-relative 3D location of a particular body part can not only be inferred from direct image evidence of that body part, but often also from the context provided by the rest of the body. The degree of contribution of information from direct and indirect context to the final pose prediction is unclear in the case of direct regression. The formulation proposed in this chapter makes the contribution of direct and indirect image context more explicit by only relying on direct image evidence.

Figure 5.2: Since the 3D pose is expressed as body root relative joint locations, shifting the person around in the crop does not change the expected pose prediction. This implies that with a direct pose vector regression approach, the network's output is expected to be shift invariant.

5.1.2 Sensitivity to the Quality of Image Crop

The direct regression approach expects image crops around the subject to be input. The use of an image crop around the subject results in better pose accuracy (Cao et al. 2017), and for the single person case it is less computationally costly than feeding the complete image to the network. However it requires additional preprocessing of the input image to localize the subject and crop the subject. In addition to the extra preprocessing cost of computing the bounding-box, the quality of the crops at deployment would not match the ground truth crops used for training, and such a domain shift degrades performance significantly, as shown in Section 5.3.3.

5.2 Motivating A Fully Convolutional 3D Pose Formulation

An alternative to the direct regression approach is making the network training objective fully convolutional, which makes the output of the network shift equivariant. Each body joint is localized in the image plane, and its pixel location is used to read out the body joint's 3D pose from the fully convolutional output. This read-out (or voting) scheme results in shift invariant predictions without the network output needing to be shift invariant.

Also, as discussed in Chaper 2, for 2D pose estimation with convolutional neural networks, the change of formulation from direct regression of x, y body-joint coordinates (Toshev et al. 2014) to a heatmap based body-joint detection formulation (Tompson et al. 2014) has been the key driver behind significant improvements in 2D pose estimation. This serves as additional motivation for developing a fully-convolutional formulation. Concurrent with the approach presented in this Chapter, Pavlakos et al. 2017 also develop a fully convolutional 3D pose formulation, but it is limited by an expensive volumetric output predicted per joint, which makes it untenable for larger image sizes. Further, as would be discussed in Chapter 6, such a volumetric formulation is not easily extensible to the multi-person case. Some approaches bypass the problems with direct image to pose vector regression by instead predicting 3D pose from 2D keypoints (Martinez et al. 2017; Xiaowei Zhou et al. 2015a) extracted using a fully convolutional 2D heatmap formulation. However 2D to 3D lifting is less accurate due to depth ambiguities, which image cues can help disambiguate. Others (Xingyi Zhou et al. 2017) jointly predict 2D keypoints as heatmaps and regress the depth dimension as a vectorized output.

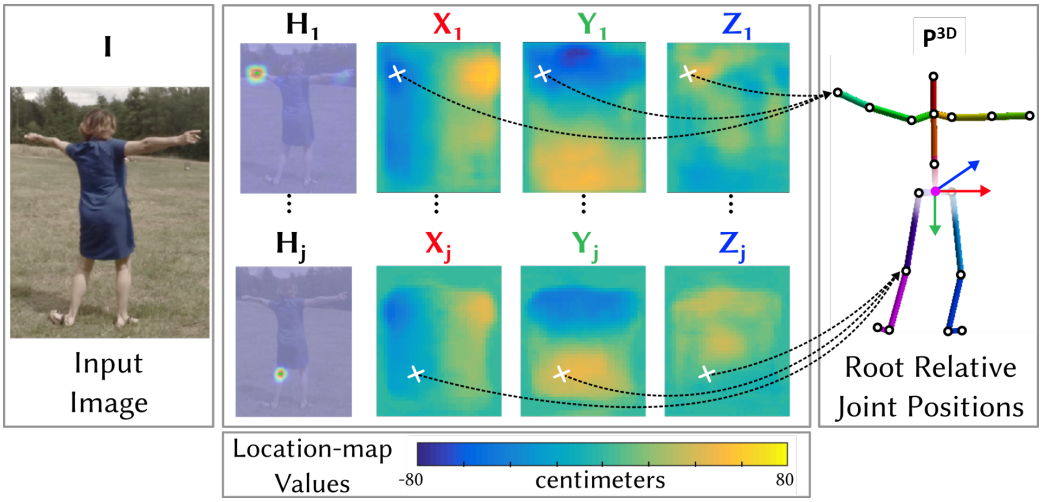

Figure 5.3: Location-Maps: Schema of the fully-convolutional formulation for predicting root-relative joint locations. For each joint j, the 3D coordinates are predicted from their respective *location-maps* $\mathbf{X}_j, \mathbf{Y}_j, \mathbf{Z}_j$ at the position of the maximum in the corresponding 2D heatmap $\mathbf{H}_j$. The structure observed here in the location-maps emerges due to the spatial loss formulation.

5.3 The Location-Map Formulation

The heatmap based formulation for 2D pose estimation naturally ties image evidence to pose estimation by predicting a confidence heatmap $\mathbf{H}_j$ over the image plane for each joint $j \in \{1..J\}$.

The 2D heatmap formulation can be extended to 3D using three additional *location-maps* $\mathbf{X}_j, \mathbf{Y}_j, \mathbf{Z}_j$ per joint j, capturing the root-relative locations x_j, y_j and z_j respectively. To have the 3D pose prediction linked more strongly to the 2D appearance in the image, the x_j, y_j and z_j values are read off from their respective location-maps at the position of the maximum of the corresponding joint's 2D heatmap $\mathbf{H}_j$, and stored in $\mathbf{P}^{\mathbf{3D}} = \{\mathbf{x}, \mathbf{y}, \mathbf{z}\}$, where $\mathbf{x} \in \mathbb{R}^{1\times J}$ is a vector that stores the coordinate x location of each joint maximum.

There are several ways of interpreting the formulation. The design can be thought of as making 3D pose inference for a particular joint by having the network's receptive field centered on that joint. Alternatively, the maps $\mathbf{X}_j, \mathbf{Y}_j, \mathbf{Z}_j$ can be seen as encoding possible candidates for joint j's pose, conditioned on the pixel location, and the image evidence for joint j votes for the candidate at the the pixel location corresponding to the maximum of heatmap $\mathbf{H}_j$. The pose formulation is visualized in Figure 5.3.

Bounding-box Tracker: Networks using this fully-convolutional formulation are not constrained in input image size, and can work without tight crops. However, for efficient inference, the 2D joint location predictions can be used to bootstrap a no-cost bounding-box tracker for cropping a region around the subject in the subsequent frame. The crop is resized to a pre-determined pixel resolution (typically 256×256 px, or 368×368 px) before being passed to the network. The predicted 2D joint locations in the crop are then converted to original image coordinates and an enclosing bounding-box obtained. A buffer region of +/- 20% of each dimension is added around the bounding box and used as the bounding-box for the next frame.

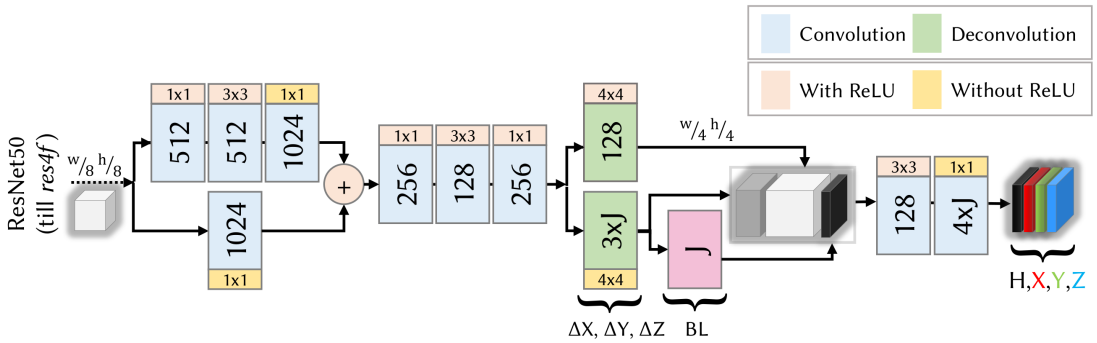

Figure 5.4: Network Structure. The structure above is preceded by ResNet50/100 till level 4. We use kinematic parent relative 3D joint location predictions $\Delta\mathbf{X}$, $\Delta\mathbf{Y}$, $\Delta\mathbf{Z}$ as well as bone length maps $\mathbf{BL}$ constructed from these as auxiliary tasks.The network predicts 2D location heatmaps $\mathbf{H}$ and root-relative 3D joint locations $\mathbf{X}$, $\mathbf{Y}$, $\mathbf{Z}$.

5.3.1 Training Objective For Location-Maps

To enforce the fact that only x_j, y_j and z_j from their respective maps at joint j's 2D location are of interest, the joint's location-map loss is weighted stronger around the joint's 2D location, and no loss is enforced far away from the joint's 2D location. Using L2 loss, for x_j the loss formulation is:

$$\text{Loss}(x_j) = \|\mathbf{H}_j^{\text{GT}} \odot (\mathbf{X}_j - \mathbf{X}_j^{\text{GT}})\|_2, \tag{5.1}$$

where GT indicates ground truth and $\odot$ is the Hadamard product. The location maps are weighted with the respective ground truth 2D heatmap $\mathbf{H}_j^{\text{GT}}$, which in turn have confidence equal to a Gaussian with a small support localized at joint j's 2D location. Note that no structure is imposed on the location-maps. The structure that emerges in the predicted location-maps is indicative of the correlation of x_j and y_j with root-relative location of joint j in the image plane. See Figure 5.3.

5.3.2 Experimental Details

Network Architecture: The ResNet-50 network architecture of K. He et al. 2016a is adapted for use with the proposed formulation by replacing the layers of ResNet-50 from *res5a* onwards with the architecture depicted in Figure 5.4, producing the heatmaps and location-maps for all joints $j \in \{1..J\}$. After training, the Batch Normalization [2015] layers are merged with the weights of their preceding convolution layers to improve the speed of the forward pass. Other network architectures can be employed in place of ResNet-50, such as the deeper and more expensive ResNet-100, or the efficient *SelecSLS* network architecture proposed in Chapter 7.

Intermediate Supervision: 2D heatmaps and 3D location-maps are also predicted from the features at *res4d* and *res5a*. During training, we taper down the weights of intermediate losses with increasing iteration count. Additionally, similar to the root-relative location-maps $\mathbf{X}_j$, $\mathbf{Y}_j$ and $\mathbf{Z}_j$, kinematic parent-relative location-maps $\Delta\mathbf{X}_j$, $\Delta\mathbf{Y}_j$ and $\Delta\mathbf{Z}_j$ are predicted from the features at *res5b*, and additional feature maps, termed as bone length-maps, are calculated as:

$$\mathbf{BL}_j = \sqrt{\Delta\mathbf{X}_j \odot \Delta\mathbf{X}_j + \Delta\mathbf{Y}_j \odot \Delta\mathbf{Y}_j + \Delta\mathbf{Z}_j \odot \Delta\mathbf{Z}_j}. \tag{5.2}$$

These intermediate predictions are subsequently concatenated with the intermediate features, to give the network an explicit notion of bone lengths to guide the predictions. See Figure 5.4.

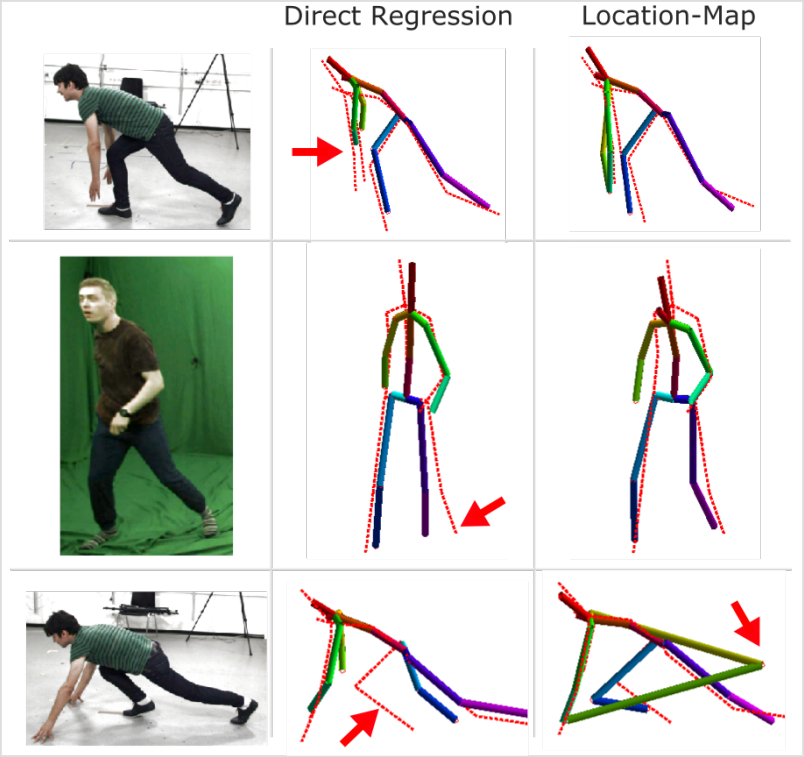

Figure 5.5: A visual look at the direct 3D predictions resulting from the fully-convolutional formulation vs direct pose vector regression. The location-map formulation allows the predictions to be more strongly tied to image evidence, leading to overall better pose quality, particular for the end effectors. The red arrows point to mispredictions. The location-map formulation produces occasional large mispredictions when the underlying 2D joint is misdetected, but these errors can be easily remedied in case of video input through filtering.

Training: The network is pretrained for 2D pose estimation on MPII (Mykhaylo Andriluka et al. 2014) and LSP (Johnson et al. 2010, 2011) to allow superior in-the-wild performance, as discussed in the previous chapter. For 3D pose, both MPI-INF-3DHP (D. Mehta et al. 2017a) and Human3.6m (Ionescu et al. 2014b) datasets are used, as per the details in Chapter 3. The network is trained with person centered crops, and image scale augmentation is used at 2 scales ($0.7\times, 1.0\times$), resulting in 75k training samples for Human3.6m and 100k training samples for MPI-INF-3DHP datasets. Figure 3.5 shows a few representative frames of training data. In addition to the 17 joints typically considered, hands and foot tip positions are also considered. The ground truth joint positions are with respect to a height normalized skeleton (knee–neck height 92 cm). A variant of the direct regression approach discussed in the previous chapter is trained on identical data with these 21 joints.

The Caffe [2014] framework is used for training with the Adadelta (Zeiler 2012) solver, with step learning rate tapering.

5.3.3 Results

The accuracy improvements over direct 3D pose vector regression (Chapter 4) are validated on Human3.6m dataset [2014] and the MPI-INF-3DHP [2017] dataset proposed in the thesis.

Quantitative Evaluation: Results of the proposed Location-Map formulation are compared primar-

Table 5.1: Comparison of location-map formulation against state of the art on MPI-INF-3DHP test set, using ground-truth bounding-boxes. The table reports the Percentage of Correct Keypoints measure in 3D, and the Area Under the Curve for the same, as proposed by MPI-INF-3DHP [2017], as well as the Mean Per Joint Position Error in mm. Higher PCK and AUC is better, and lower MPJPE is better.

Network	Scales	Stand/ Walk	Exerc	Sit On Chair	Crouch/ Reach	On the Floor	Sports	Misc	Total		
		3DPCK	3DPCK	3DPCK	3DPCK	3DPCK	3DPCK	3DPCK	3DPCK	AUC	MPJPE
Location-Maps	0.7,1.0	89.0	78.0	71.6	74.4	48.2	87.5	80.7	76.5	40.8	122.5
ResNet-100	1.0	88.0	74.3	68.3	68.5	41.9	85.5	78.9	73.3	38.5	136.1
Location-Maps	0.7,1.0	88.8	79.7	**76.3**	**75.3**	51.4	**89.1**	**80.9**	**78.1**	**42.0**	119.2
ResNet-50	1.0	88.4	75.8	71.4	68.8	45.4	87.2	80.6	75.0	39.2	132.8
Direct Regression	0.7,1.0	**89.5**	77.1	75.3	74.5	**52.4**	87.1	80.5	77.5	41.1	**113.1**
ResNet-100	1.0	89.1	74.3	72.6	67.9	49.7	86.4	79.9	75.3	39.6	122.2

Table 5.2: Results on MPI-INF-3DHP test set with the bounding-box corners randomly jittered between +/- 40px to emulate noise from a bounding-box estimator. The fully-convolutional formulation is more robust than a comparative fully-connected formulation. The evaluation is at a single scale (1.0).

Method	Stand/ Walk	Exerc	Sit On Chair	Crouch/ Reach	On the Floor	Sports	Misc	Total		Total Delta	
	3DPCK	3DPCK	3DPCK	3DPCK	3DPCK	3DPCK	3DPCK	3DPCK	AUC	3DPCK	AUC
Location-Maps (ResNet-100)	87.7	71.4	65.2	65.6	38.2	84.6	78.4	71.5	37.4	-1.9	-1.0
Location-Maps (ResNet-50)	86.6	71.6	66.6	65.5	41.0	85.6	79.4	72.1	37.4	-2.9	-1.8
Direct Regression (Resnet-100)	86.9	66.9	70.3	62.2	45.3	84.5	77.7	71.9	37.0	-3.4	-2.6

ily with the direct pose vector regression approach described in Chapter 4, on the MPI-INF-3DHP dataset, using 3D Percentage of Correct Keypoints metric (3D PCK @150mm) discussed in Chapter 3. Both methods are trained on the same data (Human3.6m + MPI-INF-3DHP), as detailed in Section 5.3.2, to be compatible in terms of the camera viewpoints selected, and use ResNet100 as the base architecture for a fair comparison. Table 5.1 shows the results of the raw 3D predictions from both methods, using ground-truth bounding-box cropped frames. The results of both are comparable. The slight increase in accuracy on going to a 50-layer network is possibly due to the better gradient estimates coming from larger mini-batches that can be fit into memory while training, on account of the smaller size of the network. Due to Location-Map based pose inference being strongly linked to image appearance, there is a decrease in accuracy for activities with significant occlusion, such as sitting and lying on the floor. Additionally, the Mean Per Joint Position Error (MPJPE) numbers in mm are also reported. Note that MPJPE is not a robust measure, and is heavily influenced by large outliers, and hence the worse performance on the MPJPE measure (119.2mm vs 113.1mm) despite the better 3D PCK results (78.1% vs 77.5%) with ResNet-50.

For Human3.6m, using the protocol as in earlier work (Pavlakos et al. 2017; B. Tekin et al. 2017), all actions and cameras for subject number 9 and 11 are evaluated, and the Mean Per Joint Position Error (mm) for root-relative 3D joint positions reported. See Table 5.3. Note that despite the occasional large outliers affecting the MPJPE measure, unfiltered Location-Map predictions are still better than most of the existing methods.

Per-jiont Analysis: The joint-wise breakup of accuracy of Location-Map ResNet-100 predictions vs direct regression predictions in Figure 5.6 shows that the accuracy of wrist predictions with Location-Maps is significantly better, while the accuracy of the head is markedly worse. Figure 5.5 shows a visual comparison between the two methods, further demonstrating the strong tie-in to image appearance that the Location-Map formulation affords, as well as the downsides of the

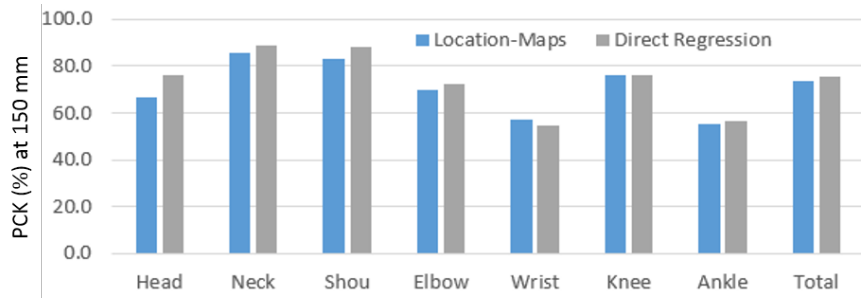

Figure 5.6: Joint-wise breakdown of the accuracy of Location-Map and direct regression based predictions with ResNet-100 based CNN on MPI-INF-3DHP test set.

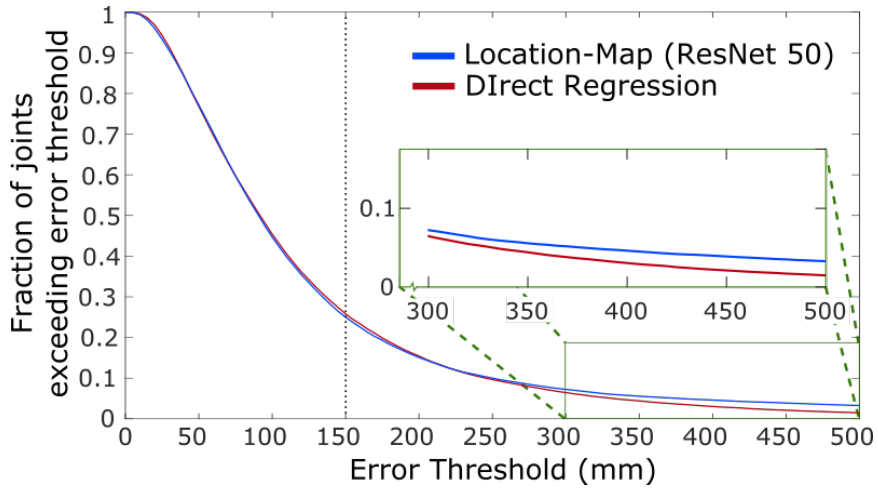

Figure 5.7: Fraction of joints incorrectly predicted on MPI-INF-3DHP test set, as determined by the distance between the predicted joint location and the ground truth joint location being greater than the error threshold. The dotted line marks the threshold for which the 3D PCK numbers are reported. Bottom right part of the curve shows that Location-Map based inference has larger occasional mispredictions, which result in higher MPJPE numbers despite otherwise similar performance.

strong tie-in. Location-Map predictions are prone to occasional large mispredictions when the body joint 2D location detector misfires. It is these large outliers that obfuscate the reported MPJPE numbers. Figure 5.7, which plots the fraction of mispredicted joints vs. the error threshold on MPI-INF-3DHP test set shows that Location-Map predictions have a higher fraction of per-joint mispredictions beyond 300mm. It explains the higher MPJPE numbers compared to direct regression, despite equivalent PCK performance. The various filtering stages employed in the full pipeline ameliorate these large mispredictions.

Effect of Inexact Crops: To demonstrate that the proposed fully-convolutional pose formulation is less sensitive to inexact cropping than networks directly regressing 3D pose vectors, a noisy bounding-box estimator is emulated by jittering the ground-truth bounding-box corners of MPI-INF-3DHP test set uniformly at random in the range of +/- 40 px. This also captures scenarios where one or more end effectors are not in the frame, so a loss in accuracy is expected for all methods. Table 5.2 shows that the direct pose vector regression approach suffers a worse hit in accuracy than the location-map approach, going down by 3.4 PCK, compared to a decrease of only 1.9 PCK with the location-map approach..

Table 5.3: Results of Location-Map based predictions on Human3.6m, evaluated on the ground truth bounding-box crops for all frames of Subject 9 and 11. The Location-Map based networks shown here use only Human3.6m as the 3D training set, and are pretrained for 2D pose prediction. $*^1$ and $*^2$ are identical, except $*^1$ is trained for 17 joints, while $*^2$ is trained for 21 joints similar to the Location-Map networks. The error measure used is Mean Per Joint Position Error (MPJPE) in millimeters. Note again that the error measure used is not robust, and subject to obfuscation from occasional large mispredictions, such as those exhibited by the raw Location-Map predictions.

Method	Direct	Discuss	Eating	Greet	Phone	Posing	Purch.	Sitting	Sit Down	Smoke	Take Photo	Wait	Walk	Walk Dog	Walk Pair	All
Xiaowei Zhou et al. 2015c	87.4	109.3	87.1	103.2	116.2	106.9	99.8	124.5	199.2	107.4	143.3	118.1	79.4	114.2	97.7	113.0
Bugra Tekin et al. 2016b	102.4	147.7	88.8	125.3	118.0	112.3	129.2	138.9	224.9	118.4	182.7	138.8	55.1	126.3	65.8	125.0
Y. Yu et al. 2016	85.1	112.7	104.9	122.1	139.1	105.9	166.2	117.5	226.9	120.0	135.9	117.7	137.4	99.3	106.5	126.5
Ionescu et al. 2014b	132.7	183.6	132.4	164.4	162.1	150.6	171.3	151.6	243.0	162.1	205.9	170.7	96.6	177.1	127.9	162.1
Xingyi Zhou et al. 2016	91.8	102.4	97.0	98.8	113.4	90.0	93.8	132.2	159.0	106.9	125.2	94.4	79.0	126.0	99.0	107.3
Pavlakos et al. 2017	58.6	64.6	63.7	62.4	66.9	57.7	**62.5**	**76.8**	**103.5**	**65.7**	**70.7**	**61.6**	69.0	**56.4**	59.5	**66.9**
D. Mehta et al. 2017a$*^1$	**52.6**	**63.8**	**55.4**	**62.3**	71.8	**52.6**	72.2	86.2	120.6	66.0	79.8	64.0	**48.9**	76.8	**53.7**	68.6
B. Tekin et al. 2017	85.0	108.8	84.4	98.9	119.4	98.5	93.8	73.8	170.4	85.1	95.7	116.9	62.1	113.7	94.8	100.1
Location-Map (ResNet-100)	61.7	77.8	64.6	70.3	90.5	61.9	79.8	113.2	153.1	80.9	94.4	75.1	54.9	83.5	61.0	82.5
Location-Map (ResNet-50)	62.6	78.1	63.4	72.5	88.3	63.1	74.8	106.6	138.7	78.8	93.8	73.9	55.8	82.0	59.6	80.5
Direct Regression [2017]$*^2$	52.4	65.6	59.9	65.8	76.7	53.1	72.7	88.6	117.2	66.9	81.9	67.1	52.4	76.3	56.3	70.5

5.4 Conclusion

This chapter highlights various issues with a direct image to 3D pose regression formulation, and proposes an alternative formulation, termed Location-Maps, which is fully convolutional and directly couples pose inference of each body part to its supporting image evidence. This not only leads to an improved per-frame prediction accuracy for various activity classes, but the formulation predicts 2D body pose along with the 3D pose, which can be used to bootstrap a no-cost bounding-box tracker. This has the advantage that the complete input image does not need to be processed at every time step, which prevents wasteful computations on image regions not occupied by the subject, and does not require a separate person detection step to be run per frame. Further, the formulation is shown to be robust to bounding box jitter, performing better than the direct regression approach under realistic deployment scenarios. These characteristics enable real-time monocular RGB based body pose inference on typical consumer desktops. The formulation suffers from occasional outliers when the underlying 2D pose prediction misfires, but in the context of a motion capture system those can be easily remedied through filtering. Chapter 8 discusses the design of a real-time monocular RGB based single-person motion capture system based on the formulation developed in this chapter, to produce temporally smooth joint angle estimates which can be readily be employed for various applications, such as driving virtual characters in interactive games. Chapter 6 builds upon the Location-Map formulation with several additional insights to allow efficient, and occlusion robust inference in multi-person scenarios.

Chapter 6

Occlusion Robust and Multi-Person 3D Pose Formulations

Moving from the unoccluded single-person scenario to more complex scenes encountered in-the-wild, the formulation should be able to handle occlusions by objects, multiple subjects in the scene, often in close interaction, and inter-personal occlusions, while generalizing to diverse clothing and scene appearance. This makes the problem significantly harder. Since the overall objective is to enable real-time motion capture deployable on a typical consumer desktop computer, the run time cost of the system is an important factor as well in the design of the algorithms.

In This Chapter

- Discussion of the limitations of the location-map formulation from Chapter 5 under partial occlusion, and failings of naïve extension of location-maps to multi-person scenarios (Section 6.1.2)
- Extension of insights from the location-map formulation to develop a pose-map formulation robust to object occlusions (ORPM) and able to handle multiple subjects without needing to scale up the inference cost in direct proportion to the number of people in the scene (Section 6.2)
- Discussion of the limitations of ORPM under inter-personal occlusion, and proposal of an alternative pose formulation that reasons only about visible body parts directly supported by image evidence, and delegates the reasoning of occluded body parts to a second small stage operating per-person with fully body context (Section 6.3)

The content of this chapter is based on D. Mehta et al. 2018 and D. Mehta et al. 2020.

6.1 Related Work

6.1.1 Multi-person 2D Pose Estimation Revisited

Multi-person 2D pose estimation approaches are discussed in Chapter 2. Multi-person 2D pose estimation methods can be broadly classified into bottom-up and top-down approaches. Top-down approaches detect individuals in a scene and utilize single-person 2D pose approaches for estimating the pose for each individual (Gkioxari et al. 2014; Iqbal et al. 2016; Papandreou et al. 2017; Pishchulin et al. 2012; Sun et al. 2011).

Bottom-up approaches localize the body parts of all subjects and subsequently group the body parts together into individuals. This part association can be done by predicting part locations and their identity embeddings together (Newell et al. 2017), or by solving a graph cut problem (Insafutdinov et al. 2017; Pishchulin et al. 2016). Cao et al. 2017 predict joint locations and part affinities (PAFs), which are 2D vectors linking each joint to its parent. PAF based greedy part association enables real time multi-person 2D pose estimation. The formulations presented in this chapter will utilize PAFs to localize and associate body parts in 2D to support 3D pose inference, and the contributions presented would be with regards to the 3D pose encoding. The 2D pose estimation formulation can be swapped out for alternative bottom-up formulations.

Top-down and bottom-up approaches, each have their own set of advantages and disadvantages. Top-down approaches which extract bounding boxes in the image and pass a scale normalized bounding box to a single person pose estimator yield more accurate pose estimates than bottom-up approaches, as shown in ibid. However, such a two stage approach is expensive, and the inference time is directly proportional to the number of subjects in the scene. Top-down approaches inspired by Faster RCNN [2015], where the bounding-box (ROI) is applied to the features instead of the input image, are fast and with a fixed inference time per frame, but lose out on the improved accuracy owing to input scale normalization of each subject. Also, with bounding-box/ROI based top-down approaches, there are multiple overlapping proposals for the same subject (Gregory Rogez et al. 2017), which need to be fused in a post-processing step. Additionally, with multiple subjects in close proximity, it may not be clear which individual to focus on in a given bounding-box/ROI proposal. For these reasons, the 3D pose formulations developed in this chapter follow a bottom-up approach, and also employ a bottom-up 2D pose formulation.

For a more thorough discussion of multi-person 2D pose estimation, also refer to Chapter 2.

6.1.2 Location-Maps Revisited

The location-map formulation presented for the singe person case in the previous chapter utilizes joint specific feature channels to store the 3D coordinates x_j, y_j, z_j of joint j in the respective joint's location maps X_j, Y_j, Z_j at the joint's 2D pixel location (u_j, v_j). Per joint heatmaps H_j encode the joint detection confidences, and the pixel location of the peak is chosen as the joint's 2D detection location.

Although this simple location-map formulation enables full 3D pose inference, it has several shortcomings. First, it assumes that all joints of a person are fully visible, and breaks down under partial occlusion, which is common in general scenes. Second, efficient extension to multiple people is not straightforward. Introducing separate location-maps per person requires dynamically changing the number of outputs or enforcing an upper limit on the number of subjects in the scene. Pre-processing

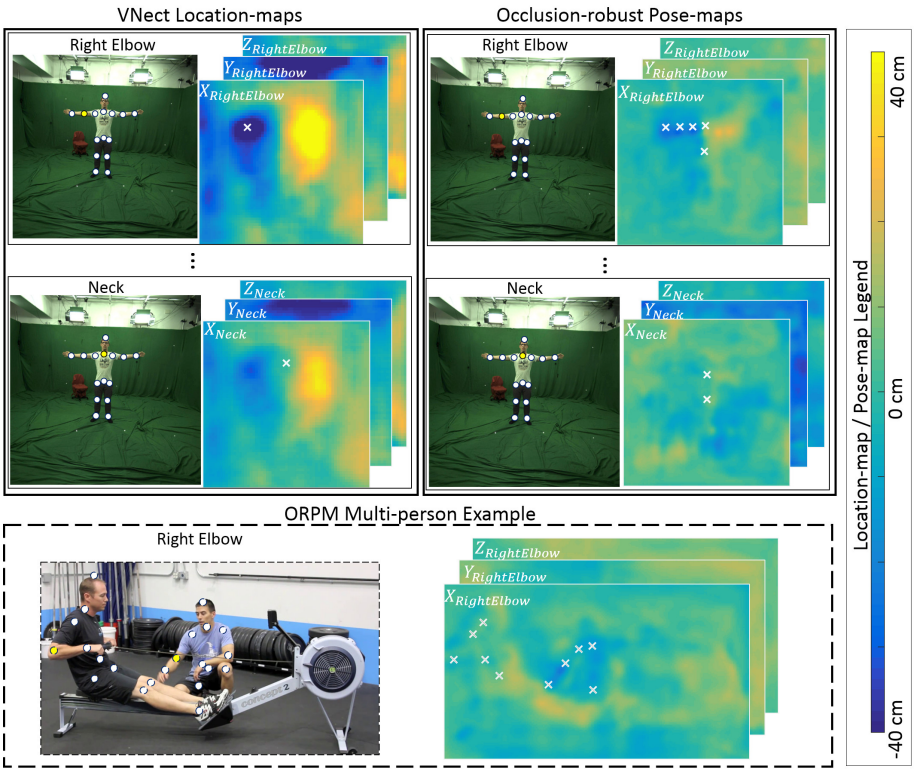

Figure 6.1: Multiple levels of selective redundancy in Occlusion-robust Pose-map (ORPM) formulation. Location-maps (D. Mehta et al. 2017b) (left) only support readout at a single pixel location per joint type. ORPMs (middle) allow the complete body pose to be read out at torso joint pixel locations (neck, pelvis). Further, each individual limb's pose can be read out at all 2D joint pixel locations of the respective limb. This translates to read-out of each joint's location being possible at multiple pixel locations in the joint's location map. The example at the bottom shows how 3D locations of multiple people are encoded into the same map per joint and no additional channels are required.

the input frame by extracting person bounding boxes before feeding it to a location-map based single person pose estimator makes the inference time scale in proportion to the number of subjects in the scene, as discussed earlier, which would make it hard to achieve real-time performance on scenes with many subjects. Similar reasons prevent the extension of the volumetric pose formulation of Pavlakos et al. 2017 to the multi-person case.

6.2 ORPMs - An Occlusion Robust Bottom-up 3D Pose Formulation

As discussed previously, the goal is to develop a bottom-up formulation with a fixed number of outputs regardless of the number of people in the scene. Further, unlike 2D pose approaches that make no prediction for occluded joints, the 3D pose formulation should output predictions for the complete skeleton regardless of the extent of occlusion. Occlusion-Robust Pose Maps (ORPMs) are a first attempt at such a formulation.

The key insight of ORPM is the incorporation of *redundancy* into the location-maps. The body is represented decomposed into torso, four limbs, and head (see Figure 6.2). Occlusion-robust

pose-maps (ORPMs) support multiple levels of redundancy: (1) they allow the read-out of the complete *base pose* $\mathbf{P} \in \mathbb{R}^{3\times n}$ at one of the torso joint locations (neck or pelvis), (2) the base pose (which – being similar to a direct pose vector regression formulation – may not capture the full extent of articulation, as discussed in Chapter 5) can be further refined by reading out the head and individual limb poses where 2D detections are available, and (3) the complete limb pose can be read out at any 2D joint location of that limb. Together, these ensure that a complete and as articulate as possible pose estimate is available even in the presence of heavy occlusions of the body (see Figure 6.2). In addition, the redundancy in ORPMs allows to encode the pose of multiple partially overlapping persons without loss of information, thus removing the need for a variable number of output channels. See Figure 6.1.

Naïve Redundancy: It is easy to see that a naïve approach to introduce redundancy by allowing full pose read-out at all body joint locations breaks down for interacting and overlapping people, leading to supervision and *read-out* conflicts in all location-map channels. Where body parts of two subjects are in close proximity, the output map would be expected to encode two entirely different poses only a few pixels apart. It is the selective introduction of redundancy that restricts these conflicts to pose-map channels of limbs of the same kind, i. e. , wrist of one person in the proximity of a knee of another person cannot cause read-out conflicts because the pose of those joints is encoded in their respective pose-maps. If the complete pose was encoded at each joint location, there would be conflicts for each pair of proximate joints across people.

Next, ORPMs (Sec. 6.2.1) are formally defined, and the inference process explained in detail (Sec. 6.2.2).

6.2.1 Formulation

Given a monocular RGB image I of size $W \times H$, the root-relative 3D pose $\mathcal{P} = \{\mathbf{P^{3D}}_i\}_{i=1}^{m}$ for each of the m persons in the image is sought. Here, $\mathbf{P^{L}}_i \in \mathbb{R}^{3\times J}$ describes the 3D locations of the $J = 17$ body joints of person i. In the rest of the chapter, the 3D pose of each person is expressed as kinematic-parent-relative joint locations $\mathbf{P}_i$, as indicated in Figure 6.2, and can trivially be converted to root-relative locations. The body joints are grouped into pelvis, neck, head, and a set of limbs: $L = \{\{\text{shoulder}_s, \text{elbow}_s, \text{wrist}_s\}, \{\text{hip}_s, \text{knee}_s, \text{ankle}_s\} \mid s \in \{\text{right}, \text{left}\}\}$. The 3D locations of the joints are then encoded in the occlusion-robust pose-maps denoted by $\mathcal{M} = \{\mathbf{M}_j\}_{j=1}^{n}$, where $\mathbf{M}_j \in \mathbb{R}^{W\times H\times 3}$. In contrast to simple location-maps, the ORPM $\mathbf{M}_j$ stores the 3D location of joint j not only at this joint's 2D pixel location $(u,v)_j$ but at a set of 2D locations $\rho(j) = \{(u,v)_{\text{neck}}, (u,v)_{\text{pelvis}}\} \cup \{(u,v)_k\}_{k\in limb(j)}$, where:

$$limb(j) = \begin{cases} l, & \text{if } \exists l \in L \text{ with } j \in l \\ \{\text{head}\}, & \text{if } j = \text{head} \\ \emptyset, & \text{otherwise} \end{cases} . \tag{6.1}$$

Note that—since joint j of all persons i is encoded in $\mathbf{M}_j$—it can happen that read-out locations coincide for different people, i. e. , $\rho_{i1}(j) \cap \rho_{i2}(j) \neq \emptyset$. In this case, $\mathbf{M}_j$ contains information about the person closer to the camera at the overlapping locations. However, due to the built-in redundancy in the ORPMs, a pose estimate for the partially occluded person can still be obtained at other available read-out locations.

To estimate where the pose-maps can be read out, 2D joint *heatmaps* $\mathcal{H} = \{\mathbf{H}_j \in \mathbb{R}^{W\times H}\}_{j=1}^{n}$ heatmaps are also predicted by the network, similar to the location-map formulation developed in the

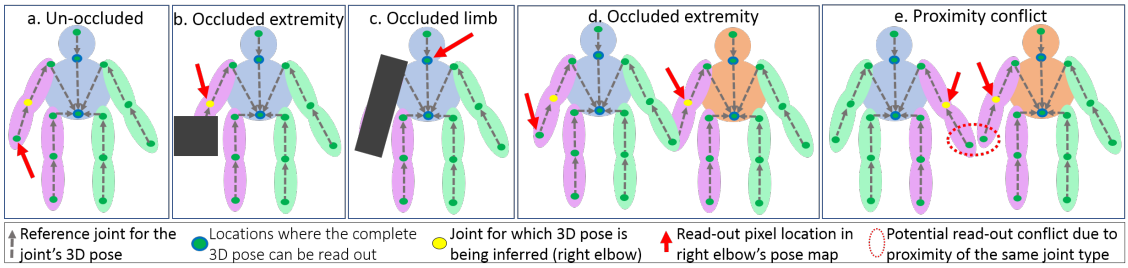

Figure 6.2: Example of the choice of read-out pixel location for right elbow pose under various scenarios. First the complete body pose is read out at one of the torso locations. a.) If the limb extremity is un-occluded, the pose for the entire limb is read out at the extremity (wrist), b.) If the limb extremity is occluded, the pose for the limb is read out at the joint location further up in the joint hierarchy (elbow), c.) If the entire limb is occluded, the limb's pose from base pose read out at one of the torso locations (neck) is retained, d.) Read-out locations indicated for inter-person interaction, e.) If two joints of the same type (right wrist here) overlap or are in close proximity, limb pose read-out is done at a safer isolated joint further up in the hierarchy.

previous chapter. Additionally, *part affinity fields* $\mathcal{A} = \{\mathbf{A}_j \in \mathbb{R}^{W \times H \times 2}\}_{j=1}^{n}$ [2017] which represent a 2D vector field pointing from a joint of type j to its parent are predicted as well. As described earlier, this facilitates association of 2D detections in the heatmaps (and hence read-out locations for the ORPMs) to person identities and enables per-person read-outs when multiple people are present.

6.2.2 Pose Inference

Read-out of 3D pose of multiple people from ORPMs starts with inference of 2D joint locations $\mathcal{P}^{2D} = \{\mathbf{P^{2D}}_i\}_{i=1}^{m}$ with $\mathbf{P^{2D}}_i = \{(u,v)_j^i\}_{j=1}^{n}$ and joint detection confidences $\mathcal{C}^{2D} = \{\mathbf{C^{2D}}_i \in \mathbb{R}^n\}_{i=1}^{m}$ for each person i in the image. Explicit 2D joint-to-person association is done with the predicted heatmaps $\mathcal{H}$ and part affinity fields $\mathcal{A}$ using the approach of Cao et al. 2017. Next, the 2D joint locations $\mathcal{P}^{2D}$ and the joint detection confidences $\mathcal{C}^{2D}$ are used in conjunction with ORPMs $\mathcal{M}$ to infer the 3D pose of all persons in the scene.

Read-Out Process: By virtue of the ORPMs 3D joint locations can be read out at select multiple pixel locations as described above. Let the following joints be defined as *extremity joints*: the wrists, the ankles, and the head. The neck and pelvis 2D detections are usually reliable, these joints are most often not occluded and lie in the middle of the body. Therefore, the full base pose is read out at the neck location. If the neck is *invalid* (as defined below) then the full pose is read at the pelvis instead. If both of these joints are invalid, the person is considered to not be visible in the scene and no pose is predicted. While robust, full poses read at the pelvis and neck are similar to the vectorized pose formulation discussed in Chapter 4, and consequently tend to be closer to the average pose in the training data, and do not match the extent of articulation of subject. Hence, for each limb, the limb pose is read out at the extremity joint. Note again that the *complete limb* pose can be accessed at any of that limb's 2D joint locations. If the extremity joint is valid, the limb pose replaces the corresponding elements of the base pose. If the extremity joint is invalid however, the kinematic chain is traversed up and the validity of the other joints of the limb is checked. If all joints of the limb are invalid, the pose for this limb coming from the base pose cannot be further refined. This read-out procedure is illustrated in Figure 6.2 and algorithmically described in Appendix C.

2D Joint Validation: A 2D body joint location $\mathbf{P^{2D}}_i^j = (u,v)_j^i$ of person i is considered a *valid read-out location* iff (1) it is un-occluded, i. e. , has confidence value higher than a threshold t_C, and

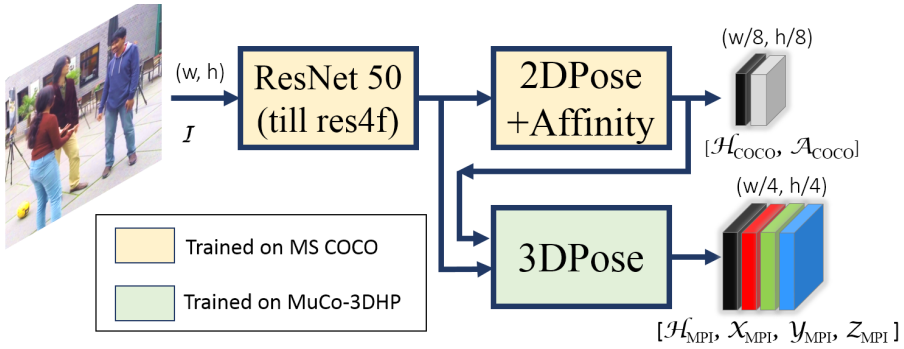

Figure 6.3: The network architecture with *2DPose+Affinity* branch predicting the 2D *heatmaps* $\mathcal{H}_{COCO}$ and *part affinity maps* $\mathcal{A}_{COCO}$ with a spatial resolution of $(W/8, H/8)$, and *3DPose* branch predicting 2D *heatmaps* $\mathcal{H}_{MPI}$ and ORPMs $\mathcal{M}_{MPI}$ with a spatial resolution of $(W/4, H/4)$, for an input image with resolution (W, H).

(2) it is sufficiently far ($\geq t_D$) away from all read-out locations of joint j of other individuals:

$$valid(\mathbf{P^{2D}}_i^j) \Leftrightarrow \mathbf{C^{2D}}_i^j > t_C \wedge ||a - \mathbf{P^{2D}}_i^j||_2 \geq t_D$$
$$\forall \bar{i} = [1:m], \bar{i} \neq i. \forall a \in \rho_{\bar{i}}(j). \tag{6.2}$$

The ORPM formulation together with the occlusion-aware inference strategy with limb refinement enables plausible pose inference even for strongly occluded body parts while exploiting all available information if individual limbs are visible. The performance of the proposed formulation is validated on both the occluded joints as well as the visible joints on the multi-person 3D pose test set MuPoTS-3D described in Chapter 3.

6.2.3 Experimental Details

ResNet-50 (K. He et al. 2016a) is used as the backbone network, and the head of the network splits into two — a *2DPose+Affinity* stream and a *3DPose* stream. The core network and the first branch are trained on MS-COCO (T.-Y. Lin et al. 2014) and the second branch is trained with single-person MPI-INF-3DHP or multi-person composited MuCo-3DHP as per the scenario. The datasets are described in Chapter 3.

The *2DPose+Affinity* stream predicts the 2D heatmaps $\mathcal{H}_{COCO}$ for the MS-COCO body joint set, and part affinity fields $\mathcal{A}_{COCO}$. The *3DPose* stream predicts 3D ORPMs $\mathcal{M}_{MPI}$ as well as 2D heatmaps $\mathcal{H}_{MPI}$ for the MPI-INF-3DHP (D. Mehta et al. 2017a) joint set, which has some overlap with the MS-COCO joint set, but does not include facial keypoint annotations and includes annotations for hands, toes and spine. The pose read-out locations as described previously, are restricted to the common minimum joint set between the two, indicated by the circles in Figure 6.2.

Training Data Processing: Following the compositing approach described in Chapter 3, composites of MuCo-3DHP are created using 12 out of the 14 available camera viewpoints (using only 1 of the 3 available top down views) in MPI-INF-3DHP (ibid.) training set. Overall 400k composite frames of MuCo-3DHP are created, of which half are without any appearance augmentation. For training, the composited frame is cropped around the subject closest to the camera, and rotation, scale, and bounding-box jitter augmentation are applied. Since the data was originally captured in a relatively restricted space, the likelihood of there being multiple people visible in the crop around the main

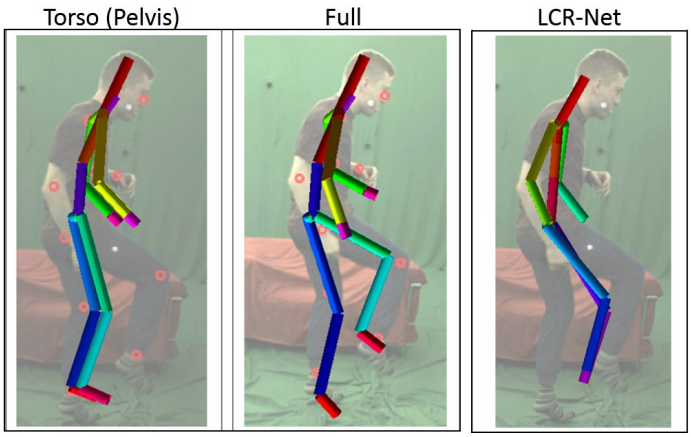

Figure 6.4: Qualitative comparison of pose read out at torso locations (neck/pelvis) and the full pose read-out scheme. LCR-net [2017] prediction is also shown, and exhibits a tendency towards neutral pose similar to the limited articulation of the pose read out at torso locations. The full read-out scheme addresses the issue.

person is high. The combination of scale augmentation, bounding-box jitter, and cropping around the subject closest to the camera results in many examples with truncation from the frame boundary, in addition to the inter-person occlusions occurring naturally due to the compositing. See Figure 3.7 for example composites from MuCo-3DHP.

Training Loss: The 2D heatmaps $\mathcal{H}_{COCO}$ and $\mathcal{H}_{MPI}$ are trained with per-pixel $L2$ loss comparing the predictions to the reference which has unit peak Gaussians with a limited support at the ground truth 2D joint locations, as is common. The part affinity fields $\mathcal{A}_{COCO}$ are similarly trained with a per-pixel $L2$ loss, using the framework made available by Cao et al. 2017. While training ORPMs with *MuCo-3DHP*, per joint type j, for all subjects i in the scene, a per-pixel $L2$ loss is enforced in the neighborhood of all possible read-out locations $\rho_i(j)$. The loss is weighted by a limited support Gaussian centered at the read-out location, similar to location-map training.

Training Details: The network is trained using the Caffe (Jia et al. 2014) framework. The core network's weights were initialized with those trained for 2D body pose estimation on MPI (Mykhaylo Andriluka et al. 2014) and LSP (Johnson et al. 2010, 2011) datasets as proposed in D. Mehta et al. 2017a, and described in Chapter 4. The core network and the *2DPose + Affinity* branch are trained for multi-person 2D pose estimation using the framework provided by Cao et al. 2017. The AdaDelta solver, with a momentum of 0.9 and weight decay multiplier of 0.005 is used, with a batch size of 8. The network is trained for 640k iterations with a cyclical learning rate ranging from 0.1 to 0.000005. The *3DPose* branch is trained with the core network and *2DPose + Affinity* branch weights frozen. A batch size of 6 is used, and trained for 360k iterations with a cyclical learning rate ranging from 0.1 to 0.000001.

6.2.4 Results and Comparisons

Even though the main goal of the method is *multi-person* 3D pose estimation in general scenes, which exhibits specific and more difficult challenges than single-person pose estimation, single-person pose estimation datasets can be used to validate the usefulness of various components of the ORPM formulation, in conjunction with multi-person pose benchmarks.

Figure 6.6 presents qualitative results of ORPM based multi-person pose estimation, showcasing the

Table 6.1: Comparison of results on MPI-INF-3DHP (D. Mehta et al. 2017a) test set. *Percentage of Correct Keypoints measure in 3D* (@150mm) for select activities, and the total 3DPCK and the Area Under the Curve for all activities are reported. The evaluations use multi-scale augmentation. Complete activity-wise breakdown is in Appendix C

	Sit	Crouch	Total	
Method	PCK	PCK	PCK	AUC
D. Mehta et al. 2017a	75.3	74.5	77.5	41.1
Location-map [2017]	76.3	75.3	**78.1**	**42.0**
LCR-net Gregory Rogez et al. 2017	58.5	69.4	59.7	27.6
Zhou et al.Xingyi Zhou et al. 2017	60.7	71.4	69.2	32.5
ORPM Single-Person (Torso)	69.4	69.9	66.4	32.9
ORPM Single-Person (Full)	**79.1**	**77.9**	76.2	38.3
ORPM Multi-Person (Torso)	66.8	66.9	65.0	31.8
ORPM Multi-Person (Full)	76.0	73.2	74.1	36.7

Table 6.2: Sequence-wise evaluation of ORPMs and LCR-net (Gregory Rogez et al. 2017) on multi-person 3D pose test set *MuPoTS-3D*. Both (a) the overall accuracy (3DPCK), and (b) accuracy only for person annotations matched to a prediction are reported.

		TS1	TS2	TS3	TS4	TS5	TS6	TS7	TS8	TS9	TS10	TS11	TS12	TS13	TS14	TS15	TS16	TS17	TS18	TS19	TS20	**Total**
a.)	LCR-net	67.7	49.8	53.4	59.1	67.5	22.8	43.7	49.9	31.1	78.1	50.2	51.0	51.6	49.3	56.2	66.5	65.2	62.9	66.1	59.1	53.8
	ORPM	**81.0**	**59.9**	**64.4**	**62.8**	**68.0**	**30.3**	**65.0**	**59.2**	**64.1**	**83.9**	**67.2**	**68.3**	**60.6**	**56.5**	**69.9**	**79.4**	**79.6**	**66.1**	**66.3**	**63.5**	**65.0**
b.)	LCR-net	69.1	**67.3**	54.6	61.7	**74.5**	25.2	48.4	**63.3**	**69.0**	78.1	53.8	52.2	60.5	60.9	59.1	70.5	76.0	70.0	**77.1**	81.4	62.4
	ORPM	**81.0**	64.3	**64.6**	**63.7**	73.8	**30.3**	**65.1**	60.7	64.1	**83.9**	**71.5**	**69.6**	**69.0**	**69.6**	**71.1**	**82.9**	**79.6**	**72.2**	76.2	**85.9**	**69.8**

ability to handle complex in-the-wild scenes, and some degree of inter-personal occlusion.

Analysis of Limb Pose Read-out Scheme

The limb pose read-out strategy is validated on the single person MPI-INF-3DHP test set quantitatively and qualitatively. The base pose read out at neck or pelvis is similar to direct 3D pose vector regression, in that it does not tie pose inference to direct image evidence. The base pose read out at one of the torso joints tends towards the mean pose of the training data. Hence, for poses with significant articulation of the limbs, the base pose does not provide an accurate pose estimate, especially for the end effectors. On the other side, empirically the limb poses read out further down in the kinematic chain, i. e. , closer to the extremities, include more detailed articulation information for that limb. The proposed read-out process exploits this fact, significantly improving overall pose quality by limb refinement when limbs are available. Table 6.1 shows that the benefit of the full read-out is consistent over all metrics independent of whether it is trained on single-person (MPI-INF-3DHP) or multi-person data (MuCo-3DHP), with a ≈10 3DPCK advantage over torso read-out. See Figure 6.4.

Comparison with Prior Art

For un-occluded sequences, ORPM inference results are comparable to methods designed for single-person cases, while showing much better performance for sequences with strong occlusions. ORPM

based inference outperforms the only other multi-person method (LCR-net (Gregory Rogez et al. 2017)) known at the time, quantitatively and qualitatively on both single-person and multi-person tasks. For fairness of comparison, in all evaluations, the predictions from LCR-net are re-targeted to a skeleton with bone-lengths matching the ground truth.

Multi-Person Pose Performance: The proposed *MuPoTS-3D* test set (see Chapter 3) to evaluate multi-person 3D pose performance in general scenes for ORPM inference and LCR-net [2017]. In addition, location-map [D. Mehta et al. 2017b] based inference is also evaluated on images cropped with the ground truth bounding box around the subject. As discussed in Chapter. 3, the predictions are evaluated for all subjects that have 3D pose annotations available. If an annotated subject is missed by the evaluated method, all of its joints are considered to be incorrect in the 3DPCK metric. Table 6.2(a) reports the 3DPCK metric for all 20 sequences when taking all available annotations into account. ORPM based inference performs significantly better than LCR-net for most sequences, while being comparable for a few, yielding an overall improved performance of 65.0 3DPCK vs 53.8 3DPCK for LCR-net. For a joint-wise breakdown of the overall accuracy, please refer to Appendix C.

Overall, ORPM based approach detects 93% of the annotated subjects, whereas LCR-net was successful for 86%. This is an additional indicator of performance. Even ignoring the undetected annotated subjects, ORPM outperforms LCR-net in terms of 3D pose error (69.8 vs 62.4 3DPCK, and 132.5 vs 146 mm MPJPE).

The location-map formulation is evaluated on ground truth crops of the subjects, and therefore it operates at a 100% detection rate. In contrast, ORPM inference without any ground truth crops, and with missed person detections counting as all joints wrong, still achieves better accuracy (65.0 vs 61.1 3DPCK, 30.1 vs 27.6 AUC).

Single-Person Pose Performance: On the MPI-INF-3DHP test dataset (see Table 6.1) ORPM based inference trained separately on MPI-INF-3DHP (single-person) and MuCo-3DHP (multi-person) is compared to two single-person methods, location-maps (ibid.), Xingyi Zhou et al. 2017, and LCR-net as the only other multi-person approach known at the time. ORPM based approach trained on multi-person data (74.1 3DPCK) performs worse than the single-person version (76.2 3DPCK) due to it being a more challenging task. Nevertheless, both the versions consistently outperform Zhou et al. (69.2 3DPCK) and LCR-net (59.7 3DPCK) over all metrics. LCR-net predictions have a tendency to be conservative about the extent of articulation of limbs as shown in Figure 6.6. In comparison to location-maps (78.1 3DPCK), ORPMs (76.2 3DPCK) achieve an average accuracy that is comparable. ORPMs outperform existing methods by ≈2-3 3DPCK for activities which exhibit significant self- and object-occlusion like *Sit on Chair* and *Crouch/Reach*. For the full activity-wise breakdown, see Appendix C.

For detailed comparisons on Human3.6m (Ionescu et al. 2014b), also refer to Appendix C. On Human3.6m, ORPM based approach at 69.6mm MPJPE performs ≈17mm better than LCR-net Gregory Rogez et al. 2017 (87.7mm), and outperforms the location-map (D. Mehta et al. 2017b) (80.5mm) formulation by ≈10mm. ORPM results are comparable to (at the time) recent state-of-the-art results of Pavlakos et al. 2017 (67.1mm), Martinez et al. 2017 (62.9mm), Xingyi Zhou et al. 2017 (64.9mm), D. Mehta et al. 2017a (68.6mm) and B. Tekin et al. 2017 (70.81mm), and better than the recent results from Xiaohan Nie et al. 2017 (79.5mm) and Tome et al. 2017 (88.39mm).

Occlusion evaluation: Occlusion robustness of the ORPM approach is demonstrated through creation of synthetic random occlusions on the MPI-INF-3DHP test set. The synthetic occlusions cover about 14% of the joints. Both single-person and multi-person variants of ORPM outperform the location-map formulation for occluded joints (62.8 vs 64.0 vs 53.2 3DPCK) by a large margin,

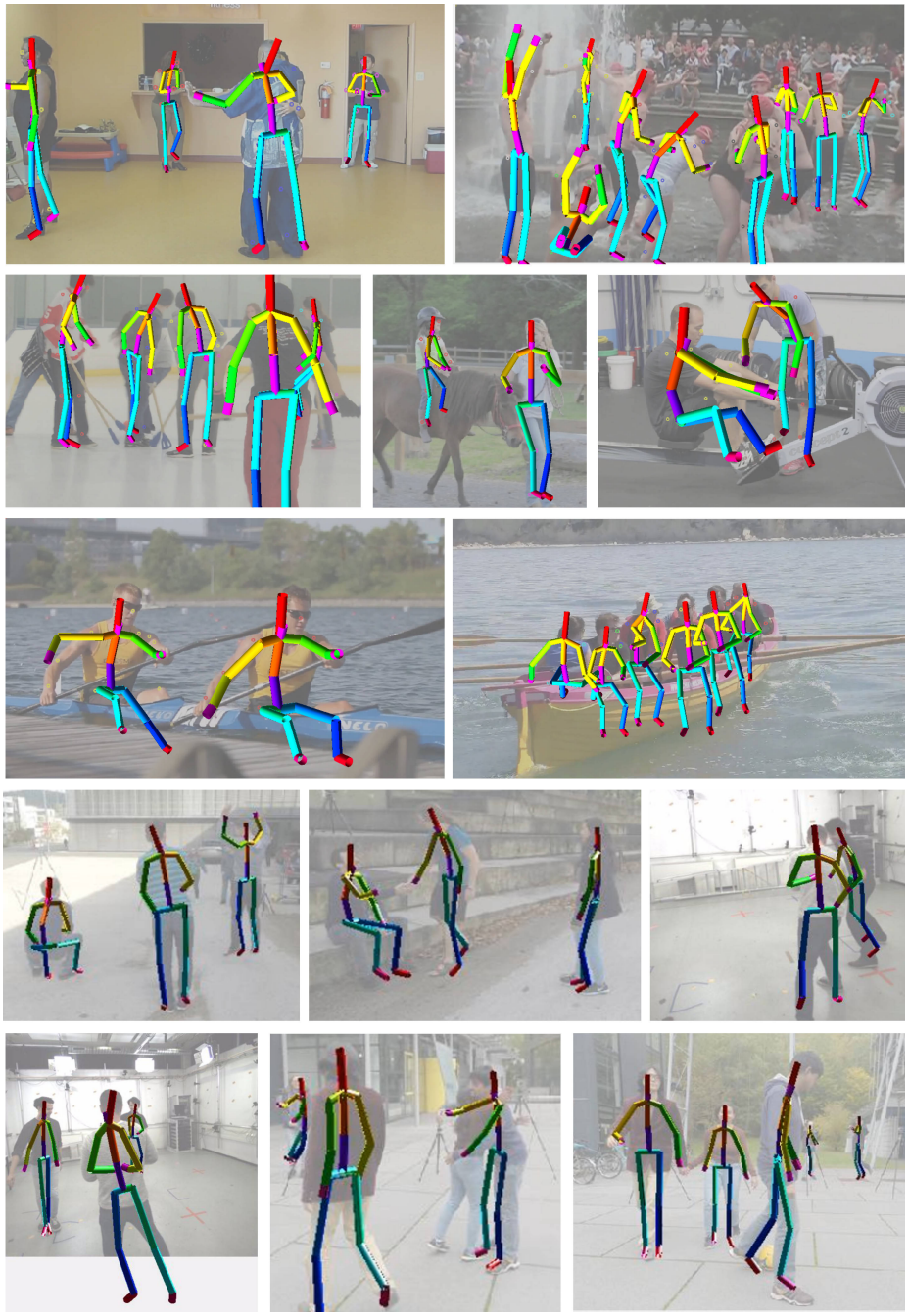

Figure 6.5: More qualitative results of ORPM approach on MPI 2D pose dataset (Mykhaylo Andriluka et al. 2014) and the proposed MuPoTS-3D test set.

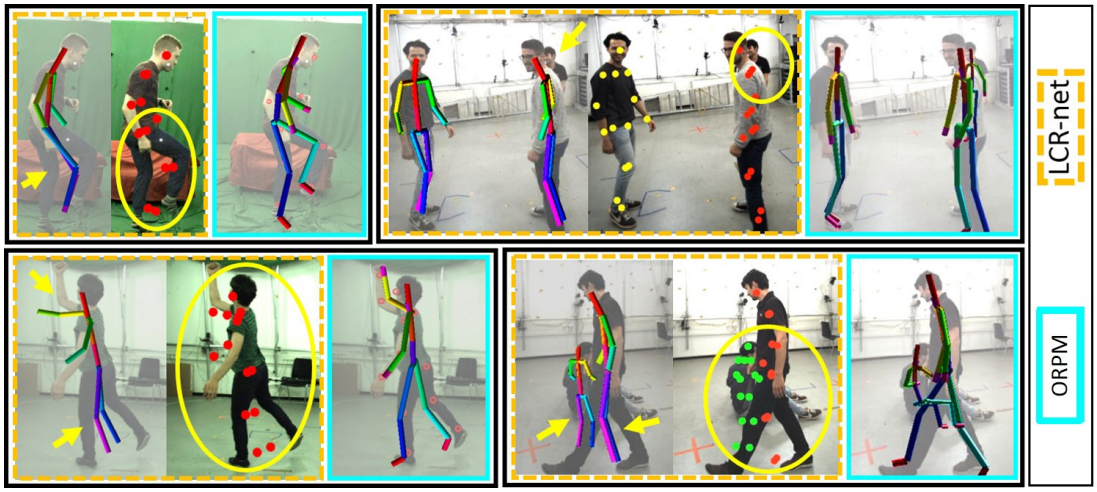

Figure 6.6: Qualitative comparison of LCR-net [2017] and ORPM based approach. LCR-net output is limited in the extent of articulation of limbs, tending towards neutral poses. LCR-net also has more detection failures under significant occlusion.

and are comparable for un-occluded joints (67.0 vs 71.0 vs 69.4 3DPCK). Note again that the single-person variant is trained on similar data as the location-map formulation, and the occlusion robustness is inherent to the formulation. See Appendix C for a more detailed breakdown by test sequence.

The per-joint occlusion annotations from *MuPoTS-3D* can be used to further assess LCR-net and ORPMs under occlusion. Considering both self- and inter-personal occlusions, ≈23.7 % of the joints of all subjects are occluded. ORPMs are more robust than LCR-net on both occluded (48.7 vs 42 3DPCK) and un-occluded (70.0 vs 57.5 3DPCK) joints.

6.2.5 Shortcomings of ORPM Inference

The ORPM formulation handles overlapping joints of the same type by only supervising the one closest to the camera. However, when joints of the same type are in close proximity (but not overlapping) the ground-truth ORPM for those may transition sharply from one person to the other, which are hard to regress and may lead to inaccurate predictions. One possible way to alleviate the issue is to increase the resolution of the output maps, though it comes at the cost of increase in inference time. Another source of failures is when 2D joints are mis-predicted or mis-associated. Though ORPM based pose inference is robust to significant occlusions by other objects, and to a large degree robust to interpersonal occlusions as well when dissimilar body parts overlap, read-out conflicts cannot be avoided when the same body part for different individuals overlaps or is in close proximity. See Figure 6.6 for qualitative results.

Thus, an alternative pose formulation is presented in the next section, to alleviate the shortcomings of the ORPM formulation.

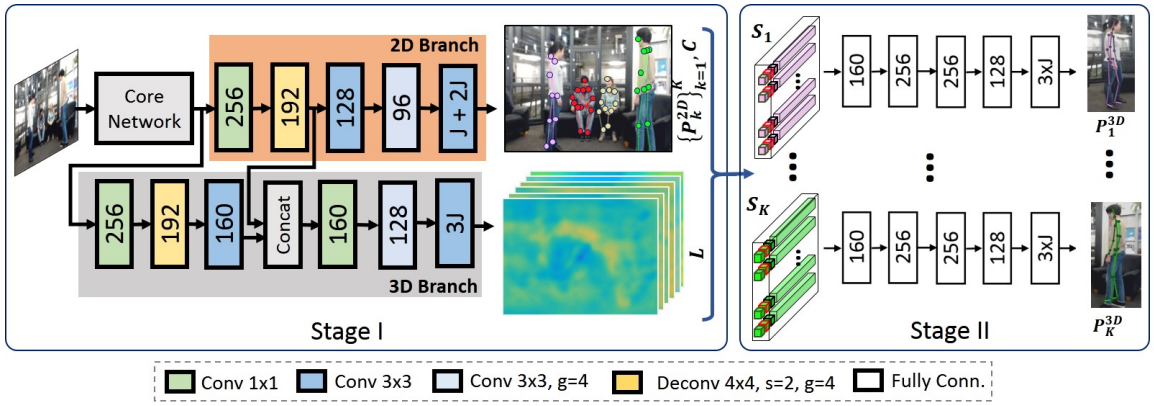

Figure 6.7: Two stage design of the XNect formulation for per-frame 3D pose prediction in multi-person scenarios. *Stage I* is a fully-convolutional network that infers 2D pose and intermediate 3D pose encoding for visible body joints. The 3D pose encoding for each joint only considers local context in the kinematic chain. *Stage II* is a lightweight fully-connected network that runs in parallel for each detected person, and reconstructs the complete 3D pose. The network 'lifts' inferred 2D body pose, augmented with joint detection confidences and 3D pose encodings to root-relative full body 3D pose (X_j, Y_j, Z_j), leveraging full body context to fill in occluded joints.

6.3 XNect: Factoring Visible and Occluded Joint Inference Into Separate Stages

The new formulation carries forward some of the essential features of location-map and ORPM formulations developed thus far. Pose inference for each body part is linked to direct image evidence, and similar to ORPMs, some degree of redundancy is included to provide backup in case of encoding and read-out conflicts. However, different from the ORPM formulation, the new formulation is comprised of two stages. The first is a fully-convolutional network similar to ORPMs, that performs local (per body joint) reasoning, but only for visible body parts. The design seeks to limit the already difficult task of image parsing to body parts for which direct image evidence is available. The second stage is a fully-connected network that leverages full body context per detected person in the scene, to produce estimates for the occluded joints, and correct for any potential errors arising out of read-out conflicts, without needing to increase the spatial resolution of the outputs of the first stage.

In prior work, 'lifting' based approaches predict the 3D pose directly from 2D keypoints predicted from the input image (Martinez et al. 2017). This has the advantage that in-the-wild 2D pose datasets are easier to obtain annotations for, and the lifting can be learned from MoCap data without overfitting on the studio conditions. While this establishes a surprisingly strong baseline, lifting is ill-posed, and can be improved upon by leveraging additional image cues for body-part depth disambiguation. Other work has proposed to augment the 2D pose with relative depth ordering of body joints as additional context to disambiguate 2D to 3D lifting (Pavlakos et al. 2018a; Pons-Moll et al. 2014).

The two stage formulation proposed here can be seen has a hybrid of regression and lifting methods: An encoding of the 3D pose of the visible joints is regressed from the image as fully-convolutional output maps (Stage I), and each body joint only reasons about its immediate kinematic neighbours (local context). This encoding, along with 2D joint detection confidences augments the 2D pose per individual detected in the scene, and is 'lifted' or 'decoded' into a complete 3D body pose by *Stage II* reasoning about all body joints (global context).

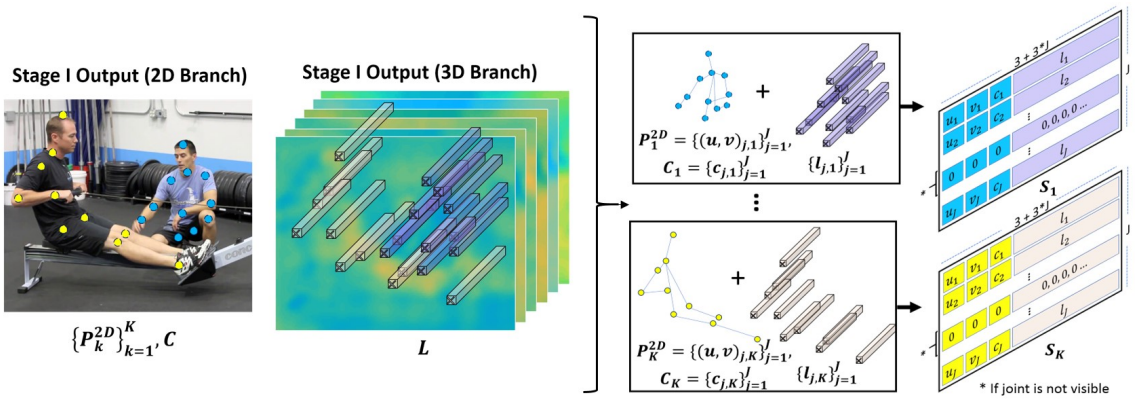

Figure 6.8: Input to *Stage II*: S_k for each detected individual k, is comprised of the individual's 2D joint locations P_k^{2D}, the associated joint detection confidence values C extracted from the 2D branch output, and the respective 3D pose encodings $\{l_{j,k}\}_{j=1}^J$ extracted from the output of the 3D branch.

Similar to the multi-person formulations developed in this chapter, Zanfir et al. 2018b jointly encode the 2D and 3D pose of all subjects in the scene using a fixed number of feature maps. They encode the full 3D pose vector at all the projected pixels of the skeleton, and not just at the body joint locations, which makes the 3D feature space rife with potential encoding conflicts, as discussed in the first part of this chapter. For association, they learn a function to evaluate limb grouping proposals. A 3D pose decoding stage extracts 3D pose features per limb and uses an attention mechanism to combine these into a 3D pose prediction for the limb.

The key insight of the formulation proposed here is to use a representation similar to ORPMs, but only as an intermediate pose encoding, to disambiguate the 2D to 3D lifting of *Stage II*. In this way, the 3D pose encodings are a strong cue for the 3D pose in the absence of occlusions and read-out conflicts, whereas the global context in *Stage II* and the 2D pose help resolve read-out conflicts and fill in the occluded joints. Different from the full pose encoding of ibid., and per limb encoding of ORPMs, the formulation presented here further reduces potential conflicts by only encoding a joint's immediate local context in the kinematic tree, i.e., the body joints directly connected to the joint under consideration by a kinematic linkage (bone).

Similar to the ORPM formulation, given an image I of dimensions $w \times h$ pixels, estimates of the root relative 3D poses $\{P_k^{3D}\}_{k=1}^K$ of the unknown number of K individuals in the scene are sought. $P_k^{3D} \in \mathbb{R}^{3\times J}$ represents the root (pelvis)-relative 3D coordinates of the J body joints, and can equivalently be represented as 3D coordinates of each body part relative to its kinematic parent. The XNect formulation implements this task in two stages, which are detailed in the following.

6.3.1 *Stage I*: Parsing Images for Visible Body Parts

Similar to location-map and ORPM formulations described previously, the first stage is based on a Convolutional Neural Network (CNN). Like the ORPM formulation, the CNN is comprised of a core (or backbone) network that splits into two separate branches for 2D pose prediction and 3D pose encoding, as shown in Figure 6.7. The core network outputs features at $\frac{w}{16} \times \frac{h}{16}$ pixel spatial resolution.

Network Architecture: The outputs of each of the 2D and 3D branches are at $\frac{w}{8} \times \frac{h}{8}$ pixels spatial resolution. The 3D pose branch also makes use of features from the 2D pose branch, and the details

of the branches along with *Stage I* network training are in the following.

Since *Stage I* operates on the complete input frame, the core (or backbone) network is typically the computational bottleneck. The single-person and multi-person formulations discussed thus far in the thesis have relied on a ResNet-50 CNN architecture, as have some recent multi-person 3D pose approaches such as LCRNet++[2019], since it affords a good compromise between inference speed and accuracy. However, for multi-person cases where the complete input frame needs to be processed at every time step, on anything but the top end GPUs, ResNet-50 based systems would not achieve real-time performance. Chapter 7 describes the design of a new network architecture, termed SelecSLS Net, which allows $\approx 1.3-1.4\times$ faster inference than ResNet-50 without compromising on accuracy. SelecSLS Net is used as the core network for *Stage I*. For comparisons of SelecSLS Net against ResNet-34 and ResNet-50 baselines on various benchmarks, refer to Chapter 7.

2D Branch: 2D Pose Prediction and Part Association

2D pose is extracted from 2D heatmaps $\mathcal{H} = \{\mathbf{H}_j \in \mathbb{R}^{W\times H}\}_{j=1}^{J}$, where each map represents the per-pixel confidence of the presence of body joint type j jointly for all subjects in the scene. As with the ORPM formulation, Part Affinity fields $A = \{A_j \in \mathbb{R}^{W\times H\times 2}\}_{j=1}^{J}$ [2017] are used to encode body joint ownership.

If the neck joint (which is likely to be visible in most situations) of an individual is not detected, that individual is not considered in the subsequent stages. For K detected individuals, this stage outputs the 2D body joint locations in absolute image coordinates $P_k^{2D} \in \mathbb{Z}_+^{2xJ}$. Further, the detection confidence $c_{j,k}$ of each body part j and person k obtained from the heatmap maxima is used in the subsequent stage.

3D Branch: Predicting Intermediate 3D Pose Encoding

The 3D branch of the *Stage I* network uses the features from the core network and the 2D branch to predict 3D pose encoding maps $L = \{L_j \in R^{W\times H\times 3}\}_{j=1}^{J}$. The encoding at the spatial location of each visible joint only encapsulates its 3D pose relative to joints it is directly connected to in the kinematic chain.

The encoding in L works as follows: Consider the $1\times 1\times (3\cdot J)$ vector $l_{j,k}$ extracted at the pixel location $(u,v)_{j,k}$ from the 3D output maps L. Here $(u,v)_{j,k}$ is the location of body joint j of individual k. This $1\times 1\times (3\cdot J)$ feature vector is of the dimensions of the full 3D body pose, where the kinematic parent-relative 3D locations of each joint reside in separate channels. Importantly however, and in contrast to ORPMs and Zanfir et al. 2018b, instead of encoding the full 3D body pose, or per-limb pose, at each 2D detection location $(u,v)_{j,k}$, the proposed formulation only encodes the pose of the corresponding joint (relative to its parent) and the pose of its children (relative to itself). In other words, at each joint location $(u,v)_{j,k}$, the supervision of the encoding vector $l_{j,k}$ is restricted to the subset of channels corresponding to the bones that meet at joint j, parent-to-joint and joint-to-child in the kinematic chain. This would be referred to as channel-sparse supervision of $\{l_{j,k}\}_{j=1}^{J}$, and it is noted again that this is distinction from channel-dense supervision where each $l_{j,k}$ encodes the full body pose for person k. Figure 6.9 shows examples for head, neck and right shoulder. Consequently, 3D pose information for all the visible joints of all subjects is still encoded in L, albeit in a spatially distributed manner, and each 2D joint location $(u,v)_{j,k}$ is used to extract its corresponding 3D bones of subject k.

The motivation for such a pose encoding is that the task of parsing in-the-wild images to detect 2D body part heatmaps under occlusion and clutter, as well as grouping the body parts with their respective person identities under inter-personal interaction and overlap is already challenging. Reasoning about the full 3D pose, including occluded body parts, adds further complexity, which not only requires increased representation capacity (thus increasing the inference cost), but also more labeled training data, which is scarce for multi-person 3D pose. The design of the proposed formulation responds to both of these challenges. Supervising only the 3D bones corresponding to each visible joint ensures that mostly local image evidence is used for prediction, where the full body context is already captured by the detected 2D pose. For instance, it should be possible to infer the kinematic-parent relative pose of the upper arm and the fore arm by looking at the region centered at the elbow. This means better generalization and less risk to overfit to dataset specific long-range correlations.

Channel-Sparse Supervision of $l_{j,k}$: Further, the use of channel-sparse (joint-type-dependent) supervision of $l_{j,k}$ is motivated by the fact that convolutional feature maps cannot contain sharp transitions as previously noted, and therefore if a location of the output map encodes the full pose of one subject, a nearby location can not encode the likely very different full pose of another subject. For example, the wrist of one person being in close proximity in the image plane to the shoulder of another person would require the full pose of two different individuals to be encoded in possibly adjacent pixel locations in the output map. Such encoding conflicts often lead to failures of previous methods, as shown in Figure 6.10. In contrast, channel-sparse encoding in L does not lead to encoding conflicts when different joints of separate individuals are in spatial proximity or even overlap in the image plane, because supervision is restricted to the channels corresponding to the body joint type. Consequently, the target output maps are smoother without sharp transitions, and more suitable for representation by CNN outputs. Section 6.3.3 shows the efficacy of channel-sparse supervision for $\{l_{j,k}\}_{j=1}^{J}$ over channel-dense supervision across various 2D and 3D pose benchmarks. Importantly, unlike ibid. and the ORPM formulation discussed previously, the 2D pose information is not discarded, and is utilized as additional relevant information for 3D pose inference in *Stage II*, allowing for a compact and fast network. This makes it more suited for a real-time system than, for instance, the attention-mechanism-based inference scheme of ibid.

For each individual k, the 2D pose P_k^{2D}, joint confidences $\{c_{j,k}\}_{j=1}^{J}$, and 3D pose encodings $\{l_{j,k}\}_{j=1}^{J}$ at the visible joints are extracted and input to Stage II, which uses a fully-connected decoding network that leverages the full body context that it available to it to give the complete 3D pose with the occluding joints filled in. Details of Stage II are in Section 6.3.2.

Stage I Training

The *Stage I* network is trained in multiple stages. First the core network and the 2D pose branch are trained for single person 2D pose estimation on the MPII [2014] and LSP [2010; 2011] single person 2D datasets. Then, using these weights as initialization, it is trained for multi-person 2D pose estimation on MS-COCO [2014]. Then the 3D pose branch is added and the two branches are individually trained on crops from MS-COCO and the proposed composited multi-person dataset MuCo-3DHP, with the core network seeing gradients from both datasets via the two branches. Additionally, the 2D pose branch sees supervision from MuCo-3DHP dataset via heatmaps of the common minimum joint set between MS-COCO and MuCo-3DHP. Pretraining on multi-person 2D pose data before introducing the 3D branch was experimentally observed to be important for the convergence of training.

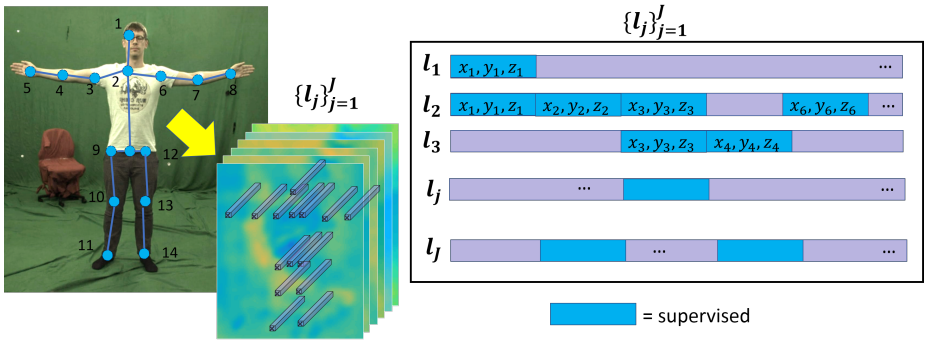

Figure 6.9: The supervision for the $1 \times 1 \times (3 \cdot J)$ 3D pose encoding vector l_j at each joint j is dependent on the type of the joint. l_j only encodes the 3D pose information of joint j relative to the joints it is directly connected to in the kinematic chain. This results in a channel-sparse supervision pattern as shown here, as opposed to each l_j encoding the full body pose. See Section 6.3.1.

6.3.2 *Stage II*: From Partial 2D and 3D Pose Cues to Full Body 3D Pose

Stage II uses a lightweight fully-connected network to predict the root-relative 3D joint positions $\{P_k^{3D}\}_{k=1}^{K}$ for each individual considered visible after *Stage I*. Before feeding the output from *Stage I* as input, the 2D joint position predictions P_k^{2D} are converted to a representation relative to the neck joint. For each individual k, at each detected joint location, we extract the $1 \times 1 \times (3 \cdot J)$ 3D pose encoding vector $l_{j,k}$, as explained in the preceding section. The input to *Stage II*, $S_k \in \mathbb{R}^{J \times (3+3 \cdot J)}$, is the concatenation of the neck relative $(u, v)_{j,k}$ coordinates of the joint, the joint detection confidence $c_{j,k}$ and the feature vector $l_{j,k}$, for each joint j. If the joint is not visible, zero vectors of appropriate dimensions are used instead (see Figure 6.8). *Stage II* comprises a 5-layer fully-connected network, which converts S_k to a root-relative 3D pose estimate P_k^{3D} (see Figure 6.7).

Providing the 2D joint positions and part confidences along with the feature vectors as input to the *Stage II* network allows it to correct any conflicts that may arise. See Figure 6.10 for a visual comparison of results against D. Mehta et al. 2018.

The inference time for *Stage II* with a batch size of 10 is 1.6ms on an Nvidia K80, and 1.1ms on a TitanX (Pascal).

Stage II Training

The *Stage II* network is trained on uncropped frames from MuCo-3DHP (ibid.). 2D pose and 3D pose encoding predictions from *Stage I* are obtained on these frames. Then for each detected individual, the ground-truth root-relative 3D pose is used as the supervision target for $\{(X_j, Y_j, Z_j)\}_{j=1}^{J}$. Since the pose prediction can be drastically different from the ground truth when there are severe occlusions, smooth-L1 (Ren et al. 2015) loss us used to mitigate the effect of such outliers. In addition to providing an opportunity to reconcile the 3D pose predictions with the 2D pose, another advantage of a second stage trained separately from the first stage is that the output joint set can be made different from the joint set used for *Stage I*, depending on which dataset was used for training *Stage II* (joint sets typically differ across datasets). For the datasets considered for training various stages, though there are no 2D predictions for foot tip, the 3D pose encoding for ankle encodes information about the foot tip, which is used in *Stage II* to produce 3D predictions for foot tips.

6.3.3 Results and Comparisons

Performance on Single Person 3D Pose Datasets

Table 6.4 compares the 3D pose output from *Stage II* against other single person methods on the MPI-INF-3DHP benchmark dataset using 3D Percentage of Correct Keypoints (**3DPCK**, higher is better), Area under the Curve (**AUC**, higher is better) and mean 3D joint position error (**MJPE**, lower is better). Since the first stage network is meant to operate on uncropped frames, the evaluation here uses uncropped input frames, and as with evaluations thus far, no rigid alignment of the 3D predictions to the ground truth is used. Note that the method is trained for the multi-person case, and evaluated on single-person benchmark. It is not a separate model trained specifically for single-person case. Even though it is trained for the much harder multi-person pose estimation task, the proposed approach outperforms most of the recent dedicated single person methods in terms of accuracy.

Similarly with Stage II trained on Human3.6m (Table 6.6), the proposed method compares favourably to recent approaches designed to handle single-person and multi-person scenarios. Further, this is an example of the ability of the 2 stage approach to adapt to different datasets by simply retraining the inexpensive *Stage II* network.

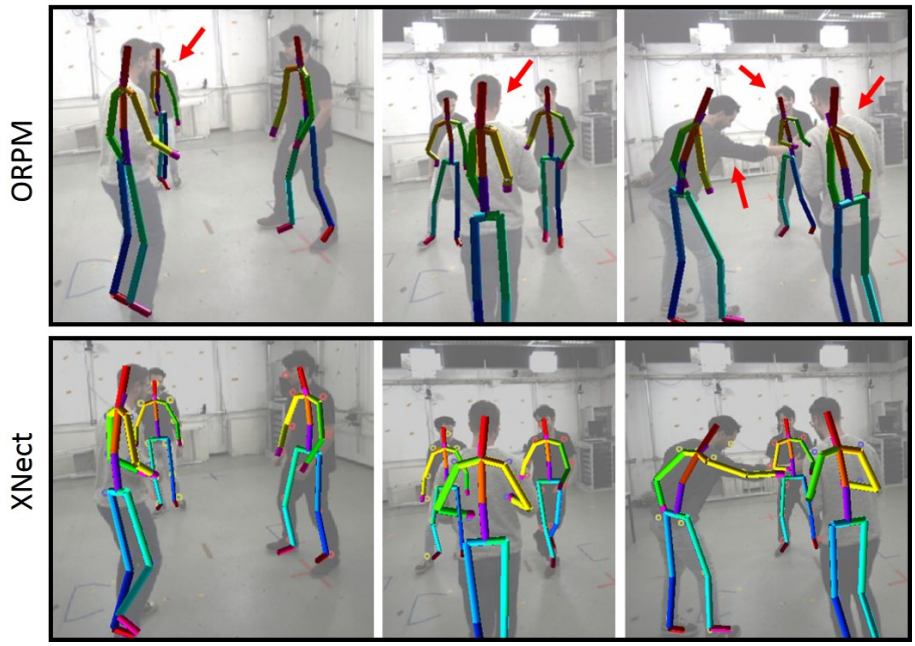

Figure 6.10: XNect *Stage II* predictions (bottom) are reliable when subjects are in close proximity or overlap, unlike the ORPM formulation (top). The red arrows indicate instances where the latter fails due to similar joints overlapping or being in close proximity, while the two stage approach handles those cases robustly.

Performance on Multi-Person 3D Pose Datasets

The method's accuracy (after *Stage II*) is quantitatively evaluated on the MuPoTS-3D monocular multi-person benchmark data set. Table 6.5(All) compares on all annotated poses in sequences **T1-T20** of the annotated test set, including poses of humans that were not detected by the methods. Table 6.5(Matched), compares only on annotated poses of humans detected by the respective methods.

Table 6.3: Evaluation of 2D keypoint detections of the complete XNect *Stage I* (both 2D and 3D branches trained), with different core networks on a subset of validation frames of MS COCO dataset. Also reported are the forward pass timings of the first stage on different GPUs (K80, TitanX (Pascal)) for an input image of size 512×320 pixels. Also shown is the 2D pose accuracy when using channel-dense supervision of $\{l_{j,k}\}_{j=1}^{J}$ in the 3D branch in place of the proposed channel-sparse supervision (Section 6.3.3).

Core Network	**FP Time K80**	**TitanX**	**AP**	**AP$_{0.5}$**	**AP$_{0.75}$**	**AR**	**AR$_{0.5}$**	**AR$_{0.75}$**
ResNet-34	29.0ms	6.5ms	45.0	72.0	46.1	49.9	74.4	51.6
ResNet-50	39.3ms	10.5ms	46.6	73.0	48.9	51.4	75.4	54.0
SelecSLS	28.6ms	7.4ms	**47.0**	**73.5**	**49.5**	**51.8**	75.6	**54.1**
3D Branch With Channel-Dense $\{l_{j,k}\}_{j=1}^{J}$ Supervision								
SelecSLS	28.6ms	7.4ms	46.8	**73.5**	49.0	51.5	**75.9**	53.8

Table 6.4: Comparison on the single person MPI-INF-3DHP dataset. Top part are methods designed and trained for single-person capture. Bottom part are multi-person methods trained for multi-person capture but evaluated on single-person capture. Metrics used are: 3D percentage of correct keypoints (3DPCK, higher is better), area under the curve (AUC, higher is better) and mean 3D joint position error (MJPE, lower is better). * Indicates that **no** test time augmentation is employed. †Indicates that **no** ground-truth bounding box information is used and the complete image frame is processed.

Method	**3DPCK**	**AUC**	**MJPE**
Location-Map	78.1	42.0	119.2
Location-Map*	75.0	39.2	132.8
Nibali et al. 2019	87.6	48.8	87.6
W. Yang et al. 2018	69.0	32.0	-
Xingyi Zhou et al. 2017	69.2	32.5	-
Pavlakos et al. 2018a	71.9	35.3	-
Dabral et al. 2018	72.3	34.8	116.3
Kanazawa et al. 2018	72.9	36.5	124.2
ORPM (SP)	76.2	38.3	120.5
ORPM (MP)	74.1	36.7	125.1
ORPM (MP)*	72.1	35.1	130.3
XNect(*SelecSLS*)*†	82.8	45.3	98.4

Both tables show that XNect approach achieves comparable accuracy in terms of the 3D percentage of correct keypoints metric (**3DPCK**, higher is better) to LCRNet++ [2019], while being much better than the ORPM formulation and LCRNet [2017]. The faster 'demo' version of LCRNet++ [2019] is less accurate than the results reported here, and runs at $10-12$ fps, while the motion capture system based on the XNect formulation runs at > 30 fps. Note that there is no test-time augmentation or ensembling applied while evaluating on the benchmark, making the reported performance on various benchmarks accurately reflect the real per-frame prediction performance of the proposed system.

Qualitative comparisons to the ORPM formulation in Figure 6.10 show that in scenarios where similar body parts of different individuals overlap, the ORPM pose formulation encounters encoding conflicts, while the XNect formulation successfully handles these cases. XNect pose estimates are comparable to LCRNet++ [2019] quantitatively and qualitatively. However, since LCRNet++ evaluates redundant region proposals, pose estimates from multiple overlapping regions need to be fused. In cases of inter-personal proximity or overlap, the pose proposal fusion step can break,

Table 6.5: Comparison of our per-frame estimates (Stage II) on the MuPoTS-3D benchmark data set. The metric used is 3D percentage of correct keypoints (3DPCK), so higher is better. The data set contains 20 test scenes TS1-TS20. Evaluations are once on all annotated poses (top row - All), and once only on the annotated poses detected by the respective algorithm (bottom row - Matched). The XNect approach achieves better accuracy than the ORPM formulation, and comparable or better accuracy than the previous monocular multi-person 3D methods from the literature (LCRNet Gregory Rogez et al. 2017, LCRNet++ G. Rogez et al. 2019) while having a drastically faster runtime. * Indicates **no** test time augmentation is used.

All	**TS1**	**TS2**	**TS3**	**TS4**	**TS5**	**TS6**	**TS7**	**TS8**	**TS9**	**TS10**	**TS11**	**TS12**	**TS13**	**TS14**	**TS15**	**TS16**	**TS17**	**TS18**	**TS19**	**TS20**	**Total**
LCRNet*	67.7	49.8	53.4	59.1	67.5	22.8	43.7	49.9	31.1	78.1	33.4	33.5	51.6	49.3	56.2	66.5	65.2	62.9	66.1	59.1	53.8
ORPM	81.0	59.9	64.4	62.8	68.0	30.3	65.0	59.2	64.1	83.9	67.2	68.3	60.6	56.5	69.9	79.4	79.6	66.1	64.3	63.5	65.0
LCRNet++*	87.3	61.9	67.9	**74.6**	**78.8**	**48.9**	58.3	59.7	**78.1**	**89.5**	69.2	**73.8**	**66.2**	56.0	**74.1**	**82.1**	78.1	**72.6**	73.1	61.0	**70.6**
XNect*	**88.4**	**65.1**	**68.2**	72.5	76.2	46.2	**65.8**	**64.1**	75.1	82.4	**74.1**	72.4	64.4	**58.8**	73.7	80.4	**84.3**	67.2	**74.3**	**67.8**	70.4

Matched	**TS1**	**TS2**	**TS3**	**TS4**	**TS5**	**TS6**	**TS7**	**TS8**	**TS9**	**TS10**	**TS11**	**TS12**	**TS13**	**TS14**	**TS15**	**TS16**	**TS17**	**TS18**	**TS19**	**TS20**	**Total**
LCRNet*	69.1	67.3	54.6	61.7	74.5	25.2	48.4	63.3	69	78.1	53.8	52.2	60.5	60.9	59.1	70.5	76	70	77.1	81.4	62.4
ORPM	81	64.3	64.6	63.7	73.8	30.3	65.1	60.7	64.1	83.9	71.5	69.6	69	69.6	71.1	82.9	79.6	72.2	76.2	85.9	69.8
LCRNet++*	88	**73.3**	67.9	**74.6**	81.8	**50.1**	60.6	60.8	**78.2**	**89.5**	70.8	**74.4**	**72.8**	64.5	**74.2**	**84.9**	**85.2**	**78.4**	75.8	74.4	74.0
XNect*	**88.4**	70.4	**68.3**	73.6	**82.4**	46.4	**66.1**	**83.4**	75.1	82.4	**76.5**	73.0	72.4	**73.8**	74.0	83.6	84.3	73.9	**85.7**	**90.6**	**75.8**

Table 6.6: Results of Stage II predictions on Human3.6m, evaluated on all camera views of Subject 9 and 11 without alignment to GT. The Stage II network is trained with only Human3.6m. The top part has single person 3D pose methods, while the bottom part shows methods designed for multi-person pose estimation. Mean Per Joint Position Error (MPJPE) in millimeters is the metric used (lower is better). Note that the reported results for Location-Maps, ORPM, and XNect do **not** use any test time augmentation or rigid alignment to ground truth.

Method	Direct	Discuss	Eating	Greet	Phone	Posing	Purch.	Sitting	Sit Down	Smoke	Take Photo	Wait	Walk	Walk Dog	Walk Pair	All
Pavlakos et al. 2017	60.9	67.1	61.8	62.8	67.5	58.8	64.4	79.8	92.9	67.0	72.3	70.0	54.0	71.0	57.6	67.1
Martinez et al. 2017	51.8	56.2	58.1	59.0	69.5	55.2	58.1	74.0	94.6	62.3	78.4	59.1	49.5	65.1	52.4	62.9
Xingyi Zhou et al. 2017	54.8	60.7	58.2	71.4	62.0	53.8	55.6	75.2	111.6	64.1	65.5	66.0	63.2	51.4	55.3	64.9
Location-Map	62.6	78.1	63.4	72.5	88.3	63.1	74.8	106.6	138.7	78.8	93.8	73.9	55.8	82.0	59.6	80.5
Katircioglu et al. 2018	57.8	64.6	59.4	62.8	71.5	57.5	60.4	80.2	104.1	66.3	80.5	61.2	52.5	70.0	60.1	67.3
B. Tekin et al. 2017	85.0	108.8	84.4	98.9	119.4	98.5	93.8	73.8	170.4	85.1	95.7	116.9	62.1	113.7	94.8	100.1
ORPM	58.2	67.3	61.2	65.7	75.8	62.2	64.6	82.0	93.0	68.8	84.5	65.1	57.6	72.0	63.6	69.9
LCRNet+[2019]	50.9	55.9	63.3	56.0	65.1	52.1	51.9	81.1	91.7	64.7	70.7	54.6	44.7	61.1	53.7	61.2
LCRNet++[2019]	55.9	60.0	64.5	56.3	67.4	55.1	55.3	84.8	90.7	67.9	71.8	57.5	47.8	63.3	54.6	63.5
XNect (*SelecSLS*)	50.2	61.9	58.3	58.2	68.8	54.1	61.5	76.8	91.7	63.4	74.6	58.5	48.3	65.3	53.2	63.6

resulting in spurious predictions as seen in Figure 6.11.

Channel-Sparse 3D Pose Encoding Evaluation

As discussed at length in Section 6.3.1, different choices for supervision of $\{l_{j,k}\}_{j=1}^{J}$ have different implications. Here we show that the channel-sparse supervision of the encoding, such that only the local kinematic context is accounted for, performs better than the naïve channel-dense supervision.

The 2D pose accuracy of the *Stage I* network with channel-dense supervision of 3D branch is comparable to that with channel-sparse supervision, as shown in Table 6.3. However, the proposed encoding performs much better across single person and multi-person 3D pose benchmarks.

Table 6.10 shows that channel-sparse encoding significantly outperforms channel-dense encoding, yielding an overall 3DPCK of 82.8 compared to 80.1 3DPCK for the latter. The difference particularly emerges for difficult pose classes such as sitting on a chair or on the floor, where a channel-sparse supervision shows substantial gains. Breakdown by joint types (Tables 6.8, 6.9) reveals that the

Table 6.7: Evaluation of different core network choices with channel-sparse supervision of 3D pose branch of *Stage I*, as well as a comparison to channel-dense supervision on the multi-person 3D pose benchmark MuPoTS-3D. The evaluations are on on all annotated subjects using the 3D percentage of correct keypoints (**3DPCK**) metric, as well as only for predictions that were matched to an annotation. Also shown is the accuracy split for visible and occluded joints.

	3DPCK				% Subjects
	All	**Matched**	**Visible**	**Occluded**	**Matched**
ResNet-34	67.0	72.6	70.4	55.3	92.1
ResNet-50	70.1	75.3	73.7	57.3	**93.0**
SelecSLS	**70.4**	**75.8**	**74.1**	**57.8**	92.8
Channel-Dense $\{l_{j,k}\}_{j=1}^{J}$ Supervision					
SelecSLS	68.1	73.4	71.4	56.3	92.7

Table 6.8: Comparison of limb joint 3D pose accuracy on MPI-INF-3DHP (Single Person) for different core network choices with channel-sparse supervision of 3D pose branch of *Stage I*, as well as a comparison to channel-dense supervision. Metrics used are 3DPCK and AUC (higher is better).

	3DPCK				Total	
	Elbow	**Wrist**	**Knee**	**Ankle**	**3DPCK**	**AUC**
ResNet-34	79.6	61.2	83.0	52.7	79.3	41.8
ResNet-50	**82.4**	61.8	87.1	58.9	82.0	44.1
SelecSLS	81.2	**62.0**	**87.6**	**63.3**	**82.8**	**45.3**
Channel-Dense $\{l_{j,k}\}_{j=1}^{J}$ Supervision						
SelecSLS	79.0	60.2	82.5	59.0	80.1	43.3

Table 6.9: Comparison of limb joint 3D pose accuracy on MuPoTS-3D (Multi Person) for different core network choices with channel-sparse supervision of 3D pose branch of *Stage I*, as well as a comparison to channel-dense supervision. The metric used is 3D Percentage of Correct Keypoints (3DPCK), evaluated with a threshold of 150mm.

	3DPCK					FP Time	
	Elbow	**Wrist**	**Knee**	**Ankle**	**Total**	**K80**	**TitanX**
ResNet-34	63.7	50.5	69.1	37.3	67.0	29.0ms	6.5ms
ResNet-50	65.8	**53.2**	71.0	47.3	70.1	39.3ms	10.5ms
SelecSLS	**66.8**	52.9	**72.2**	**47.6**	**70.4**	28.6ms	7.4ms
Channel-Dense $\{l_{j,k}\}_{j=1}^{J}$ Supervision							
SelecSLS	64.2	51.1	70.1	44.3	68.1	28.6ms	7.4ms

Table 6.10: Evaluation of the impact of the different components from *Stage I* that form the input to *Stage II*. The method is trained for multi-person pose estimation and evaluated on the MPI-INF-3DHP single person 3D pose benchmark. The components evaluated are the 2D pose predictions P_k^{2D}, the body joint confidences C_k, and the set of extracted 3D pose encodings $\{l_{j,k}\}_{j=1}^{J}$. Metrics used are: 3D percentage of correct keypoints (**3DPCK**, higher is better), area under the curve (**AUC**, higher is better) and mean 3D joint position error (**MJPE**, lower is better). Also shown are the results with channel-dense supervision of 3D pose encodings $\{l_{j,k}\}_{j=1}^{J}$, as well as evaluation of *Stage III* output.

	3DPCK			Total	
Stage II Input	**Stand /Walk**	**Sitt.**	**On The Floor**	**3DPCK**	**AUC**
P_k^{2D} (2D Branch Only)	86.4	76.3	44.9	76.0	42.1
P_k^{2D}	79.8	78.4	58.5	75.5	41.3
$P_k^{2D} + C_k$	85.9	79.4	58.7	77.2	42.2
$P_k^{2D} + C_k + \{l_{j,k}\}_{j=1}^{J}$	**88.4**	**85.8**	**70.7**	**82.8**	**45.3**
Channel-Dense $\{l_{j,k}\}_{j=1}^{J}$ Supervision					
$P_k^{2D} + C_k + \{l_{j,k}\}_{j=1}^{J}$	87.0	83.6	61.5	80.1	43.3

channel-sparse supervision is significantly more accurate for limb joints.

Ablation Study of Input to *Stage II*

Variants of *Stage II* network are evaluated on single-person and multi-person benchmarks, taking different subsets of outputs from *Stage I* as input. On the single person benchmark (Table 6.10), using only the 2D pose from the 2D branch as input to Stage II, without having trained the 3D branch for Stage I, results in a 3DPCK of 76.0. When using 2D pose from a network with a 3D branch, trained additionally on MuCo-3DHP dataset, there is a minor performance decrease to 75.5 3DPCK. Though it comes with a performance improvement on challenging pose classes such as 'Sitting' and 'On The Floor' which are under-represented in MSCOCO. Adding other components on top of 2D pose, such as the joint detection confidences C_k, and output features from the 3D branch $\{l_{j,k}\}_{j=1}^{J}$ (as described in Section 6.3.1) leads to consistent improvement as more components are subsequently used as input to *Stage II*. Using joint detection confidences C_k with 2D pose increases the accuracy to 77.2 3DPCK, and incorporating 3D pose features $\{l_{j,k}\}_{j=1}^{J}$ increases the accuracy to 82.8 3DPCK, and both lead to improvements in AUC and MPJPE as well as improvements for both simpler poses such as upright 'Standing/walking' as well as more difficult poses such as 'Sitting' and 'On the Floor'

Similarly for multi person 3D pose evaluation on the proposed MuPoTS-3D benchmark in Table 6.11, introduction of additional components as input to *Stage II* leads to improvement on the overall 3DPCK metric, for both visible and occluded joints. Each subsequently introduced component leads to an overall ≈5 3DPCK improvement. The most significant impact of the intermediate 3D pose features $\{l_{j,k}\}_{j=1}^{J}$ is on pose accuracy for occluded joints.

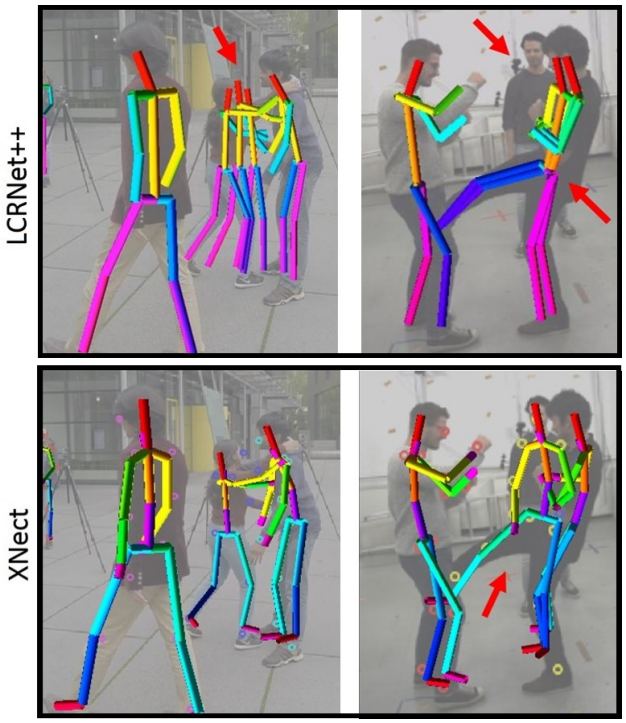

Figure 6.11: XNect *Stage II* pose estimates (bottom) are qualitatively and quantitatively comparable to LCRNet++ (G. Rogez et al. 2019) (top). LCRNet++ occasionally predicts multiple skeletons for one individual, particularly when people are in close proximity, or does not detect occluded individuals, as marked with arrows. XNect predictions avoid such issues, though they may exhibit alternative modes of failure, discussed in Chapter 8.

Table 6.11: Evaluation of choices for input to the 2nd stage on MuPoTS. The metric used is 3D percentage of correct keypoints (**PCK**), so higher is better. The data set contains 20 test scenes **T1-T20**. We evaluate once on all annotated poses (top row - **All**), Evaluation of the impact of the different components from *Stage I* that form the input to *Stage II*, evaluated on the multi person 3D pose benchmark MuPoTS-3D D. Mehta et al. 2018. We evaluate on all annotated subjects using the 3D percentage of correct keypoints (**3DPCK**) metric, also showing the accuracy split for visible and occluded joints. The components evaluated are the 2D pose predictions P_k^{2D}, the body joint confidences C_k, and the set of extracted 3D features $\{l_{j,k}\}_{j=1}^{J}$.

Stage II Input	**3DPCK**			
	All	**Matched**	**Visible**	**Occluded**
P_k^{2D}	59.3	63.9	61.6	50.0
$P_k^{2D} + C_k$	64.1	69.1	67.6	51.7
$P_k^{2D} + C_k + \{l_{j,k}\}_{j=1}^{J}$	70.4	75.8	74.1	57.8

6.4 Conclusion

This chapter builds upon the location-map formulation from the previous chapter, to develop formulations that extend beyond unoccluded single-person scenarios to true in-the-wild scenes which include multi-person scenarios as well as occlusions. To be able to handle general scenes with multiple subjects as well as foreground objects and clutter in the scene, a degree of redundancy is built into the representations (occlusion-robust-pose-map, XNect). In addition to the receptive field centered at the body joint in question predicting the 3D pose representation for that joint, receptive fields centered at a few other joints further up in the kinematic chain also predict the pose representation for that joint. To be able to jointly encode the pose representations for all the subjects in the scene, who may be in close proximity or overlapping, the degree of redundancy has to be restricted to avoid encoding conflicts. Hence, the XNect formulation restricts this redundancy only to the joints directly connected together by a linkage (bone) in the kinematic tree. Pose inference for occluded joints, as well as body joints in close proximity to the same body joint of other individuals, is then done by a small fully connected network which operates independently per detected individual, and incorporates the full body context to correct and fill in the missing joints.

The inference cost of the proposed multi-person formulations does not scale up in direct proportion to the number of subjects in the scene, which is a key factor in enabling real-time multi-person motion capture. The next chapter discusses ways of speeding up the core (or backbone) convolutional network used by the developed formulation, and proposes the *SelecSLS* convolutional neural network design to enable $\approx 1.4\times$ faster inference than ResNet-50 without compromising the accuracy.

Chapter 8 makes use of the multi-person 3D pose formulation developed in this chapter to develop a real-time multi-person motion capture system, which produces temporally smooth joint angle estimates as well as camera relative location estimates of the subjects in the scene.

Chapter 7

A Fast Convolutional Core Network Architecture

Thus far in the thesis the focus has been on the pose formulation and training schema, to allow fast, accurate and in-the-wild pose estimation. Beyond the pose formulation's role in inference time, various steps can be taken to speed up the forward pass of the image through the backbone/core Convolutional Neural Network (CNN). Reducing the inference time of the core network without compromising on accuracy is particularly crucial in the case of the formulation proposed for multi-person pose estimation in Chapter 6. It makes it possible for the run time to not scale linearly with the number of subjects in the scene, but it still requires the complete input frame to be processed at every time step. Using popular network architectures such as ResNet-50 (K. He et al. 2016a) as the backbone would only allow real-time inference on top-end GPUs, and alternatives such as MobileNetV2 (Sandler et al. 2018) compromise accuracy for inference speed improvement. This chapter examines inference speed up through feature pruning, as well as through a novel convolutional network design that allows speedup without sacrificing accuracy.

In This Chapter

- Presentation of some surprising results on implicit filter level sparsity in Convolutional Neural Networks (Section 7.1)
- Discussion of prunability of ResNet type networks which use additive skip connectivity (Section 7.2.2)
- Design of a new Convolutional Neural Network architecture termed *SelecSLS*, that matches the accuracy of ResNet-50, but with 1.2-1.4× the inference speed (Section 7.2.3)

The content of this chapter is based on D. Mehta et al. 2019 and D. Mehta et al. 2020.

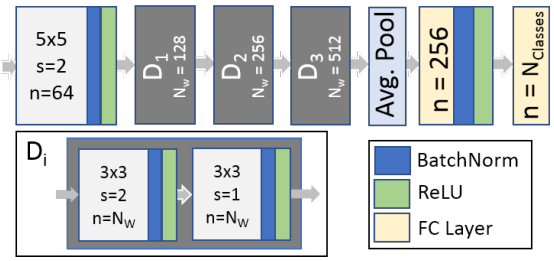

Figure 7.1: BasicNet: Structure of the basic convolution network studied in the following sections. The convolution layers are referred to as C1-7.

Table 7.1: Convolutional filter sparsity for BasicNet with leaky ReLU with different negative slopes, trained on CIFAR-100 with Adam and L2 regularization (1e-4). Average of 3 runs.

Neg. Slope	Train Loss	Val Loss	Val Err.	% Spar. by γ
0.00	0.10	1.98	36.6	46
0.01	0.10	1.99	36.8	41
0.10	0.14	2.01	37.2	43

Table 7.2: Convolutional filter sparsity for BasicNet trained on TinyImageNet, with different mini-batch sizes.

Batch Size	Train Loss	Val Loss	Top 1 Val Err.	Top 5 Val Err.	% Spar. by γ
20	1.05	2.13	47.7	22.8	63
40	0.16	2.96	48.4	24.7	48
120	0.01	2.48	48.8	27.4	26

Table 7.3: Convolutional filter sparsity in *BasicNet* trained on CIFAR10/100 for different combinations of regularization and gradient descent methods. Shown are the % of non-useful / inactive convolution filters, as measured by activation over training corpus (max act. $< 10^{-12}$) and by the learned BatchNorm scale ($|\gamma| < 10^{-03}$), averaged over 3 runs. The lowest test error per optimizer is highlighted, and sparsity (green) or lack of sparsity (red) for the best and near best configurations indicated via text color. L2: L2 regularization, WD: Weight decay (adjusted with the same scaling schedule as the learning rate schedule).

		CIFAR10			CIFAR100		
		% Sparsity		Test	% Sparsity		Test
	L2	by Act	by γ	Error	by Act	by γ	Error
SGD	1e-03	27	27	21.8	23	23	47.1
	1e-04	0	0	11.8	0	0	37.4
	1e-05	0	0	10.5	0	0	39.0
	0	0	0	11.3	0	0	40.1
Adam	2e-03	88	86	14.7	82	81	42.7
	1e-04	71	70	10.5	47	47	36.6
	1e-05	48	48	10.7	5	5	40.6
	0	3	0	11.0	0	0	40.3
Adadelta	5e-04	82	82	13.6	61	61	39.1
	2e-04	40	40	11.3	3	3	35.4
	1e-04	1	1	10.2	1	1	35.9
Adagrad	2e-02	75	75	11.3	88	88	63.3
	1e-02	65	65	11.2	59	59	37.2
	5e-03	56	56	11.3	24	25	35.9
AMSGrad	1e-02	93	93	20.9	95	95	71.9
	1e-04	51	47	9.9	20	13	35.6
	1e-06	0	0	11.2	0	0	40.2
Adamax	1e-02	75	90	16.4	74	87	51.8
	1e-04	49	50	10.1	10	10	39.3
	1e-06	4	4	11.3	0	0	39.8
RMSProp	1e-02	95	95	26.9	97	97	78.6
	1e-04	72	72	10.4	48	48	36.3
	1e-06	29	29	10.9	0	0	40.6
		CIFAR10			CIFAR100		
		% Sparsity		Test	% Sparsity		Test
	WD	by Act	by γ	Error	by Act	by γ	Error
SGD	1e-03	27	27	21.6	23	23	47.6
	2e-04	0	0	13.3	0	0	39.4
	1e-04	0	0	12.4	0	0	37.7
Adam	5e-04	81	81	18.1	59	59	43.3
	2e-04	60	60	13.4	16	16	37.3
	1e-04	40	40	11.2	3	3	36.2

7.1 Implicit Pruning in Convolutional Neural Networks

As discussed in Chapter 2, one of the avenues of speeding up convolutional neural network inference is to apply structured pruning to different layers, removing redundant on non-useful filters.

Among the various explicit filter level sparsification heuristics and approaches (H. Hu et al. 2016; H. Li et al. 2017; Z. Liu et al. 2017; Molchanov et al. 2017; Mozer et al. 1989; Srinivas et al. 2016; Theis et al. 2017; J. Ye et al. 2018), some (Z. Liu et al. 2017; J. Ye et al. 2018) make use of the learned scale parameter γ associated with Batch Normalization (Ioffe et al. 2015) for enforcing sparsity on the filters. J. Ye et al. 2018 argue that Batch Normalization makes feature importance less susceptible to scaling reparameterization, and the learned scale parameters, γ, can be used as indicators of feature importance.

Different from explicit sparsification, the results presented here show that convolutional neural networks which employ Batch Normalization and ReLU activation, when trained with adaptive gradient descent flavours, implicitly prune certain features, causing them to not activate for any input image. Importantly, the sparsity emerges in the presence of regularizers such as L2 and weight decay (WD) which are in general understood to be non sparsity inducing, and the sparsity vanishes when regularization is removed.

The following observations are made experimentally:

- The sparsity is much higher when using adaptive flavors of SGD such as ADAM (Kingma et al. 2014), AdaDelta (Zeiler 2012), AdaGrad Duchi et al. 2011 etc. as compared to the extent of sparsity seen when using momentum-SGD (mSGD).
- Adaptive flavors see higher sparsity with L2 regularization than with WD. No sparsity emerges in the absence of regularization.
- In addition to the regularizers, the extent of the emergent sparsity is also influenced by hyperparameters seemingly unrelated to regularization. The sparsity decreases with increasing mini-batch size, decreasing network size and increasing task difficulty.
- The primary hypothesis put forward is that selective features[1] see a disproportionately higher amount of regularization than non-selective ones. This consistently explains how parameters such as mini-batch size, network size, and task difficulty indirectly impact sparsity by affecting feature selectivity.
- A secondary hypothesis to explain the higher sparsity observed with adaptive methods is that Adam (and possibly other) adaptive approaches learn more selective features. Though there is evidence of highly selective features with Adam, this requires further study.
- Synthetic experiments show that the interaction of L2 regularizer with the update equation in adaptive methods causes stronger regularization than WD. This can explain the discrepancy in sparsity between L2 and WD.

Quantifying Feature Sparsity: Feature sparsity can be measured by per-feature activation and by per-feature scale. For sparsity by activation, the absolute activations for each feature are max pooled over the entire feature plane. If the value is less than 10^{-12} over the entire *training* corpus, the feature is inactive. For sparsity by scale, the scale γ of the learned affine transform in the Batch Norm layer is considered. Batch normalization uses additional learned scale γ and bias β that casts each normalized convolution output $\hat{x}_i$ to $y_i = \gamma\hat{x}_i + \beta$. A feature is considered inactive if $|\gamma|$ for the feature is less than 10^{-3}. Explicitly zeroing the features thus marked inactive does not affect the test

[1] Feature selectivity is the fraction of training exemplars for which a feature produces max activation less than some threshold.

error, which ensures the validity of the chosen thresholds. The thresholds chosen are purposefully conservative, and comparable levels of sparsity are observed for a higher feature activation threshold of 10^{-4}, and a higher $|\gamma|$ threshold of 10^{-2}.

The key experimental results presented in the following section are in the context of image classification, primarily to be able to compare with existing literature on neural network sparsification and pruning (H. Li et al. 2017; Molchanov et al. 2017; J. Park et al. 2017; Wen et al. 2016), as well as other related experimental results on feature selectivity Morcos et al. 2018; B. Zhou et al. 2018. Refer to Appendix D for detailed experiments and results. The mechanism of the emergence of implicit sparsity that is uncovered is not limited to image classification, and it also manifests for other problems, such as pose estimation. See Figure D.4.

7.1.1 Observing Filter Sparsity

Preliminary Experiments: A 7-layer convolutional network with 2 fully connected layers, as shown in Figure 7.1, is used for much of the experiments that follow. This network architecture would be referred to as *BasicNet* in the following sections. The network structure is inspired by VGG (Simonyan et al. 2014), but is more compact. The convolutional layers of the network would be referred to as C1-7. For the basic experiments on CIFAR-10/100, a variety of gradient descent approaches are examined, with a mini-batch size of 40, with a method specific base learning rate for 250 epochs which is scaled down by 10 for an additional 75 epochs. CIFAR-10/100 (Krizhevsky et al. 2009) are commonly used datasets for image classification, comprised of 10 and 100 classes respectively. The base learning rates and other hyperparameters are kept at default values, which are as follows: Adam (1e-3, β_1=0.9, β_2=0.99, ε=1e-8), Adadelta (1.0, ρ=0.9, ε=1e-6), SGD (0.1, mom.=0.9), Adagrad (1e-2), AMSGrad (1e-3), AdaMax (2e-3), RMSProp (1e-3).

The objective is to study the effect of varying the amount and type of regularization[2] on the extent of sparsity and test error, with various optimization approaches. The results in Table 7.3 show that significant convolutional filter sparsity emerges with adaptive gradient descent methods when combined with L2 regularization. The extent of sparsity is reduced when using Weight Decay instead, and absent entirely in the case of SGD with moderate levels of regularization. Table 7.1 shows that using leaky ReLU does not prevent sparsification.

Sparsity Manifests Across Network Architectures and Datasets: The emergence of sparsity is not an isolated phenomenon specific to CIFAR-10/100 and *BasicNet*. In addition to CIFAR10 and CIFAR100, other network architecture and other datasets are examined as well. TinyImageNet (Tiny ImageNet n.d.) is a 200 class subset of ImageNet (Deng et al. 2009) with images resized to 64×64 pixels. ImageNet (ibid.) is a 1000 class dataset for image classification, with images resized to 256×256 pixels. Tables 7.4, 7.6, and 7.7 show that sparsity manifests in VGG-11/16 [2014], and ResNet-50 [2016] on ImageNet and Tiny-ImageNet. ResNet-50 shows a significantly higher overall filter sparsity than non-residual VGG networks. Beyond classification, sparsity also manifests for regression tasks such as the pose regression tasks the thesis examines.

Sparsity Increases with Decreasing Mini-Batch Size: It can be seen in Tables 7.5, 7.2, 7.4, 7.6, and 7.7 that decreasing the mini-batch size (while maintaining the same number of iterations) leads to increased sparsity across network architectures and datasets.

[2]Note that L2 regularization and weight decay are distinct. See Loshchilov et al. 2017 for a detailed discussion.

7.1.2 Explaining Filter Sparsity

Feature Selectivity Hypothesis: From Figure 7.3 the differences between the nature of features learned by Adam and SGD become clearer. For zero mean, unit variance BatchNorm outputs $\{\hat{x}_i\}_{i=1}^{N}$ of a particular convolutional kernel, where N is the size of the training corpus, due to the use of ReLU, a gradient is only seen for those datapoints for which $\hat{x}_i > -\beta/\gamma$. Both SGD and Adam (L2: 1e-5) learn positive γs for layer C6, however βs are negative for Adam, while for SGD some of the biases are positive. This implies that all features learned for Adam (L2: 1e-5) in this layer activate for less than or equal to half the activations from the training corpus, while SGD has a significant number of features activate for more than half of the training corpus. That means, Adam learns more selective features in this layer. Features which activate only for a small subset of the training corpus, and consequently see gradient updates from the main objective less frequently, continue to be acted upon by the regularizer. If the regularization is strong enough (Adam with L2: 1e-4 in Fig. 7.3), or the gradient updates infrequent enough (feature too selective), the feature may be pruned away entirely. The propensity of later layers to learn more selective features with Adam would explain the higher degree of sparsity seen for later layers as compared to SGD. Understanding the reasons for emergence of higher feature selectivity in Adam than SGD, and verifying if other adaptive gradient descent flavours also exhibit higher feature selectivity remains open for future investigation.

Quantifying Feature Selectivity: Similar to feature sparsity by activation, the feature's absolute activations over the entire feature plane are max pooled. For a particular feature, these pooled activations over the entire training corpus are considered in quantifying feature selectivity, and the pooled activations for each filter are normalized by the max of the filter's pooled activations over the entire training corpus. The percentage of the training corpus for which this normalized pooled value exceeds a threshold of 10^{-3} is then used to quantify selectivity. This is termed as the feature's *universality*. A feature's selectivity is then defined as 100-*universality*. Unlike the selectivity metrics employed in literature (Morcos et al. 2018), this is class agnostic, and provides preliminary quantitative evidence that Adam (and perhaps other adaptive gradient descent methods) learn more selective features than (m)SGD, which consequently see a higher relative degree of regularization. In Figure 7.2 note the distinct shift towards less selective features in *BasicNet* with increasing task difficulty, as well as fewer selective features being pruned with larger batch sizes than with smaller batch sizes.

Table 7.4: Sparsity by γ on VGG-16, trained on TinyImageNet, and on ImageNet. Also shown are the pre- and post-pruning top-1/top-5 single crop validation errors. Pruning using $|\gamma| < 10^{-3}$ criteria.

	# Conv	Pre-pruning		Post-pruning	
TinyImageNet	Feat. Pruned	top1	top5	top1	top5
L2: 1e-4, B: 20	3016 (71%)	45.1	21.4	45.1	21.4
L2: 1e-4, B: 40	2571 (61%)	46.7	24.4	46.7	24.4
ImageNet					
L2: 1e-4, B: 40	292	29.93	10.41	29.91	10.41

Interaction of Regularization with Optimizer Updates: Figure 7.4 examines the behaviour of L2 regularization in the low gradient regime for different optimizers through synthetic experiments, and shows that coupling of L2 regularization with certain adaptive gradient update equations yields a faster decay than weight decay, or L2 regularization with SGD, even for smaller regularizer values. This is an additional source of regularization disparity between parameters which see frequent updates and those which don't see frequent updates or see lower magnitude gradients. It manifests

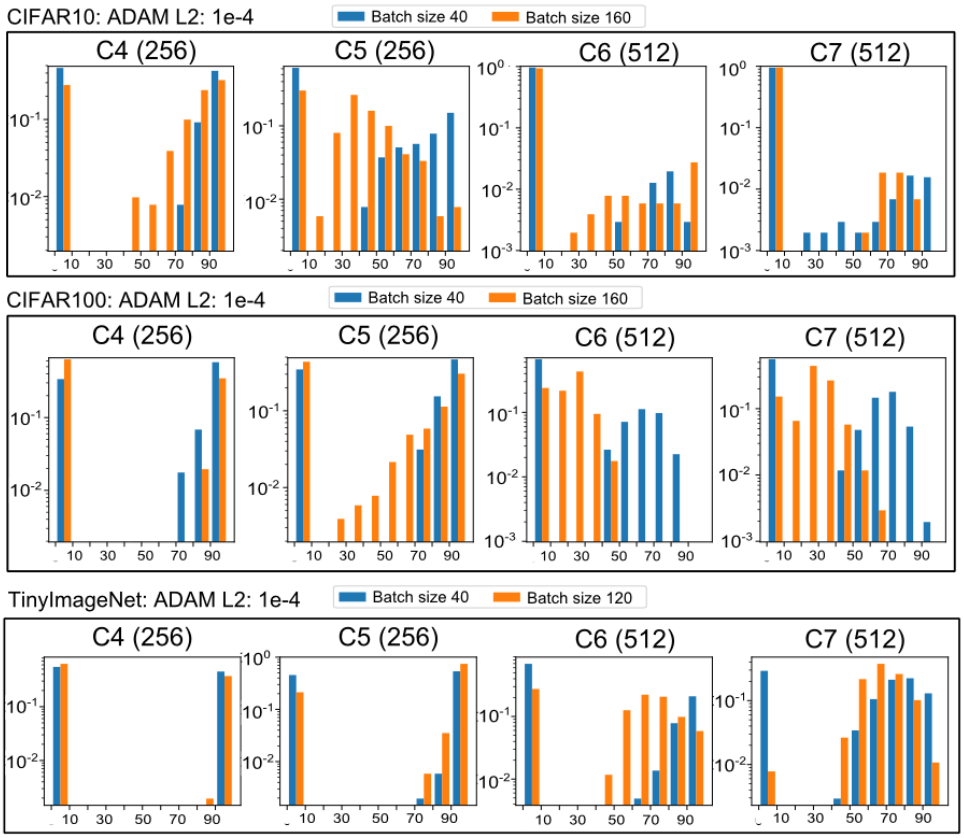

Figure 7.2: Feature Selectivity For Different Mini-Batch Sizes for Different Datasets Feature universality (1 - selectivity) plotted for layers C4-C7 of *BasicNet* for CIFAR10, CIFAR100 and TinyImagenet. Batch sizes of 40/160 considered for CIFAR, and 40/120 for TinyImagenet.

Table 7.5: BasicNet sparsity variation on CIFAR10/100 trained with Adam and L2 regularization.

		CIFAR 10				CIFAR 100			
	Batch Size	Train Loss	Test Loss	Test Err	%Spar. by γ	Train Loss	Test Loss	Test Err	%Spar. by γ
L2: 1e-4	20	0.17	0.36	11.1	70	0.69	1.39	35.2	57
	40	0.06	0.43	10.5	70	0.10	1.98	36.6	46
	80	0.02	0.50	10.1	66	0.02	2.21	41.1	35
	160	0.01	0.55	10.6	61	0.01	2.32	44.3	29

Table 7.6: Effect of different mini-batch sizes on sparsity (by γ) in VGG-11, trained on ImageNet. Same network structure employed as Z. Liu et al. 2017. * indicates finetuning after pruning

	# Conv Feat. Pruned	Pre-pruning top1	Pre-pruning top5	Post-pruning top1	Post-pruning top5
Adam, L2: 1e-4, B: 90	71	30.50	10.65	30.47	10.64
Adam, L2: 1e-4, B: 60	140	31.76	11.53	31.73	11.51
Z. Liu et al. 2017	85	29.16		31.38*	-

for certain adaptive gradient descent approaches.

Task 'Difficulty' Dependence: As per the hypothesis developed thus far, as the task becomes

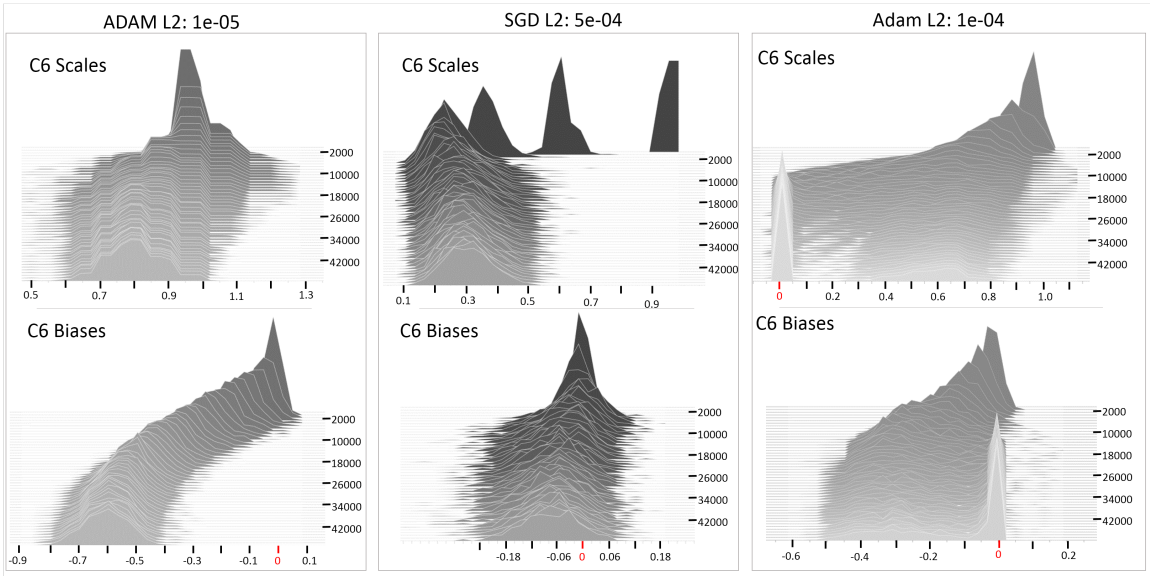

Figure 7.3: Emergence of Feature Selectivity with Adam The evolution of the learned scales (γ, top row) and biases (β, bottom row) for layer C6 of *BasicNet* for Adam and SGD as training progresses. Adam has distinctly negative biases, while SGD sees both positive and negative biases. For positive scale values, as seen for both Adam and SGD, this translates to greater feature selectivity in the case of Adam, which translates to a higher degree of sparsification when stronger regularization is used.

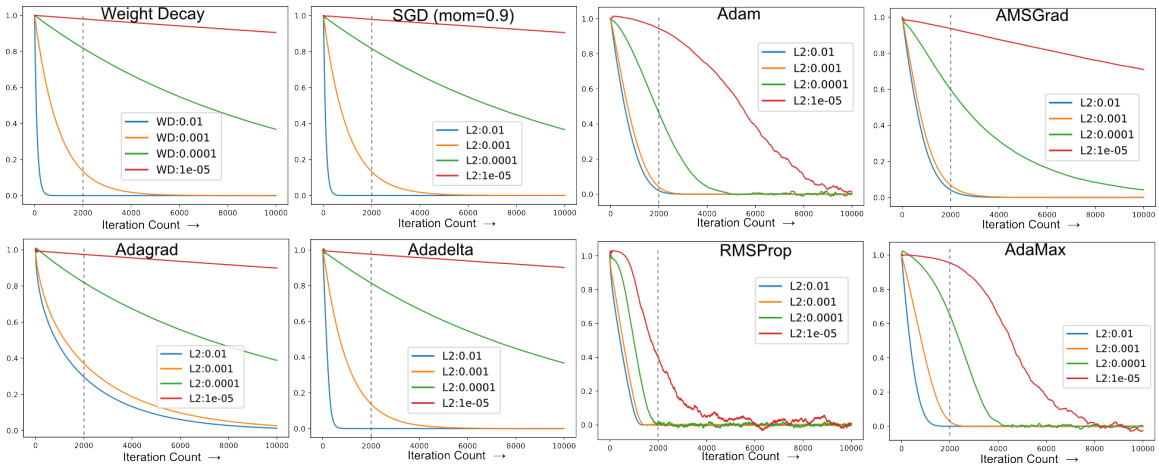

Figure 7.4: The action of regularization on a scalar value for a range of regularization values in the presence of simulated low gradients drawn from a mean=0, std=10^{-5} normal distribution. The gradients for the first 100 iterations are drawn from a mean=0, std=10^{-3} normal distribution to emulate a transition into low gradient regime rather than directly starting in the low gradient regime. The scalar is initialized with a value of 1. The learning rates are as follows: SGD(momentum=0.9,lr=0.1), ADAM(1e-3), AMSGrad(1e-3), Adagrad(1e-2), Adadelta(1.0), RMSProp(1e-3), AdaMax(2e-3). The action of the regularizer in low gradient regime is only one of the factors influencing sparsity. Different gradient descent flavours promote different levels of feature selectivity, which dictates the fraction of features that fall in the low gradient regime. Further, the optimizer and the mini-batch size affect together affect the duration different features spend in low gradient regime.

Table 7.7: Convolutional filter sparsity for different levels of ResNet-50 on ImageNet, with different batch sizes, using Adam and L2 regularization (1e-4).

Batch	Train	Test	Top 1	Top 5	% Sparsity by γ					
Size	Loss	Loss	Val Err.	Val Err.	conv1	res2	res3	res4	res5	Total
32	1.3	1.1	27.7	9.2	0	0	1	17	46	26
64	1.0	1.0	25.2	7.7	0	0	1	3	42	19

more difficult, for a given network capacity, we expect the fraction of features pruned to decrease corresponding to a decrease in selectivity of the learned features (B. Zhou et al. 2018). Though the task difficulty cannot be cleanly decoupled from the number of classes, a synthetic experiment is devised to closely examine task difficulty. Grayscale renderings of 30 object classes from ObjectNet3D (Xiang et al. 2016) are used, and 2 identical sets of $\approx 50k$ 64×64 pixel renderings constructed. One set has a clean background (BG) and the other has a cluttered BG. *BasicNet* is trained with a mini-batch size of 40 on the two datasets, and as expected there is a much higher sparsity (70%) with the clean BG set than with the more difficult cluttered set (57%). See Appendix D for representative images and a list of the object classes selected.

7.1.3 Implications of the Findings Regarding the Emergent Sparsity

Effect of L2 regularization vs. Weight Decay for Adam Prior work (Loshchilov et al. 2017) has indicated that Adam with L2 regularization leads to parameters with frequent and/or large magnitude gradients from the main objective being regularized less than the ones which see infrequent and/or small magnitude gradients. Though weight decay is proposed as a supposed fix, the findings in this chapter show that there are rather two different aspects to consider. The first is the disparity in effective regularization due to the frequency of updates. Parameters which update less frequently would see more regularization steps per actual update than those which are updated more frequently. This disparity would persist even with weight decay due to Adam's propensity for learning more selective features, as detailed in the preceding section. The second aspect is the additional disparity in regularization for features which see low/infrequent gradient, due to the coupling of L2 regularization with Adam.

Attributes of Generalizable Neural Network Features: Dinh et al. 2017 show that the geometry of minima is not invariant to reparameterization, and thus the flatness of the minima may not be indicative of generalization performance (Keskar et al. 2017a), or may require other metrics which are invariant to reparameterization. Morcos et al. 2018 suggest based on extensive experimental evaluation that good generalization ability is linked to reduced selectivity of learned features. They further suggest that individual selective units do not play a strong role in the overall performance on the task as compared to the less selective ones. They connect the ablation of selective features to the heuristics employed in neural network feature pruning literature which prune features whose removal does not impact the overall accuracy significantly (H. Li et al. 2017; Molchanov et al. 2017). The findings of B. Zhou et al. 2018 concur regarding the link between emergence of feature selectivity and poor generalization performance. They further show that ablation of class specific features does not influence the overall accuracy significantly, however the specific class may suffer significantly. The findings in this chapter show that the emergence of selective features in Adam, and the increased propensity for pruning the said selective features when using L2 regularization presents a direct tradeoff between generalization performance and network capacity which practitioners using Adam must be aware of.

Observations on Adaptive Gradient Descent Several works have noted the poorer generalization performance of adaptive gradient descent approaches over SGD. Keskar et al. 2017b propose to leverage the faster initial convergence of ADAM and the better generalization performance of SGD, by switching from ADAM to SGD while training. Reddi et al. 2018 point out that exponential moving average of past squared gradients, which is used for all adaptive gradient approaches, is problematic for convergence, particularly with features which see infrequent updates. This short term memory is likely the cause of accelerated pruning of selective features seen for Adam in Figure 7.4(and other adaptive gradient approaches), and the extent of sparsity observed would be expected to go down with AMSGrad (ibid.) which tracks the long term history of squared gradients.

Reducing the Extent of Emergent Sparsity The root cause of sparsity is revealed to be a combination of the propensity of Adam (and other adaptive gradient descent methods) to learn more selective features, a disproportionately higher degree of effective regularization seen by selective features in a mini-batch training setup, and the added acceleration of L2 regularization in a low gradient regime. This mechanism remains in play even when supposed fixes such as Leaky ReLU are used. With *BasicNet* on CIFAR100, Leaky ReLU with a negative slope of 0.01 only marginally reduces the extent of sparsity in the case of Adam with L2: 10^{-4} (41% feature sparsity for 36.8% test error, vs. 47% feature sparsity for 36.6% test error with ReLU), but does not completely remove it. Beyond the hyperparameters discussed in the preceding sections, other hyperparameters, such as initial values of the affine BatchNorm parameters and their associated learning rates also impact the extent of sparsity, and can be exploited to reduce the extent of emergent sparsity. For further discussion, refer to Appendix D.3.

Implicit Sparsity as a Possible Tool for Neural Network Speed-up The emergent sparsity is not strictly harmful, as it targets selective features, which have been shown by Morcos et al. 2018 to be indicative of overfitting. Adam with L2 regularization can also be an attractive alternative to explicit network slimming approaches for speeding up test time performance of convolutional neural networks, without requiring any tooling changes to the traditional neural network training pipeline supported by the existing frameworks. Though it is not a complete replacement, as explicit sparsification approaches have certain additional desirable properties such as layer-computation-cost aware sparsification, or being able to successively build multiple compact models out of a single round of training like in J. Ye et al. 2018. However, this opens up a line of future work in investigating ways of incorporating a similar level of fine grained control over implicit sparsity as is afforded by explicit sparsification approaches.

In conclusion, awareness and a better understanding of the mechanism of sparsification would prompt attempts at better solutions to address the harmful aspects of the emergent sparsity, while retaining the useful aspects.

7.2 SelecSLS Net: A Fast and Accurate Pose Inference CNN

7.2.1 Convolutional Network Designs

ResNet [2016] and derivatives (Xie et al. 2017) incorporate explicit information flowing from earlier to later feature layers in the network through summation-skip connections. This permits training of deeper and more powerful networks. Many architectures based on this concept have been proposed,

such as Inception (Szegedy et al. 2017) and ResNext (Xie et al. 2017).

Because increased depth and performance comes at the price of higher computation times during inference, specialized architectures for faster test time computation were proposed, such as AmoebaNet (Real et al. 2018), Mobilenet (Sandler et al. 2018), ESPNet (S. Mehta et al. 2018), ERFNet (Romera et al. 2018). These are however not suited for application to the 3D pose estimation system developed in this thesis because: Many architectures with depthwise convolutions are optimized for inference on specific edge devices (Sandler et al. 2018), and lose accuracy in lieu of speed. Increasing the width or depth of these networks to bring the accuracy closer to that of vanilla ResNets results in GPU runtimes comparable to typical ResNet architectures. In addition to significant losses in accuracy to increase inference speed, ESPNet (S. Mehta et al. 2018) produces non-smooth output maps with grid artifacts due to the use of dilated convolutions. These artifacts impair part association performance in our pose estimation setting. DenseNet (G. Huang et al. 2017) uses full dense concatenation-skip connectivity, which results in a parameter efficient network but is slow due to the associated cost of the enormous number of concatenation operations. Table 7.8 shows the performance of some common network architectures on single-person 2D pose estimation, and striking the right balance on the speed-accuracy tradeoff ResNet-34 and ResNet-50 emerge as the appropriate baselines for further experiments.

7.2.2 ResNet-50 and Filter Pruning

ResNet-50 [2016] as core network architecture is a good compromise between performance and speed for many problems beyond image classification, and has been employed for other multi-person pose estimation methods such as G. Rogez et al. 2019. However, processing the complete input frame on anything but the top-end GPUs with a ResNet-50 core, along with the other components of the system which are discussed in Chapter 8, would not reach real-time performance of > 25 FPS. The 'demo' version of LCRNet++ (ibid.) uses ResNet-50 and only works at $10-12$ FPS on a GTX 1050 for 512×320 pixel input. The forward pass time on a TitanX (Pascal) GPU is 16 ms, while on a K80 GPU it takes $>$100 ms.

Explicit or implicit pruning approaches, such as the ones discussed in the previous section can speed up ResNet-50 inference time by about $1.3\times$, but the inference speed gains come at the cost of reduced accuracy. Reference models (Paszke et al. 2017) for ResNet-50 on ImageNet [2009] report a Top-1 accuracy of 76.15% and a Top-5 accuracy of 92.87%. Explicit pruning approaches showing a $1.2-1.4\times$ inference speed up yield significantly lower Top-1 accuracies of 72.04% (Luo et al. 2017), 75% (R. Yu et al. 2018), 74.61% (Y. He et al. 2018), and 74.95% (Zhuang et al. 2018). Table 7.7 shows that implicit sparsity makes a similar tradeoff between accuracy and extent of sparsity, with a Top-1 accuracy of 74.8% with a filter sparsity of 19%.

Hence, a new network architecture is developed in the next section, with the objective of achieving a comparable speed-up to post-pruning ResNet-50, while retaining the accuracy of un-pruned ResNet-50. The proposed design avoids the artifacts and accuracy deficit of ESPNet, and eliminates the memory and speed bottlenecks associated with DenseNet, by relying on a selective use of skip connectivity.

7.2.3 SelecSLS Net: Selective Short and Long Range Skip Connections

The key insight behind the design of the new network architecture is to use short range and long range concatenation-skip connections in a selective way instead of additive-skip connections. Additive-skip

Table 7.8: Evaluation of baseline architecture choices for the backbone convolutional network for 3D pose estimation. Here the choices are evaluated on the pre-requisite single person 2D pose estimation task on LSP Johnson et al. 2010 test set. The core network architectures are jointly trained on MPI Mykhaylo Andriluka et al. 2014 and LSP Johnson et al. 2010, 2011 single person 2D pose datasets. The timings are evaluated on an NVIDIA K80 GPU, with 320×320 pixel input, using *Caffe* 2018 with optimized depthwise convolution implementation.

Core Network	PCK	FP Time (K80)
MobileNetV2 1.0x [2018]	85	13.8ms
MobileNetV2 1.3x [2018]	86	16.4ms
Xception [2017]	81	36.6ms
InceptionV3 [2016]	88	25.7ms
ResNet-34 [2016]	89	19.4ms
ResNet-50 [2016]	89	24.7ms

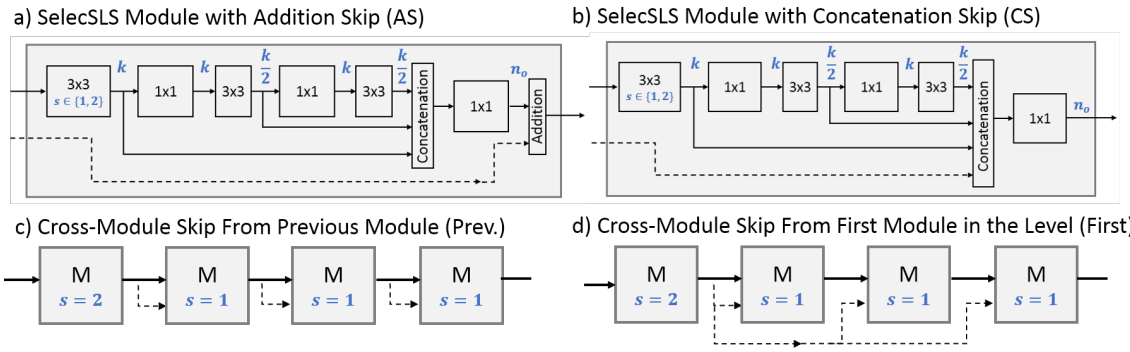

Figure 7.5: Variants of *SelecSLS* module design (a) and (b). Both share a common design comprised of interleaved 1×1 and 3×3 convolutions, with different ways of handling cross-module skip connections internally: (a) as additive-skip connections, or (b) as concatenative-skip connections. The cross module skip connections can themselves come either from the previous module (c) or from the first module which outputs features at a particular spatial resolution (d). In addition to the different skip connectivity choices, the design is parameterized by module stride (s), the number of intermediate features (k), and the number of module ouputs n_o.

gets element-wise added to the features at the point of incorporation of the skip connection, whereas concatenative-skip connections get concatenated with the features in the channel-dimension.

Networks with dense concatenation-skip connections such as DenseNet [2017] are more parameter efficient than typical network architectures with addition-skip connections such as ResNet [2016], and designed for better information flow across layers. However, concatenation-skip connections tend to consume much more memory, and the substantial cost of concatenation operations is usually not accounted for in FLOPs computations, making the reported FLOPs not reflective of actual speeds on consumer hardware using off-the-shelf libraries (Bianco et al. 2018). In SelecSLS Net, the selective use of concatenation-skip connectivity promotes information flow through the network, without the exorbitant memory and compute cost associated with a full dense connectivity. The network shows comparable accuracy to ResNet-50, with a substantially lower inference time across single person and multi-person 2D and 3D pose benchmarks.

***SelecSLS* Module**: The network is comprised of *SelecSLS* modules, with intra-module short-range skip connectivity and cross-module longer-range skip connectivity, for the latter of which different architectural variants are explored. The module design is as shown in Figure 7.5 (a) and (b). All

Table 7.9: *SelecSLS Net* Architecture: The table shows the network levels, overall number of modules, number of intermediate features k, and the spatial resolution of features of the network designs evaluated in Section 7.2.4. The design choices evaluated are the type of module (additive skip *AS* vs concatenation skip *CS*), the type of cross module skip connectivity (From previous module (*Prev*) or first module (*First* in the level), and the scheme for the number of outputs of modules n_o ((B)ase or (W)ide).

Level	**Output Resolution**	***SelecSLS* Module**	**Stride** s	k	**Cross-Module Skip Conn.**	n_o (B)	n_o (W)
L0	w/2 x h/2	Conv. 3x3	2	-	-	32	32
L1	w/4 x h/4	CS/AS	2	64	No	64	64
	w/4 x h/4	CS/AS	1	64	Prev/First	64	128
L2	w/8 x h/8	CS/AS	2	128	No	128	128
	w/8 x h/8	CS/AS	1	128	Prev/First	128	128
	w/8 x h/8	CS/AS	1	128	Prev/First	128	288
L3	w/16 x h/16	CS/AS	2	288	No	288	288
	w/16 x h/16	CS/AS	1	288	Prev/First	288	288
	w/16 x h/16	CS/AS	1	288	Prev/First	288	288
	w/16 x h/16	CS/As	1	288	Prev/First	416	416

SelecSLS module variants have a common design part which comprises a series of 3×3 convolutions interleaved with 1×1 convolutions. This is to enable mixing of channels when grouped 3×3 convolutions are used. All convolutions are followed by batch normalization and ReLU non-linearity. The module hyperparameter k dictates the number of features output by the convolution layers within the module. The outputs of all 3×3 convolutions ($2k$) are concatenated and fed to a 1×1 convolution which produces n_o features. The first 3×3 in the module has a stride of 1 or 2, which dictates the feature resolution of the entire module. The cross-module skip connection is the second input to the module.

On the one hand, two variants of *SelecSLS* module are compared, that handle cross-module skip connections *inside* the module in different ways: as additive-skip connections, henceforth *AS* (Figure 7.5 (a)) or as concatenation-skip connections, henceforth *CS* (Figure 7.5 (b)). The additive skip connection is added to the final 1×1 convolution before ReLU, and the ReLU is placed after the sum. When the number of features in the skip connection does not match n_o, a 1×1 convolution is used on the skip path to match the number of channels.

On the other hand, two variants of the cross-module connectivity design itself are investigated. The first is skip connectivity from the previous module, henceforth *Prev* (Figure 7.5 (c)), as it has been commonly employed. The second is skip connectivity from the first module in a level, henceforth *First* (Figure 7.5 (d)). A level is comprised of all modules in succession which output feature maps of a particular spatial resolution.

***SelecSLS Net* Architecture**: Table 7.9 shows the overall architecture of the proposed *SelecSLS Net*, parameterized by the type of module (*SelecSLS* concatenation-skip *CS* vs addition-skip *AS*), the stride of the module, s, the intermediate features in the module, k, cross-module skip connectivity (previous module or first module in the level), and number of outputs of the module, n_o. With the aim to promote information flow in the network, in addition to the base case of n_o, a wider case is also considered at the points of transition in spatial resolution. These will be indicated with (B) and (W) respectively. For 2D and 3D pose experiments, all 3×3 convolutions with more than 96 outputs use a group size of 2, and those with more than 192 outputs use a group size of 4. No grouping is used for ImageNet experiments.

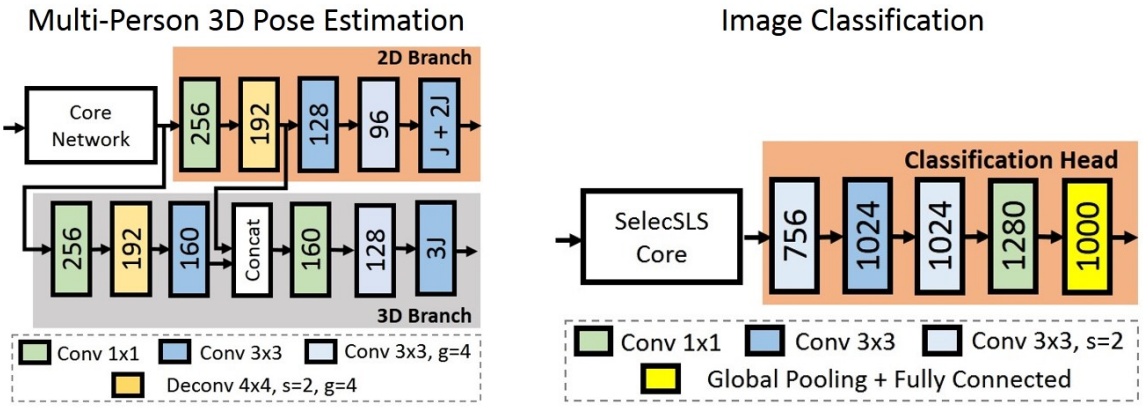

Figure 7.6: Architecture of *Stage I* of the multi-person 3D pose formulation described in Chapter 6 shown on the left, used here to compare different convolutional network backbones to the proposed SelecSLS architecture, as well as for validation of SelecSLS design choices. SelecSLS with an image classification head shown on the right for experiments on ImageNet dataset.

7.2.4 SelecSLS Design Evaluation

The best network design is determined by testing on 2D multi-person pose estimation, which, as discussed in Chapter 6, plays an integral role in the overall pipeline. The designs are used as the core network of the architecture shown in Figure 7.6 with only the 2D branch included for the purposes of *SelecSLS* design validation.

ResNet-50 and ResNet-34 based networks are used as the core to establish appropriate baselines. For ResNets, the network architecture is maintained until the first residual module in level-5 and striding is removed from level-5. The evaluations are done on a held-out 1000 frame subset of the MS-COCO validation set, and the Average Precision (AP) and Recall (AR) reported along with the inference time on different hardware in Table 7.10. Using the *AS* module with *Prev* connectivity and $n_o^{(B)}$ outputs for modules, the performance as well as the inference time on an Nvidia K80 GPU is close to that of ResNet-34. Using *CS* instead of addition-skip significantly improves the average precision from 47.0 to 47.6, and the average recall from 51.7 to 52.6. Switching the number of module outputs to the wider $n_o^{(W)}$ scheme leads to further improvement in AP and AR, at a slight increase in inference time. Using *First* connectivity further improves performance, namely to 48.6 AP and 53.3 AR, reaching close to ResNet-50 in AP (48.8) and performing slightly better with regard to AR (53.2). The proposed design has a 1.4-1.8$\times$ faster inference time than ResNet-50 across all devices. The publicly available model of Cao et al. 2017 evaluated on the same validation subset is 11 percentage points better on AP and AR than the proposed design, while being $10-20\times$ slower.

For subsequent experiments, a *SelecSLS Net* with concatenation-skip modules, cross-module skip connectivity to the first module in the level, and $n_o^{(W)}$ scheme for module outputs would therefore be used.

7.2.5 Comparisons With Other Network Architectures

Multi Person 2D Pose Performance: In the previous section (Table 7.10), the design choices were evaluated with regards to the *SelecSLS* module, and it was established that the proposed *SelecSLS* Net performs comparably to ResNet-50 while being significantly faster, when trained for multi-person

Table 7.10: Evaluation of design decisions for SelecSLS network. Different variants of SelecSLS, as well as ResNet-34/50 are used as core networks with the 2D pose branch of the multi-person network described in Chapter 6. The evaluations are done on a subset of validation frames of MS COCO dataset. Also reported are the forward pass timings of the core network and the 2D pose branch on different GPUs (K80, TitanX (Pascal)) as well as Xeon E5-1607 CPU on 512×320 pixel input. The publicly available model of Cao et al. 2017 is also evaluated on the same subset of validation frames.

	FP Time								
Core Network	**K80**	**TitanX**	**CPU**	**AP**	**$AP_{0.5}$**	**$AP_{0.75}$**	**AR**	**$AR_{0.5}$**	**$AR_{0.75}$**
ResNet-50	35.7ms	9.6ms	349ms	48.8	74.6	52.1	53.2	76.8	56.3
ResNet-34	25.7ms	5.7ms	269ms	46.4	72.7	47.3	51.3	75.2	52.8
SelecSLS									
Add-Skip Prev. (B)	24.5ms	6.5ms	167ms	47.0	73.4	49.7	51.7	75.6	54.5
Conc.-Skip Prev. (B)	24.3ms	6.3ms	172ms	47.6	73.3	50.7	52.6	76.1	55.6
Conc.-Skip Prev. (W)	25.0ms	6.7ms	184ms	48.3	74.4	51.1	52.9	76.5	55.7
Conc.-Skip First (W)	25.0ms	6.7ms	184ms	48.6	74.2	52.2	53.3	76.6	56.7
Cao et al. 2017	243ms	73.4ms	3660ms	58.0	79.5	62.9	62.1	81.2	66.5

Table 7.11: Evaluation of 2D keypoint detections of the complete *Stage I* XNect (both 2D and 3D branches trained), with different core networks on a subset of validation frames of MS COCO dataset. Also reported are the forward pass timings of the first stage on different GPUs (K80, TitanX (Pascal)) for an input image of size 512×320 pixels.

Core	**FP Time**							
Network	**K80**	**TitanX**	**AP**	**$AP_{0.5}$**	**$AP_{0.75}$**	**AR**	**$AR_{0.5}$**	**$AR_{0.75}$**
ResNet-34	29.0ms	6.5ms	45.0	72.0	46.1	49.9	74.4	51.6
ResNet-50	39.3ms	10.5ms	46.6	73.0	48.9	51.4	75.4	54.0
SelecSLS	28.6ms	7.4ms	**47.0**	**73.5**	**49.5**	**51.8**	75.6	**54.1**

2D pose estimation.

The networks are further trained with the 2 stage multi-person 3D pose pipeline introduced in Chapter 6, which adds a 3D pose branch to the core network to predict local 3D pose encodings, as shown in Figure 7.6(left). A separate 2nd stage comprised of a full-connected network uses the outputs of the first stage to yield full 3D poses for each subject in the scene. After adding the 3D pose branch, and training with additional MuCo-3DHP 3D pose training data, the networks are evaluated on the same MS-COCO validation subset in Table 7.11. Due to the addition of the 3D pose task, the 2D pose performance expectedly decreases, going down to 47.0 AP from 48.6, and 51.8 AR from 53.3, which outperforms ResNet-50. Even with the addition of the 3D pose branch, the inference time of Stage I stays under 29ms on an Nvidia K80 GPU.

Single Person and Multi Person 3D Pose Performance: The networks, trained for the multi-person 3D pose task, are then evaluated on both single and multi-person 3D benchmarks.

On the single person MPI-INF-3DHP benchmark, Table 6.8 shows that *SelecSLS* core architecture performs significantly better than ResNet-34 and slightly better than ResNet-50, with a higher 3DPCK and AUC and a lower MPJPE error. *SelecSLS* particularly results in significantly better performance for lower body joints (Knee, Ankle) than the ResNet baselines.

Similarly on the multi person 3D pose benchmark MuPoTS-3D, Table 7.12 shows that *SelectSLS*

Table 7.12: Evaluation of different core convolutional network choices with the 2 Stage multi-person 3D pose formulation from Chapter 6. The evaluations are on all annotated subjects in MuPoTS-3D using the 3D percentage of correct keypoints (**3DPCK**) metric. Also shown is the 3DPCK only for predictions that were matched to an annotation, as well as the accuracy split for visible and occluded joints.

	3DPCK				% Subjects
	All	Matched	Visible	Occluded	Matched
ResNet-34	67.0	72.6	70.4	55.3	92.1
ResNet-50	70.1	75.3	73.7	57.3	**93.0**
SelecSLS	**70.4**	**75.8**	**74.1**	**57.8**	92.8

Table 7.13: Comparison of limb joint 3D pose accuracy on MPI-INF-3DHP (Single Person) for different core convolutional network choices with the 2 Stage multi-person 3D pose formulation from Chapter 6. Metrics used are 3DPCK and AUC (higher is better).

	3DPCK				Total	
	Elbow	Wrist	Knee	Ankle	3DPCK	AUC
ResNet-34	79.6	61.2	83.0	52.7	79.3	41.8
ResNet-50	**82.4**	61.8	87.1	58.9	82.0	44.1
SelecSLS	81.2	**62.0**	**87.6**	**63.3**	**82.8**	**45.3**

Table 7.14: Comparison of limb joint 3D pose accuracy on MuPoTS-3D (Multi Person) for different core convolutional network choices with the 2 Stage multi-person 3D pose formulation from Chapter 6. The metric used is 3D Percentage of Correct Keypoints (3DPCK), evaluated with a threshold of 150mm.

	3DPCK					FP Time	
	Elbow	Wrist	Knee	Ankle	Total	K80	TitanX
ResNet-34	63.7	50.5	69.1	37.3	67.0	29.0ms	6.5ms
ResNet-50	65.8	**53.2**	71.0	47.3	70.1	39.3ms	10.5ms
SelecSLS	**66.8**	52.9	**72.2**	**47.6**	**70.4**	28.6ms	7.4ms

Table 7.15: Comparison of the proposed SelecSLS Net with ResNet-50 for image classification on ImageNet dataset. The timings are measured on an NVIDIA Titan Xp GPU.

	Forward Pass Time (ms) for different image resolutions and batch sizes						Forward/Backward Compute and Memory		ImageNet Accuracy		Total Params
Resolution	512x512		400x400		224x224		224x224		Top-1	Top-5	
Batch Size	1	16	1	16	1	16	FLOPS	Memory			
ResNet-50	15.0	175.0	11.0	114.0	7.2	39.0	4.1B	146.7MB	76.15	92.87	25.55M
SelecSLS	11.0	115.0	9.5	85.0	7.3	29.0	3.5B	92.55MB	76.22	92.96	30.67M

network architecture outperforms ResNet-50 and ResNet-34. With *SelecSLS* architecture, the pose accuracy for both visible and occluded joints improves over the ResNet baselines, even though there is a slight decrease in the number of detected subjects. Additionally, similar to the single person benchmark, Table 7.14 shows that *SelecSLS* particularly results in significantly better performance for lower body joints (Knee, Ankle) than the ResNet baselines.

Additionally, keeping the same multi-person *Stage I* convolutional networks, with *Stage II* trained on the single-person pose estimation task on Human3.6m, *SelecSLS* again outperforms ResNet baselines. The use of *SelecSLS* results in a mean per joint position error of 63.6mm, compared to 64.8mm using ResNet-50 and 67.6mm using ResNet-34.

Image Classification Performance: For evaluation on the task of image classification, the 1000 class dataset ImageNet is employed. Since ResNet-50 architecture was originally designed for image classification, the network is used unchanged from the reference model in Pytorch (Paszke et al. 2017). For SelecSLS, a classification head is used on top of the core network, in place of the 2D and 3D pose branches used for the pose estimation experiments. No channel grouping is used in the core network. The network architecture is shown in Figure 7.6(right).

SelecSLS Net's 76.22 Top-1 accuracy and 92.96 Top-5 accuracy on ImageNet is marginally better than that of ResNet-50, as shown in Table 7.15. However, the proposed network runs $1.3\times$ faster than ResNet-50 for larger batch sizes, as well as for larger images such as the ones used for multi-person 3D pose estimation. *SelecSLS* has 5M more parameters than ResNet-50, but 0.6B fewer FLOPS for 224x224 pixel images, and takes less than $2/3$ the memory of ResNet-50, which allows larger batch sizes while training.

7.3 Conclusion

This chapter presents surprising insights into implicit filter pruning in convolutional neural networks which employ Batch Normalization and ReLU non-linearities, and are trained with adaptive gradient descent methods. It is hypothesized that the root cause of sparsity is a combination of the propensity of Adam (and other adaptive gradient descent methods) to learn more selective features, a disproportionately higher degree of effective regularization seen by selective features in a mini-batch training setup, and the added acceleration of L2 regularization in a low gradient regime. The hypothesis is validated through extensive experimentation, and it is shown that the implicit sparsification is not due to the 'dying ReLU' phenomenon, and consequently is not ameliorated by supposed fixes such as Leaky ReLU.

This chapter also shows that filter pruning, explicit or implicit, is not an immediately viable route to faster convolutional network inference due to the loss in accuracy associated with an increase in sparsity and inference speed. Consequently, an new neural network architecture is proposed, which employs selective short-range concatenative-skip connections within the proposed building blocks, and selective long-range concatenative skip connections across modules. Experimental validation on our pose estimation task and various other tasks shows that the network performs at par or better than ResNet-50, while being $1.3-1.4\times$ faster on GPUs, particularly for larger images and larger batch sizes. The real-time multi-person motion capture system presented in the next chapter leverages the proposed architecture, SelecSLSNet, to allow accurate and real-time pose estimation from a monocular RGB camera.

Chapter 8

Real-time Monocular RGB Based Motion Capture

As detailed in Chapter 1, the objective of this thesis is to enable real-time, markerless, monocular-RGB based motion capture. The design decisions and development pertaining to the datasets, training schema, neural network architectures, and task formulation detailed in this thesis thus far have all been guided by this objective. The neural network based approaches developed in Chapters 5, 6, and 7 allow for fast, accurate, and in-the-wild 3D body pose estimation of one or more subjects given an RGB image as input, and additionally yield 2D body joint locations in the image plane.

This chapter develops monocular RGB based real-time motion capture systems which leverage the per-frame 3D and 2D pose estimates from approaches proposed in the thesis, combining it with model based skeleton-fitting approaches to produce temporally smooth joint angle estimates, as well as camera relative location estimates of the subjects in the scene.

In This Chapter

- Discussion of requirements of a motion capture system beyond per-frame root relative 3D body pose estimates (Section 8.1)
- Design of VNect, the first real-time monocular motion capture solution in the literature, targeted towards single-person scenes, leveraging the formulation proposed in Chapter 5, and a discussion of the salient components for fast and temporally smooth inference for the single-person case (Section 8.2)
- Extension of the monocular motion capture solution to XNect, the first monocular multi-person method in the literature, leveraging the formulation developed in Chapter 6, the fast neural network architecture developed in Chapter 7, and presentation of the additional salient components to enable temporally smooth motion capture in multi-person scenes (Section 8.3)
- Discussion of the applications of monocular RGB based motion capture to interactive virtual character control, 3D pose tracking in community videos etc.

The content of this chapter is based on D. Mehta et al. 2017b and D. Mehta et al. 2020.

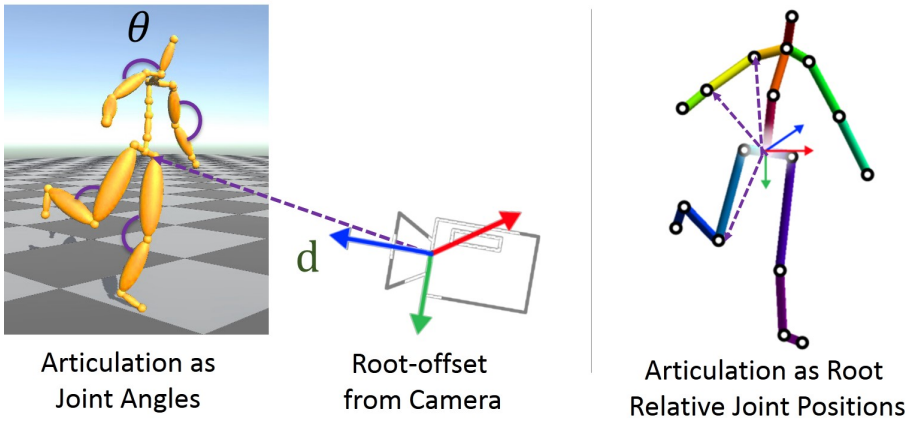

Figure 8.1: 3D Pose Representation: As described in Section 1.1.1, the output sought from the solution is the body skeleton articulation of the subjects in the image, expressed in terms of joint angles, θ, and localization of the subject relative to the camera, $\mathbf{d}$. The learning based component of the proposed solutions expresses body skeleton articulation in terms of root (pelvis) relative joint positions (right), and also predicts the 2D body keypoint location. These are converted to temporally smooth joint angles by the kinematic model fitting step described in this chapter.

8.1 Beyond Per-Frame Body Joint Location Estimates

The location-map approach developed in Chapter 5 for single-person scenarios, and the two stage XNect approach developed in Chapter 6 for multi-person scenarios, provide 3D articulation estimates of the kinematic structure of the subjects present in the scene. These articulation estimates are expressed in terms of joint positions relative to a reference joint, and assume the same height for all subjects, though the relative body proportions of the predictions may differ between subjects. These approaches are designed for the general case of single image input. Motion capture systems, as the name suggests, are meant to capture motion, i.e., temporally coherent sequences of poses. Applying the proposed per-frame methods to temporal sequences of input frames results in jittery pose estimates, particularly in cases of occlusion or disocclusion of body parts, and inconsistent bone lengths between frames. Further, many applications, particularly for computer graphics, require the kinematic structure articulation to be expressed in terms of joint angles, such that it can be readily used for animating rigged characters. Enabling experiences which allow one to control a character that inhabits and moves around in a virtual space require estimates of the camera relative location of the subject in addition to the subject's articulation.

Prior work has examined the use of spatio-temporal features (Bugra Tekin et al. 2016b) to gather information from consequtive frames for predicting the 3D pose. The work of M. Lin et al. 2017 instead aggregates 3D pose predictions across frames using a recurrent neural network module. Recent work of Pavllo et al. 2019 examines an alternative direction, and predicts 3D poses from long sequences of 2D pose predictions.

The approaches, VNect and XNect, discussed in the following sections use kinematic model fitting (inverse kinematics) on filtered sequences of 2D and 3D pose predictions, with various additional constraints, to produce temporally smooth joint angle estimates, and localize the skeletal structures relative to the camera. The following sections separately discuss the design of monocular motion capture system for single- and multi-person scenarios, with several design decisions specifically tailored to each scenario.

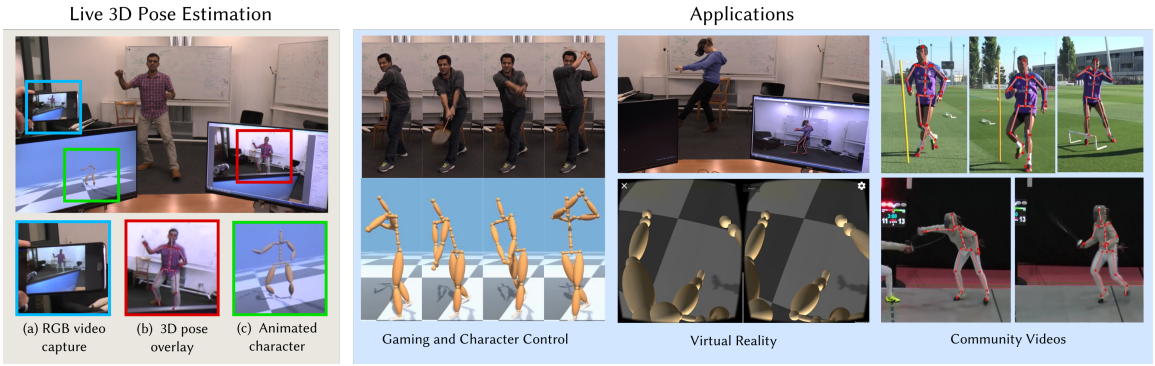

Figure 8.2: VNect recovers the full global 3D skeleton pose of a single subject in real-time from a single RGB camera, even wireless capture is possible by streaming from a smartphone (left). It enables applications such as controlling a game character, embodied VR, sport motion analysis and reconstruction of community video (right).

8.2 VNect: Real-time Monocular RGB Based Single-Person Motion Capture

For the case of single-person monocular RGB motion capture, the location-map approach described in Chapter 5 is used to predict the 2D joint positions $\mathbf{P^{2D}}_t$ and root-relative 3D joint positions $\mathbf{P^{3D}}_t$ per image frame $\mathbf{I}_t$ from a continuous input steam of monocular RGB images $\{...,\mathbf{I}_{t-1},\mathbf{I}_t\}$. For frame t in the input stream, the final output is $\mathbf{P}^{\mathrm{G}}_t(\boldsymbol{\theta},\mathbf{d})$ which is the full global 3D skeletal pose of the person being tracked, parameterized by the global position $\mathbf{d} \in \mathbb{R}^3$ in camera space, and joint angles $\boldsymbol{\theta} \in \mathbb{R}^{30}$ of the kinematic skeleton S. It is obtained using the kinematic model fitting step that combines the 2D and 3D joint position predictions to estimate a smooth, temporally consistent pose $\mathbf{P}^{\mathrm{G}}_t(\boldsymbol{\theta},\mathbf{d})$. The frame-number subscript t is dropped in certain sections to aid readability.

8.2.1 Kinematic Model Fitting

Skeleton Initialization: The kinematic model fitting step is set up with a default skeleton which works well out of the box for most adults. For more accurate estimates, the relative body proportions of the underlying skeleton can be adapted to that of the subject, by averaging the 3D predictions for a few frames at the beginning. Since monocular reconstruction is ambiguous without a scale reference, the neural network predicts height normalized 3D joint positions. If the subject's height is provided or gleaned from a calibration step, the camera relative localization estimates more closely resemble true metric space. Otherwise the camera relative localization is up to a scale.

First, the 2D predictions $\mathbf{K}_t$ are temporally filtered (Casiez et al. 2012) and used to obtain the 3D coordinates of each joint from the location-map predictions, yielding $\mathbf{P^{3D}}_t$, as described in Chapter 5. To ensure skeletal stability, the bone lengths inherent to $\mathbf{P^{3D}}_t$ are replaced by the bone lengths of the underlying skeleton in a simple retargeting step that preserves the bone directions of $\mathbf{P^{3D}}_t$. The resulting 2D and 3D predictions are combined by minimizing the objective energy

$$\begin{aligned} E_{\mathrm{total}}(\boldsymbol{\theta},\mathbf{d}) = E_{\mathrm{IK}}(\boldsymbol{\theta},\mathbf{d}) + E_{\mathrm{proj}}(\boldsymbol{\theta},\mathbf{d}) \\ + E_{\mathrm{smooth}}(\boldsymbol{\theta},\mathbf{d}) + E_{\mathrm{depth}}(\boldsymbol{\theta},\mathbf{d}), \end{aligned} \tag{8.1}$$

for skeletal joint angles $\boldsymbol{\theta}$ and the root joint's location in camera space $\mathbf{d}$. The 3D inverse kinematics term E_{IK} determines the overall pose by similarity to the 3D CNN output $\mathbf{P^{3D}}_t$. The projection term

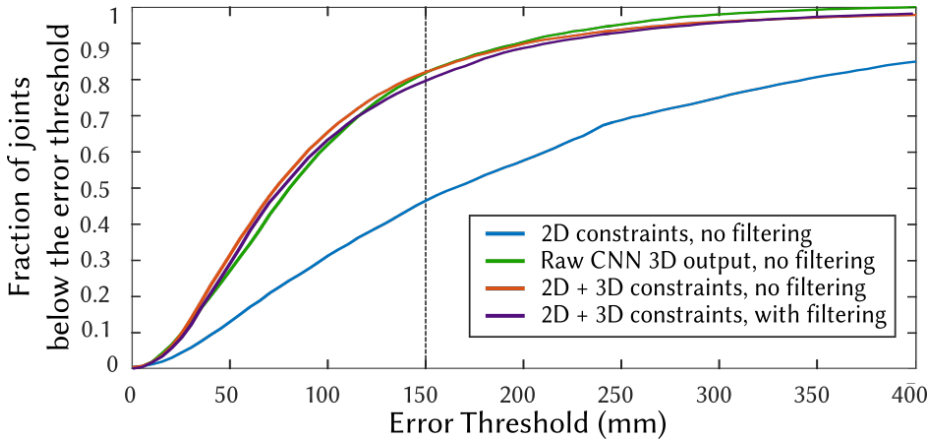

Figure 8.3: Fraction of joints correctly predicted on the TS1 sequence of MPI-INF-3DHP test set, as determined by the distance between the predicted joint location and the ground truth joint location being below the error threshold. The dotted line marks the 150mm threshold for which the 3D PCK numbers are reported. Only using the 2D predictions as constraints for skeleton fitting (blue) performs significantly worse than using both 2D and 3D predictions as constraints (red). Though adding 1 Euro filtering (purple) visually improves the results, the slightly higher error here is due to the sluggish recovery from tracking failures. The 3D predictions from the CNN (green) are also shown.

E_{proj} determines global position $\mathbf{d}$ and corrects the 3D pose by re-projection onto the detected 2D keypoints $\mathbf{P^{2D}}_t$. Both terms are implemented with the L2 norm,

$$E_{\text{proj}} = \|\Pi(\mathbf{P}^{\text{G}}_t) - \mathbf{P^{2D}}_t\|_2 \text{ and } E_{\text{IK}} = \|(\mathbf{P}^{\text{G}}_t - \mathbf{d}) - \mathbf{P^{3D}}_t\|_2, \tag{8.2}$$

where Π is the projection function from 3D to the image plane, and $\mathbf{P}^{\text{G}}_t = FK(\boldsymbol{\theta}, \mathbf{d})$ is joint positions obtained by applying forward kinematics to the kinematics skeleton. A pinhole projection model is used. If the camera calibration is unknown a vertical field of view of 54 degrees is assumed. Temporal stability is enforced with smoothness prior $E_{\text{smooth}} = \|\widehat{\mathbf{P}^{\text{G}}_t}\|_2$, penalizing the acceleration $\widehat{\mathbf{P}^{\text{G}}_t}$. To counteract the strong depth uncertainty in monocular reconstruction, large variations in depth are additionally penalized with $E_{\text{depth}} = \|[\widehat{\mathbf{P}^{\text{G}}_t}]_z\|_2$ where $[\widehat{\mathbf{P}^{\text{G}}_t}]_z$ is the z component of 3D velocity $\widehat{\mathbf{P}^{\text{G}}_t}$. Finally, the 3D pose is also filtered with the 1 Euro filter (Casiez et al. 2012).

Parameters: The energy terms $E_{\text{IK}}, E_{\text{proj}}, E_{\text{smooth}}$ and E_{depth} are weighted with $\omega_{\text{IK}} = 1, \omega_{\text{proj}} = 44, \omega_{\text{smooth}} = 0.07$ and $\omega_{\text{depth}} = 0.11$, respectively. The parameters of the 1 Euro Filter are empirically set to $f_{c_{\text{min}}} = 1.7$, $\beta = 0.3$ for filtering $\mathbf{P^{2D}}_t$, to $f_{c_{\text{min}}} = 0.8$, $\beta = 0.4$ for $\mathbf{P^{3D}}_t$, and to $f_{c_{\text{min}}} = 20$, $\beta = 0.4$ for filtering $\mathbf{P}^{\text{G}}_t$. The implementation uses the Levenberg-Marquardt algorithm from the Ceres library (S. Agarwal et al. 2017).

8.2.2 Bootstrapped No-cost Bounding Box Tracking

Although the fully-convolutional location-map formulation allows the neural network to work without requiring cropping, the run-time of the neural network is highly dependent on the input image size. Additionally, the neural network is trained for subject sizes in the range of 250–340 px in the frame, requiring averaging of predictions at multiple image scales per frame or a scale space search if processing the full frame at each time step. Guaranteeing real-time rates, particularly with a ResNet-50 backbone, necessitates restricting the size of the input to the network and tracking the scale of the person in the image to avoid searching the scale space in each frame. Even with the

use of the fast *SelecSLSNet* architecture proposed in Chapter 7 as the backbone, further significant speedup can be obtained by tracking the location and scale of the person in the scene in single-person scenarios.

With the proposed location-map formulation, the person's location and scale can be tracked in an integrated manner, with negligible runtime and compute overhead. The 2D pose predictions from the neural network at each frame are used to determine the bounding-box for the next frame through a slightly larger box around the predictions. The smallest rectangle containing the keypoints $\mathbf{K}$ is computed and augmented with a buffer area $0.2\times$ the height vertically and $0.4\times$ the width horizontally. To stabilize the estimates, the bounding-box is shifted horizontally to the centroid of the 2D predictions, and its corners are filtered with a weighted average with the previous frame's bounding-box using a momentum of 0.75. To normalize scale, the bounding-box crop is resized to 368x368 px. The bounding-box tracker starts with (slow) multi-scale predictions on the full image for the first few frames, and hones in on the person in the image making use of the bounding-box agnostic predictions from the fully convolutional network.

8.2.3 System Characteristics and Comparisons

Real-time Performance: The system runs on a 6-core Xeon CPU, 3.8 GHz and a single Titan X (Pascal architecture) GPU. The CNN computation (ResNet-50) takes ≈18 ms, the skeleton fitting ≈7–10 ms, and preprocessing and filtering 5 ms, allowing live applications to run at ≈ 33 FPS. The ResNet-50 based backbone network can be replaced with *SelecSLSNet* for further speedup. For video results, refer to gvv.mpi-inf.mpg.de/projects/VNect/.

Quality of Tracking: The reconstruction quality is high, as shown in Chapter 5 and additional video results at gvv.mpi-inf.mpg.de/projects/VNectDemo/, and the usefulness of the proposed method is shown for various applications such as 3D character control, embodied virtual reality, and pose tracking from low quality smartphone camera streams. Results are best observed in motion in the video. The importance of the steps towards enabling these applications with a video solution are thoroughly evaluated on more than 10 sequences. Results are comparable in quality to depth-camera based solutions like the Kinect (Microsoft Corporation 2013).The system is tested on a variety of subjects, and succeeds for different body shapes, gender and skin tone, as well as to a variety of backgrounds and camera types, as shown applied on community videos in the video results.

The kinematic skeleton fitting estimates global translation $\mathbf{d}$ up to a scale. Figure 8.4 demonstrates that the translation estimates exhibit negligible drift, the feet position matches with the same reference point after performing a circular walk. The smoothness constraint in depth direction limits sliding of the character away from the marker, as pictured in the video results available at gvv.mpi-inf.mpg.de/projects/VNect/.

Ablation of VNect Components: The effect of the various components of the full VNect pipeline is shown on the TS1 sequence of MPI-INF-3DHP test set in Figure 8.3. Without the E_{IK} component of E_{total} the tracking accuracy goes down to a PCK of 46.1% compared to a PCK of 81.7% when E_{IK} is used. The raw CNN 3D predictions in conjunction with the bounding-box tracker result in a PCK of 80.3%. Using E_{IK} in E_{total} produces consistently better results for all thresholds lower than 150 mm. This shows the improvements brought about by the skeleton fitting term. Additionally, as shown in the video results, using 1 Euro filtering produces qualitatively better results, but the overall PCK decreases slightly (79.7%) due to slower recovery from tracking failures.

The influence of the smoothness and filtering steps on the temporal consistency are further analyzed

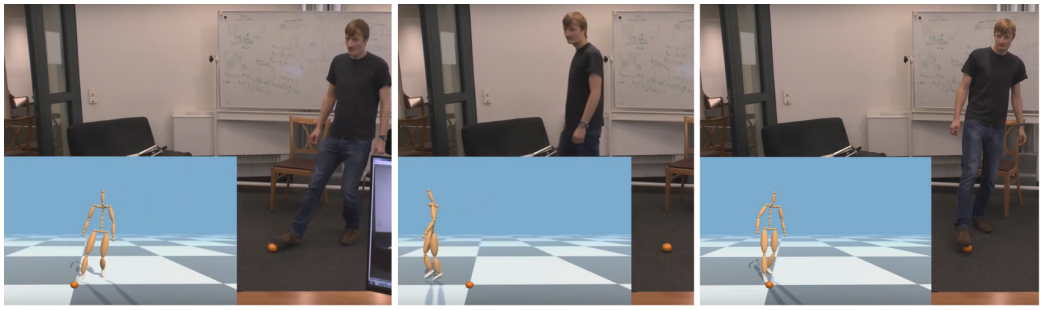

Figure 8.4: The estimated 3D pose from VNect is drift-free. The motion of the person starts and ends at the marked point (orange), both in the real world and in our reconstruction.

in the video.

Comparison with Active Depth Sensors (Kinect): To compare with depth sensing based solutions, video from an RGB camera and a co-located Kinect sensor were synchronously recorded in a living room scenario. Figure 8.7 shows representative frames. Although the depth sensor provides additional information, VNect reconstructions from just RGB are of a similar quality. The Kinect results are of comparable stability to VNect, but yield erroneous reconstructions when limbs are close to scene objects, such as when sitting down. VNect, however, succeeds in this case, although is slightly less reliable in depth estimation. A challenging case for both methods is the tight crossing of legs. The video contains a visual comparison.

The video solution succeeds also in situations with direct sunlight (Figure 8.6), where IR-based depth cameras are inoperable. Moreover, RGB cameras can simply be equipped with large field-of-view (FOV) lenses and, despite strong distortions, successfully track humans (Rhodin et al. 2016a). On the other hand, existing active sensors are limited to relatively small FOVs, which severely limits the tracking volume.

8.2.4 Applications

The real-time, temporally stable, fully automatic, joint angle estimates produced by VNect are suitable for various interactive applications, and amenable to direct application to 3D character control.

Character Control: Real-time motion capture solutions provide a natural interface for game characters and virtual avatars, which go beyond classical mouse and gamepad control. VNect works successfully for a wide range of motions common in activities like tennis, dance, and juggling, see Figures 8.2 and 8.5. The swing of the arm and leg motion is nicely captured and could, for instance, be used in a casual sports and dancing game, but also for motion analysis of professional athletes to optimize their motion patterns. Successful results are also shown in non front-facing motions such as turning and writing on a wall, as well as squatting.

Virtual Reality: The recent availability of cheap head-mounted displays has sparked a range of new applications. Many products use handheld devices to track the user's hand position for interaction. VNect enables such experiences from a single consumer color camera. Beyond interaction, the marker-less full-body solution enables embodied virtual reality, see Figure 8.2. A rich immersive feeling is created by posing a virtual avatar of the user exactly to their own real pose. With the proposed solution the real and virtual pose are aligned such that users perceive the virtual body as

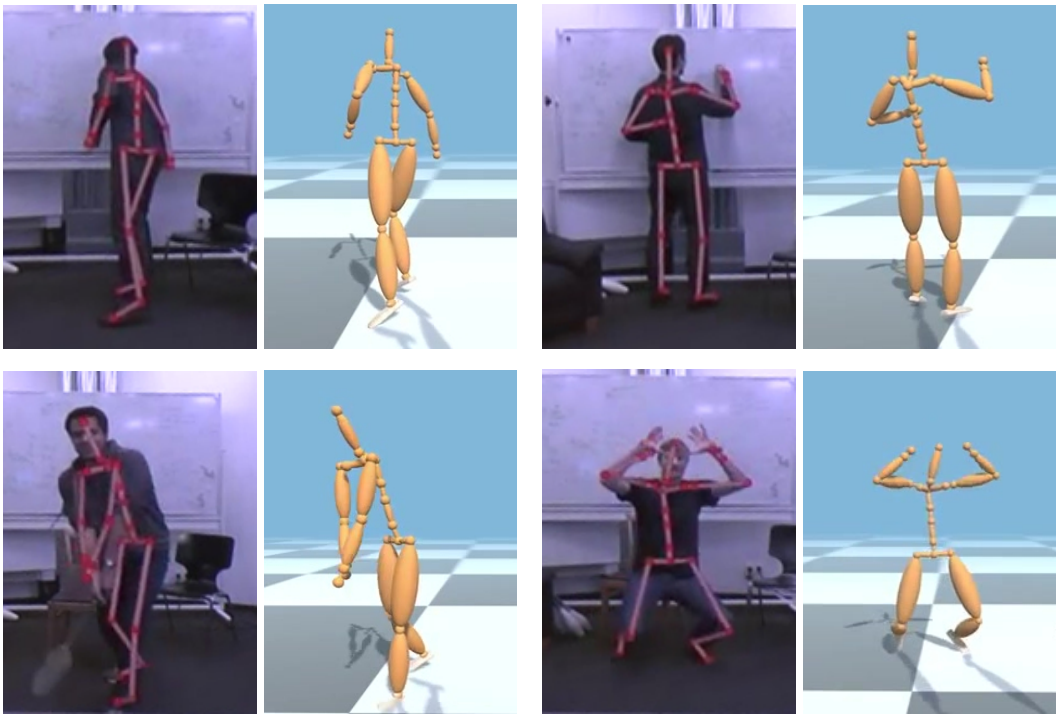

Figure 8.5: Application to entertainment. VNect, the single-person real-time 3D pose estimation method provides a natural motion interface, e.g. for sport games.

Figure 8.6: VNect, being based on RGB camera input succeeds in strong illumination and sunlight (center right and right), while the IR-based depth estimates of the Microsoft Kinect are erroneous (left) and depth-based tracking fails (center left).

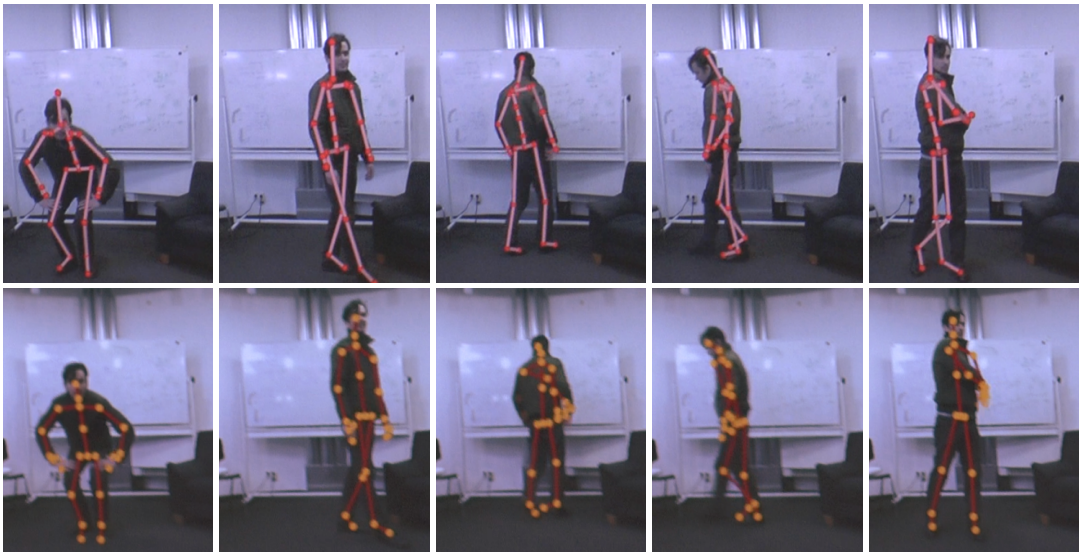

Figure 8.7: Side-by-side pose comparison with VNect (top) and Kinect (bottom). Overall estimated poses are of similar quality (first two frames). Both the Kinect (third and fourth frames) and VNect (fourth and fifth frames) occasionally predict erroneous poses.

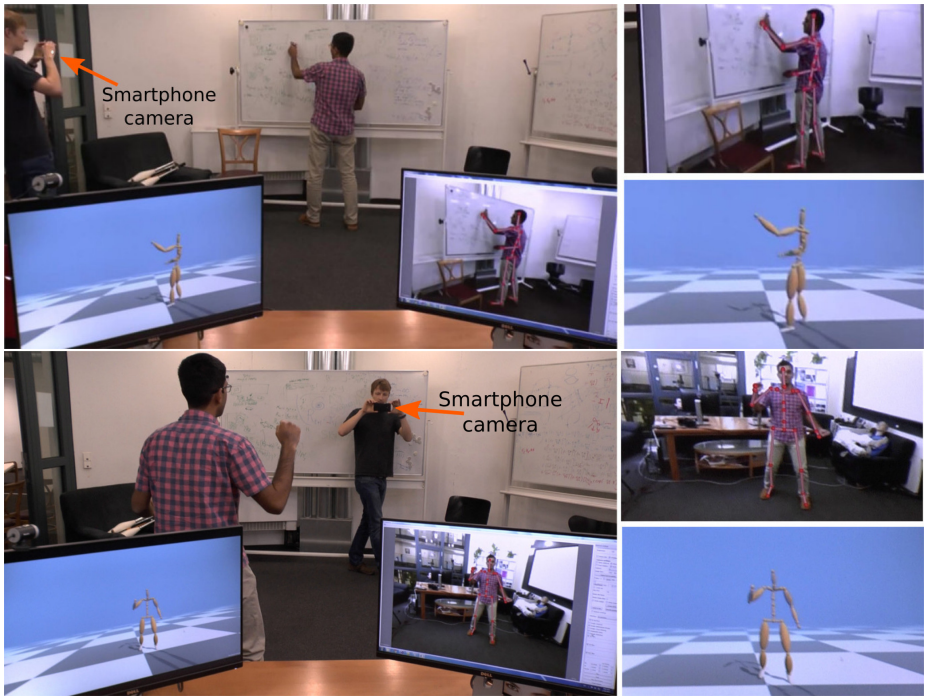

Figure 8.8: Handheld recording with a readily available smartphone camera (left) and the estimated pose from VNect (right), streamed to and processed by a GPU enabled PC.

their own.

Ubiquitous Motion Capture with Smartphones: Real-time monocular 3D pose estimation lends itself to application on low quality smartphone video streams. By streaming the video to a machine with sufficient capabilities for the proposed algorithm, one can turn any smartphone into a lightweight, fully-automatic, handheld motion capture sensor, see Figure 8.8 and the accompanying video. Since smartphones are widespread, it enables the aforementioned applications for casual users without requiring additional sensing devices. Recent work on developing more capable mobile computing, targeted towards accelerating on-device neural network inference, could allow the system to partially or entirely run on the mobile capture device.

8.3 XNect: Real-time Monocular RGB Based Multi-Person Motion Capture

For the multi-person case, the occlusion robust and efficient 2-stage per-frame XNect formulation developed in Chapter 6 is employed. As discussed in Chapters 6 & 7, bounding box tracking based inference in multi-person scenarios would lead to a linear dependence of the inference time on the number of subjects in the scene. Hence, a different approach is adopted for multi-person scenarios, wherein the complete image frame is processed at every time step. The output jointly encodes the 2D and 3D pose of all the subjects in the scene. Real-time inference in this case is made possible through the use of *SelecSLSNet* architecture proposed in Chapter 7.

There are additional key ways in which the multi-person solution differs from the single-person solution. In order to reconcile 2D and 3D pose predictions of subjects across frames, the subject identities need to be associated across frames. Further, because of occasional full body occlusions in multi-person scenarios, the identity association should also be able to re-identify lost person tracks. The estimation of camera relative translation is only up to scale, and requires knowledge of the subject heights to accurately localize the subject in the scene. Estimating subject heights is particularly crucial for multi-person scenarios where subjects have starkly different heights, as without knowing the relative heights of the subjects, their relative localization estimates would not be correct. With a fixed height assumption, shorter subjects would get placed further away from the camera and vice-versa. Hence, a calibration step is required to estimate the heights of the subjects in the scene.

8.3.1 Identity Tracking and Re-identification

The per-frame root-relative 3D pose estimates $\{P_i^{3D}[t]\}$, and 2D pose estimates $\{P_i^{2D}[t]\}$ for each individual i in the scene are obtained using the 2-stage per-frame XNect formulation. The identity tracking step assigns correspondences between person detections at the current timestep t, $\{P_i[t]\}_{i=1}^{K[t]}$, to the preceding ones $\{P_k[t-1]\}_{k=1}^{K[t-1]}$. Person appearance is modeled with an HSV color histogram of the upper body region. The hue and saturation channels are discretized into 30 bins each and determine the appearance $A_{i[t]}$ as the class probabilities across the bounding box enclosing the torso joints in $\{P_i^{2D}[t]\}_i$. This descriptor is efficient to compute and can model loose and tight clothing alike, but might suffer from color ambiguities across similarly dressed subjects.

To be able to match subjects robustly, detections from current frame t are assigned to previously known identities not only based on appearance similarity, $S_{i,k}^A = (A_i[t] - A_k[t-1])^2$, but also on the

Figure 8.9: XNect, the real-time monocular RGB based 3D motion capture system provides temporally coherent estimates of the full 3D pose of multiple people in the scene, handling occlusions and interactions in general scene settings, and localizing subjects relative to the camera. The design allows the system to handle large groups of people in the scene with the run-time only minimally affected by the number of people in the scene. The method yields full skeletal pose in terms of joint angles, which can readily be employed for interactive character animation.

2D pose similarity $S^{P2D}_{i,k}(i,k) = (P^{2D}_{i[t]} - P^{2D}_{k[t-1]})^2$ and 3D pose similarity $S^{P3D}_{i,k}(i,k) = (P^{3D}_{i[t]} - P^{3D}_{k[t-1]})^2$. A threshold on the dissimilarity is set to detect occlusions, persons leaving the field of view, and new persons entering. That means the number of persons $K[t]$ can change. Person identities are maintained for a certain number of frames after disappearance, to allow for re-identification after momentary occlusions such as those caused by the tracked subjects passing behind an occluder. The appearance histogram of known subjects is updated at arrival time and every 30 seconds to account for appearance changes such as varying illumination. Once the identity is assigned, all following steps are performed separately per person.

8.3.2 Relative Bone Length and Absolute Height Calculation

Relative bone length estimates are obtained similar to VNect, described in the previous section. Similar to VNect, camera relative position up to a scale is recovered through a re-projection constraint. To allow more accurate camera relative localization, the ground plane can optionally be used as reference geometry since camera calibration is less cumbersome than measuring the height of every person appearing in the scene. First, the camera relative position of a person is determined by shooting a ray from the camera origin through the person's foot detection in 2D and computing its intersection with the ground plane. The subject height, h_k, is then the distance from the ground plane to the intersection point of a virtual billboard placed at the determined foot position and the view ray through the detected head position. Since capturing dynamic motions such as jumping, running and partial (self-)occlusions is desired, the ankle cannot be assumed to be visible and touching the ground at every frame. Instead, this strategy is used only once when the person appears in the scene, and the height is tracked with the identity. To determine the ground plane and compute intrinsic and extrinsic camera parameters, checkerboard calibration can be used once prior to using the motion capture system. Other object-free calibration approaches would be feasible alternatives (F. Yang et al. 2018; Zanfir et al. 2018a).

Figure 8.10: Virtual Character Control: The temporally smooth joint angle predictions from XNect *Stage III* can be readily employed for driving virtual characters.

8.3.3 Kinematic Model Fitting

After 2D and 3D joint position prediction, the skeletal pose $\{\mathbf{P}_t^{\mathrm{G}}(\boldsymbol{\theta},\mathbf{d})_k[t]\}_{k=1}^{K[t]}$ of all $K[t]$ people in the scene is optimized for, parameterized by the global position $\mathbf{d}_k[t] \in \mathbb{R}^3$ and joint angles $\boldsymbol{\theta}_k[t] \in \mathbb{R}^{29}$.

Both, per-frame 2D and 3D pose estimates from previous stages are temporally filtered (Casiez et al. 2012) before skeleton fitting. Note that $\theta_k \in \mathbb{R}^D$ describes the pose of a person in terms of joint angles of a fixed skeleton plus the global root position, meaning that XNect *Stage III* output is directly compatible with CG character animation pipelines. Similar to VNect, $\mathbf{P}_k^{\mathrm{G}}[t]$ is estimated by minimizing the fitting energy

$$\begin{aligned}\mathcal{E}(\mathbf{P}_1^{\mathrm{G}}[t],\cdots,\mathbf{P}_K^{\mathrm{G}}[t]) = w_{3\mathrm{D}}E_{3\mathrm{D}} + w_{2\mathrm{D}}E_{2\mathrm{D}} + w_{\mathrm{lim}}E_{\mathrm{lim}} \\ + w_{\mathrm{temp}}E_{\mathrm{temp}} + w_{\mathrm{depth}}E_{\mathrm{depth}}\ .\end{aligned} \tag{8.3}$$

$\frac{\partial \mathcal{E}}{\partial \theta_k[t]}$ is formulated in closed form to perform efficient minimization by gradient descent. The influence of the individual terms is balanced with $w_{3\mathrm{D}} = 9\mathrm{e}{-1}$, $w_{2\mathrm{D}} = 1\mathrm{e}{-5}$, $w_{\mathrm{lim}} = 5\mathrm{e}{-1}$, $w_{\mathrm{temp}} = 1\mathrm{e}{-7}$, and $w_{\mathrm{depth}} = 8\mathrm{e}{-6}$. The following explains each term in more detail.

3D Inverse Kinematics Term:: The 3D fitting term measures the 3D distance between predicted root-relative 3D joint positions $P_k^{3D}[t]$ and the root-relative joint positions in the skeleton $\mathbf{P}_k^{\mathrm{G}}[t] = FK(\boldsymbol{\theta}_k[t],\mathbf{d}_k[t])$ posed by forward kinematics for every person k, joint j.

$$E_{3\mathrm{D}} = \sum_{k=1}^{K}\sum_{j=1}^{J3D} ||\mathbf{P}_{k,j}^{\mathrm{G}} - \mathbf{d}_k - P_{k,j}^{3D}[t]||_2^2 \quad . \tag{8.4}$$

2D Re-projection Term: The 2D fitting term is calculated as the 2D distance between predicted 2D

joint positions $P_k^{2D}[t]$ and the projected skeleton joint positions $\mathbf{P}_k^{\mathrm{G}}[t]$ for every person k and joint j,

$$E_{\mathrm{2D}} = \sum_{k=1}^{K} \sum_{j=1}^{J2D} c_{j,k} ||\Pi(h_k \mathbf{P}_k^{\mathrm{G}}[t])_j - P_{k,j}^{2D}[t]\,||_2^2 \quad , \tag{8.5}$$

where c is the 2D prediction confidence, and Π is the camera projection matrix. Note that $\mathcal{P}$ outputs unit height, the scaling with h_k maps it to metric coordinates, and the projection constraint thereby reconstructs absolute position in world coordinates.

Joint Angle Limit Term: The joint limits regularizer enforces a soft limit on the amount of joint angle rotation based on the anatomical joint rotation limits θ^{min} and θ^{max}. It is expressed as

$$E_{\mathrm{lim}} = \sum_{k=1}^{K} \sum_{j=7}^{26} \begin{cases} (\theta_j^{min} - \theta_{k,j}[t])^2 & ,\text{if } \boldsymbol{\theta}_{k,j}[t] < \theta_j^{min} \\ (\theta_{k,j}[t] - \theta_j^{max})^2 & ,\text{if } \boldsymbol{\theta}_{k,j}[t] > \theta_j^{max} \\ 0 & ,\text{otherwise} \end{cases} \quad , \tag{8.6}$$

where constraints start from $j = 4$ since there are no limits placed on the root rotation parameters.

Temporal Smoothness Term: The temporal stability of the estimated poses is improved by

$$E_{\mathrm{temp}}(\Theta) = \sum_{k=1}^{K} ||\nabla \mathbf{P}_k^{\mathrm{G}}[t-1] - \nabla \mathbf{P}_k^{\mathrm{G}}[t]||_2^2 \quad , \tag{8.7}$$

where the rate of change in parameter values, $\nabla \mathbf{P}_k^{\mathrm{G}}$, is approximated using backward differencing. In addition, variations in the less constrained depth direction is penalized stronger, using the smoothness term $E_{\mathrm{depth}} = ||\mathbf{d}_{k,3}[t]_z - \mathbf{d}_{k,3}[t-1]||$, where $\mathbf{d}_{k,3}$ is the degree of freedom that drives the z-component of the root position.

8.3.4 Results, Comparisons, and Applications

The system provides efficient and accurate 3D motion capture that is ready for live character animation and other interactive CG applications, rivaling depth-based solutions despite using only a single RGB video feed.

Real-time Performance: The live system uses a standard webcam as input, and processes 512×320 pixel resolution input frames. The system running on a Desktop with an Intel Xeon E5 with 3.5 Ghz and an Nvidia GTX 1080Ti is capable of processing input at > 30 FPS, while on a laptop with an Intel i7-8780H with a 1080-MaxQ it runs at ≈ 25 fps. Additional video results are available at gvv.mpi-inf.mpg.de/projects/XNect/, showing examples of the live setup of XNect running on a laptop

Multi-Person Scenes and Occlusion Robustness:: Figures 8.13 and 8.12 show qualitative results of the full XNect system on MuPoTS-3D and Panoptic [2015] datasets with scenes containing multiple interacting and overlapping subjects. Single-person real-time approaches such as VNect are unable to handle such scenes containing occlusions or multiple people in close proximity. For further qualitative results on a variety of scene settings, including community videos and live scene setups, refer to video results and Figure 8.15.

Comparison With KinectV2: The quality of XNect pose estimates with a single RGB camera is comparable to those from off the shelf depth sensing based systems such as KinectV2 (Figure 8.11),

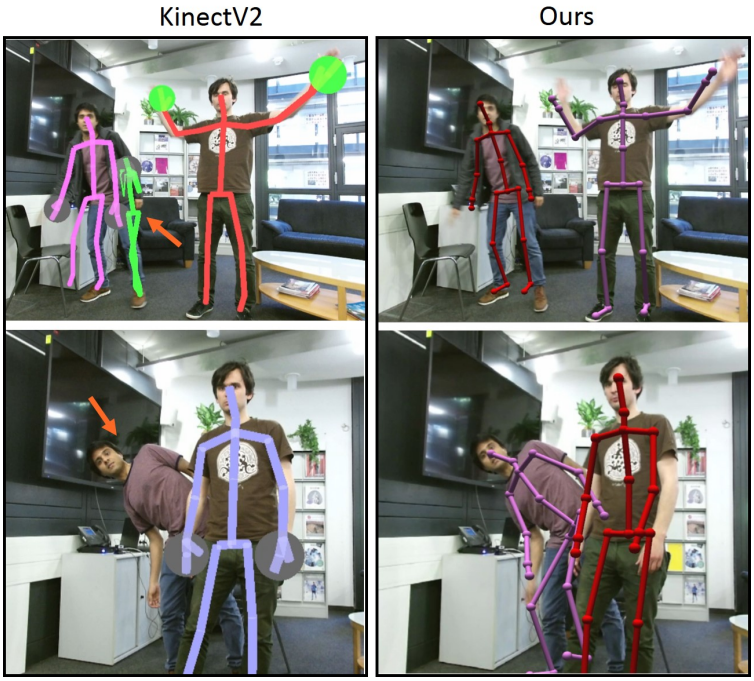

Figure 8.11: The quality of XNect pose estimates is comparable to depth sensing based approaches such as KinectV2, and XNect handles certain cases of significant inter-personal overlap and cluttered scenes better than KinectV2. In the top row, due to scene clutter, KinectV2 predicts multiple skeletons for one subject. In the bottom row, the person at the back with lower body occlusion is not detected by KinectV2.

with XNect succeeding in certain cluttered scenarios where person identification from depth input would be ambiguous. The accompanying video contains further visual comparisons.

Character Animation: Since XNect reconstructs temporally coherent joint angles and the camera relative subject localization estimates are stable, the output of the system can readily be employed to animate virtual avatars as shown in Figure 8.10. The video demonstrates the stability of the localization estimates of XNect and contains further examples of real-time interactive character control with a single RGB camera.

Evaluation of Skeleton Fitting (Stage III): Skeleton fitting to reconcile 2D and 3D poses across time results in smooth joint angle estimates which can be used to drive virtual characters. However, for pose classes with significant self occlusions, where 2D pose estimates are not reliable, there is a significant decrease in joint position accuracy after skeleton fitting. On the single person 3D pose benchmark MPI-INF-3DHP shown in Table 8.1, the overall 3DPCK decreases to 79.3 from 82.8. However, for pose classes such as standing, exercising etc, the pose accuracy is not affected significantly after skeleton fitting.

8.4 Discussion

Despite the real-time, in-the-wild performance of the proposed systems, certain limitations remain to be addressed through future work.

As with other monocular approaches, the accuracy of the proposed approaches is not comparable yet to the accuracy of multi-view capture algorithms. Failure cases in the proposed systems can arise

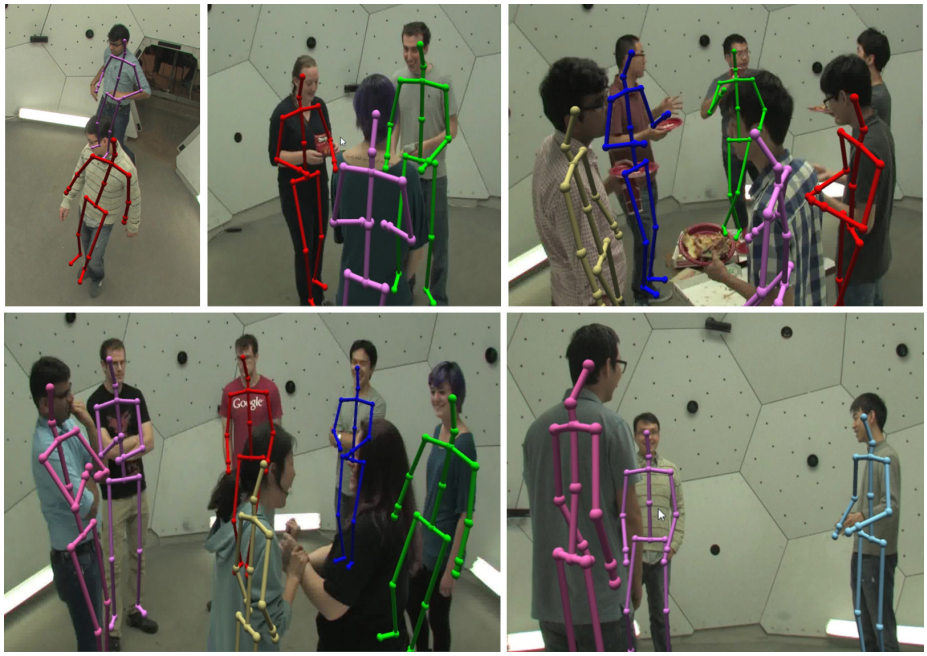

Figure 8.12: Qualitative results of XNect (Stage III) on the Panoptic [2015] dataset. XNect works with significant occlusions, such as the half body view and interpersonal occlusions seen here, as well as overhead viewpoints.

Table 8.1: Comparison of XNect *Stage II* 3D pose output (before skeleton fitting) with *Stage III* (after skeleton fitting), on MPI-INF-3DHP dataset. Metrics used are: 3D percentage of correct keypoints (3DPCK) and, area under the curve (AUC), higher is better.

	3DPCK			Total	
	Stand /Walk	Sitt.	On The Floor	3DPCK	AUC
Stage II Output	**88.4**	**85.8**	**70.7**	**82.8**	**45.3**
Stage III Output	88.5	82.6	52.6	79.3	41.2

from each of the constituent stages. The 3D pose estimates can be incorrect if the underlying 2D pose estimates (in single-person or multi-person case) or part associations (in the multi-person case) are incorrect. Also, since XNect requires the neck to be visible for a successful detection of a person, scenarios such as that in Figure 8.14(b) result in the person not being detected despite being mostly visible.

XNect successfully captures the pose of occluded subjects even under difficult inter-person occlusions that are generally hard for monocular methods. However, the approach still falls short of reliably capturing extremely close interactions, like hugging. Incorporation of physics-based motion constraints could further improve pose stability in such cases, may add further temporal stability, and may allow capturing of fine-grained interactions of persons and objects.

For both VNect and XNect, in some cases individual poses have higher errors for a few frames, e. g. after strong occlusions or when the face is occluded (see video results). However, the proposed approaches manage to recover from this. The kinematic fitting step may suffer from inaccuracies under cases of significant inter-personal or self occlusion, making the camera relative localization

Figure 8.13: Qualitative results of XNect (Stage III) on MuPoTS-3D dataset. As seen here, XNect works in different scene settings, and handles significant interpersonal occlusions.

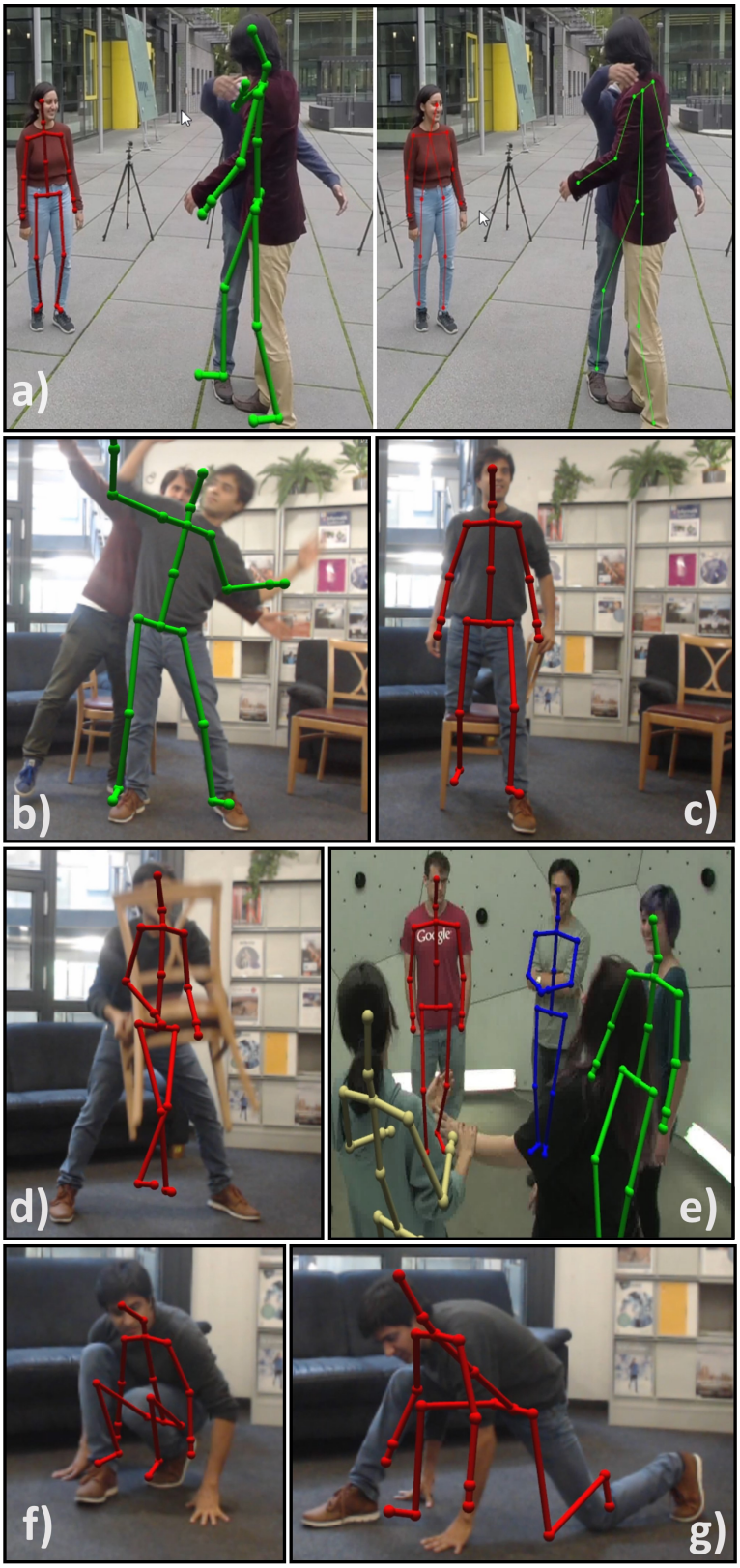

Figure 8.14: XNect Failure Cases: a),c) 3D pose inaccuracy due to 2D pose limb confusion, b) Person not detected due to neck occlusion, d),e) 3D misprediction and person undetected under extreme occlusion, f),g) 2D-3D pose alignment becomes unreliable in cases with significant self occlusion

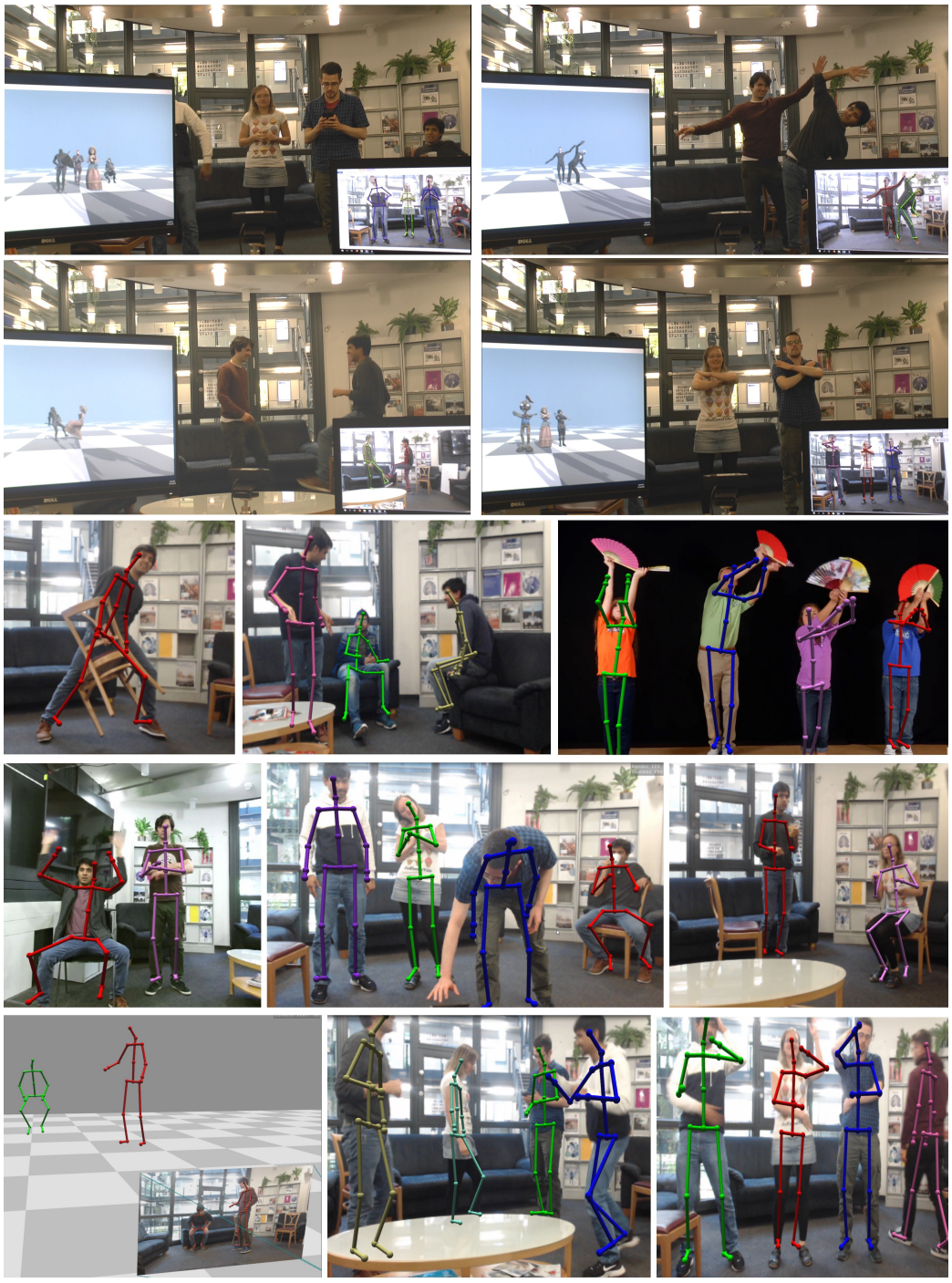

Figure 8.15: Real-time 3D motion capture results with XNect on a wide variety of multi-person scenes. XNect handles challenging motions and poses, including interactions and cases of self-occlusion. The top two rows show the live system tracking subjects in real-time and driving virtual characters with the captured motion. Refer to the video for more results.

less stable in those scenarios. Still, reconstruction accuracy and stability is appropriate for many real-time applications.

Since XNect has many additional components compared to VNect, it consequently has additional modes of failure arising from those components. For XNect, the relatively simple identity tracker may swap identities of people when tracking through extended person occlusions, drastic appearance changes, and similar clothing appearance. More sophisticated space-time tracking would be needed to remedy this. As with all learning-based pose estimation approaches, pose estimation accuracy for both VNect and XNect worsens on poses very dissimilar from the training poses. One possible approach to remedying this would be to leverage unlabeled videos in an unsupervised or semi-supervised manner.

8.5 Conclusion

This chapter presents the first real-time approaches for single-person and multi-person 3D motion capture using a single RGB camera. These operate in generic scenes, and the multi-person system is robust to occlusions both by other people and objects. The approaches provides joint angle estimates and localize subjects relative to the camera. Importantly, the proposed systems run on consumer hardware at more than 30 FPS while achieving state-of-the-art accuracy, and these advances are demonstrated on a range of challenging real-world scenes.

Chapter 9

Conclusion and Outlook

This thesis develops the key components to enable accurate and real-time pose estimation for one or more subjects in general scene settings. The proposed innovations result in the first monocular RGB based motion capture systems (Chapter 8), for single- and multi-person scenarios, that run in real-time on consumer grade laptops and desktops. These provide temporally smooth joint angle estimates which can readily be employed for various interactive and non-interactive applications, such as driving virtual characters in games or for motion analysis.

Towards this, the thesis examines aspects of data (Chapter 3), training pipeline (Chapter 4), task formulation (Chapters 5 & 6), and neural network architecture design insights (Chapter 7).

The proposed single- and multi-person training datasets with their diverse appearance, pose, and camera viewpoint variation, together with the proposed transfer learning based training pipeline, has enabled the developed solutions to work in general scene settings. The proposed formulations for single- and multi-person settings are based on the insight of strongly linking pose estimation for each body part to its supporting image evidence, and selectively incorporating redundancy to enable occlusion handling. This results in accurate pose estimates, and allows the proposed solutions to handle scenarios with inter-personal or object occlusions. The formulations are designed to have a fixed inference cost regardless of the number of subjects in the scene. The inference cost is significantly reduced through insights into convolutional neural network architecture design, speeding up the costliest component of the system without any loss in accuracy, thus enabling real-time inference on consumer grade devices.

The methods developed not only enable a low-cost and instrumentation-free motion capture solution in diverse settings, but would also facilitate the current state-of-the art marker-based and marker-less multi-view solutions that are typically used in these settings.

Despite the wide range of contributions of the thesis towards enabling real-time monocular RGB based motion capture solutions, many questions remain unresolved, and many new avenues of investigation have arisen in the course of this thesis' work. The next section presents these in detail.

9.1 Outlook, Possible Improvements, and Future Work

9.1.1 Open Challenges With Regards to Training and Test Data

Despite the success of the approaches proposed with regards to data, there remain several challenges, which also present opportunities for future investigation.

Capturing Unseen Poses by Mining Motions From Videos: Even though the capture of the training dataset was designed to cover a diverse range of poses and motion, it still does not cover the full range of activities and motion which a motion capture system could be needed for. For instance, while capturing the dataset, it was not possible to cover all exercise or dance scenarios. One possible approach to acquire motion capture data is from videos, particularly of group activities, such as group exercise and group dances. In those scenarios, different individuals could be seen as different viewpoints of the same person, and treated instead as a multi-view scenario, combined with temporal consistency constraints from the video. This presents its own set of challenges, particularly with regards to motion synchronicity, but is an interesting and likely fruitful area of future investigation.

Representing True Interaction Scenarios: The compositing approach used for MuCo-3DHP generates plausible multi-person scenarios at scale. However, it does not produce true interaction scenarios. Capturing interpersonal interaction, particularly in cases with significant overlap and occlusion is challenging for marker based and markerless motion capture systems alike. One possible approach is to augment these motion capture systems with inertial measurement(Y. Huang et al. 2018), but even that brings with it its own set of challenges. On body sensors can slip and move under physical interpersonal interaction, leading to incorrect pose estimates. Further, due to the sheer number of possible interaction scenarios, it may not be possible to motion capture the complete set of possibilities. Physical simulation and rendering based approaches can be particularly helpful for these scenarios, with various sim-to-real approaches such as appearance augmentation and transfer learning providing a path to generalization to real world scenes.

Better Appearance Augmentation and Transfer Learning: The per-frame image space augmentation approach proposed and used for MPI-INF-3DHP and MuCo-3DHP is helpful for increasing the appearance diversity of the dataset, leading to better generalization to in-the-wild scenes. However, there are segmentation and augmentation artefacts still present which can be picked up by learning based approaches trained on the datasets. Though the transfer of low and mid level representations from in-the-wild 2D pose datasets ameliorates this to a large extent, improving the segmentation and augmentation quality could help further close the generalization gap. It would also make the augmentations useful for tasks beyond pose estimation, such as surface reconstruction, which are more sensitive to image space artefacts. Additionally, the textures applied for augmentation are only applied in the image space, and are neither consistent across views and nor over time. This prevents appearance diversification of the dataset for multi-view tasks, as well as for tasks that require learning of motion priors in the image space. Since MPI-INF-3DHP is accompanied by 3D scans of each subject, future approaches to register the scans to the multi-view data can enable appearance diversification by applying augmentations on the mesh surface.

Recent work (T. Han et al. 2019) indicates that frame-to-frame inconsistent augmentations could even be desirable at learning motion representations. Other work (Geirhos et al. 2018) has proposed to use image stylization as an augmentation approach for ensuring that the representations learned by neural network focus on the overall structure in the input image rather than local texture. Such an approach could be complementary to the augmentation approach proposed in the thesis, and could also help ameliorate the segmentation and augmentation artefacts present in the proposed dataset.

Additionally, the transfer learning proposed in the thesis is based on intuition and heuristics. Recent work (B. Tekin et al. 2017) has attempted to approach transfer learning in a more principled manner. Further investigations in this direction could be fruitful.

Better Annotations For Evaluation Benchmarks: The proposed multi-person evaluation benchmark MuPoTS-3D utilizes multi-view motion capture for 3D pose annotations. However, due to extensive interpersonal interaction and occlusion, the annotations required significant manual correction, and a large portion of the originally captured sequences had to be skipped owing to unreliable pose estimates, leaving only 20 sequences with $\approx 8k$ frames overall. Further, the quality of pose estimates for certain subjects in the selected sequences is unreliable, hence those subjects are excluded from the evaluation. The approaches proposed in the thesis, in combination with multi-view and temporal constraints could be leveraged to improve the ground truth pose annotation quality, and improve the evaluation benchmark.

9.1.2 Incorporating Additional Priors and Constraints For Better Pose Representation Learning

Temporal Reasoning: The representations proposed in this thesis are designed for per-frame inference. Motion capture applications however use one or more video streams as input. Hence, representations which leverage motion priors would lead to improved pose accuracy, particularly in cases of partial occlusion. There exist many challenges however. The design of motion priors needs to extend beyond the range of motions captured in the training datasets. Further, temporally consistent composites of multi-person scenes is challenging, particularly in cases where the camera relative depths of the subjects is similar and they overlap in the frame. Game engines (Fabbri et al. 2018) and other means of physical simulation can be a good alternative for generating temporally consistent multi-person interaction scenarios. Some approaches (Bugra Tekin et al. 2016b; Tome et al. 2017) attempted to build in temporal reasoning into pose inference, but the approaches are restricted to single-person scenarios.

Pose Inference Supported By Scene Layout (and vice versa): The use of a green screen background in the training dataset, as well as the application of randomly chosen images and textures for background augmentation leads to pose inference for each subject happening in isolation from its surroundings. While it allows the trained pose estimators to be scene appearance invariant, humans can use information of the scene layout to inform pose estimates. Also, ambiguities in scene layout estimation may be resolved in turn by observing how humans inhabit, move around, and interact with the physical space and objects. Recent work (Hassan et al. 2019; Monszpart et al. 2019) has explored this direction, and further investigation is warranted.

9.1.3 Extending Neural Network Architecture Insights

Implicit Sparsification: The results open several avenues of future investigation. Understanding why features learned with Adam (and perhaps other adaptive methods) are more selective than with (m)SGD can further shed light on the practical differences between adaptive methods and SGD. Also, the insights from the thesis will lead practitioners to be more aware of the implicit tradeoffs between network capacity and generalization being made below the surface, while changing hyperparameters such as mini-batch size, which are seemingly unrelated to network capacity.

Beyond ResNet-like Architectures For Inference on Edge Devices: At present, the most suc-

cessful network architectures for on device inference, such as MobileNetV1/V2 [2018; 2018] are based on variants of the ResNet architecture. Considering the inference speed gains achieved by the proposed SelecSLSNet over ResNet-50 at similar accuracy levels, mobile derivatives of SelecSLSNet could be investigated for faster alternatives to MobileNet designs.

9.1.4 Improving RGB Input Based Motion Capture

Better Identity Tracking: Tracking subject identities across frames (and viewpoints) is a complex problem, and is its own area of research. Approaches from work on person re-identification (Lavi et al. 2018) could be adapted for use in this case, but those introduce an additional compute cost into the system. Light weight facial feature identification approach could be used for sub-sequences where the subject's face is visible.

Foot Planting Constraints For Improved Camera Relative Localization: For scenarios with known ground plane geometry, detection of foot planting can improve pose estimates, as well as estimates of the relative location of the subject from the camera.

Multi-view Motion Capture: The approaches proposed in the thesis can be used to further improve multi-view markerless motion capture. Contemporary work (Iskakov et al. 2019; Joo et al. 2018) has leveraged successes in in-the-wild 2D pose estimation to improve multi-view motion capture. For the single person case Iskakov et al. 2019 use backprojection of heatmaps from each view into a volumetric space, and utilize 3D convolutions to predict the 3D pose. A similar volumetric approach could be used with 3D predictions from each view. 3D pose predictions from each view can also help match subjects across views in the case of multi-person scenarios, and construct such a volumetric representation per subject. The approach of Joo et al. 2018 works for multiple subjects, and requires matching of the detected 2D keypoints across frames. 3D pose predictions from each viewpoint can also help simplify matching across views for such and approach. Other ways of fusing the 3D pose predictions from multiple views could be explored as well.

Appendices

Appendix A

MPI-INF-3DHP Acquisition – Additional Details

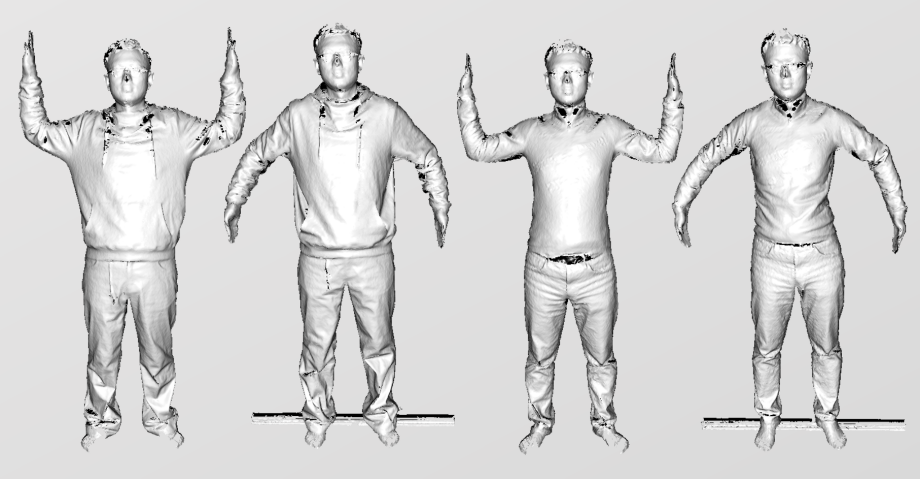

Figure A.1: Scans of one of the actors recorded in MPI-INF-3DHP training set. The scans are captured with both clothing sets worn by the actor during the capture, and with different articulations to later make rigging of the mesh easier.

A.1 Other Associated Data Captured

The recording setup also consists of one camera with a fisheye lens and one depth camera. Additionally, 3D scans of each subject in two different poses are captured, for each of the clothing sets worn by the subject. The intention is to future-proof the dataset, and make it useful for tasks beyond body pose estimation, such as body surface estimation. The scans are not textured, and are captured in different poses to make rigging of the mesh easier. Examples of the scans are shown in Figure A.1.

A.2 Prompts for Guiding the Actors

For each actor, with each clothing set, the recording session lasts 4 minutes, with 1 minute per activity or pose category type. Thus, for each actor, there are 8 minutes of recording, which cover 7 activities or pose categories. Both 4 minute sessions involve sitting poses as the third activity set, included to help the actors catch their breath inbetween intense activities. Camera flashes as well as claps are used after each activity set to provide additional cues for allowing the additional depth sensor to be synchronized with the motion capture system. The verbal prompts used to guide the actors through the activity sets are shown in Figure A.2. In addition to the verbal prompts, the actors are shown examples of the expected activities, and the prompts are only used to ensure that the actors keep time and cover all intended activities.

Walking / Jogging	**Exercise Scenarios**	**Sitting 1**	**Crouching/Reaching**
Exaggerated Walking, Jogging	Lunges	Eating	Crouch down and lift something from the floor
Walk with hands by the side	Wave arms while lunging	Working at a computer	Bend down from the waist to pick something, put it on a high shelf
Walk with hands held out in front	Push ups	Pretend to pick something from the floor, put it on a table	Tie shoe laces
Wait in a queue impatiently	Upside down bridge	Pretend to pick something from the table and put it on the floor	Pick something from a high shelf
Pretend phone call	Stretch legs, lean side to side	Lie back on the chair	Open a fridge and rummage inside
Point to things	Other exercises and stretches	Cross feet at ankles	Photography poses
Have an animated conversation		Move arms around in front and behind the torso	Crouch walk
Pretend cigarette, touch the face		Sit up straight with feet crossed the other way	
Walk with arms moving to and fro together		Interact with imaginary objects	
Lying Down	**Sports Scenarios**	**Sitting 2**	**Misc.**
Cycling with legs in the air	Boxing, with proper stance	Move the chair around while seated	Crazy body contortions and poses
Cross legs, wave arms around	Dodge punches, side to side	Wave someone over	Don't forget lower body articulation
Crunches	Soccer kick, dribble, dodge	Cheer for a sports team	Dancing
Side crunches	Tennis serve	Exaggerated conversation while seated	Jumping
Sit on the floor cross legged	Play difficult to reach shots in tennis	Cross legs at thighs	
Sit on the floor with legs straight in front	Basketball jump shot	Lean back in the chair with legs crossed	
Various poses with arms in front of the torso or behind head	Golf / Cricket hit	Cross legs the other way, and move arms around in front of and behind the torso	

Figure A.2: Verbal prompts used to guide the actors through the activity sets.

Appendix B

Towards In-the-Wild 3D Pose Estimation – Additional Details

B.1 Systematic Errors in Evaluation Due to Perspective Distortion

Approaches which use bounding-box crops around the subject for predicting the 3D pose are no longer considering the actual camera used to acquire the image. This new virtual camera is oriented towards the crop center and its field of view covers the crop area. Since cropped image pose networks only 'see' the cropped input, their predictions live in this rotated view. This bias is systematic, and can be compensated by computing the rotation matrix R that rotates the virtual camera to the original view.

A simplifying assumption can be made that much of the perspective distortion exists in the horizontal direction, due to a more limited range of motion of the subjects in the scene in the vertical direction. R is then the rotation around the camera up direction by the angle between the original and the virtual view direction, see Figure B.1. On MPI-INF-3DHP test set perspective correction improves the PCK by 3 percent points. On other datasets, such as HumanEva with a more limited field of view, there is still a consistent, albeit small, improvement of up to 3mm MPJPE.

Experimentally, using the vector from the camera origin to the centroid of 2D keypoints as the virtual view direction yielded better results. However, the crop center can also be used instead. Opposed to the Perspective-n-Point algorithm applied by Xiaowei Zhou et al. 2015c, any regression method

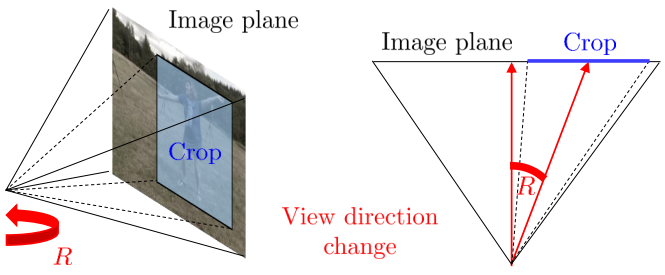

Figure B.1: Sketch of the input image cropping and resulting change of field of view. The corresponding rotation R of the view direction is sketched in 2D on the right.

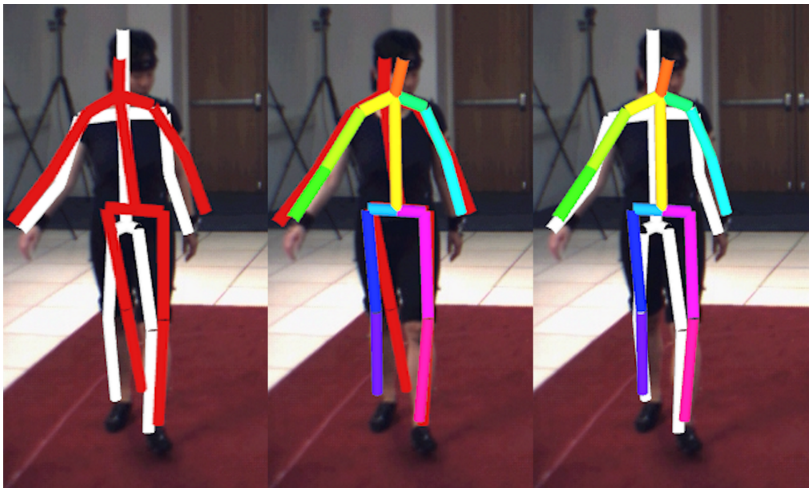

Figure B.2: The predicted pose (red) is inaccurate for positions away from the camera center (left), compared against the ground truth (white). Perspective correction (colored) corrects the orientation (center) and is closer to the ground truth (right). Here tested on the walking sequence of HumanEva [2010] S1.

that works on cropped images could immediately profit from this perspective correction, without computing 2D keypoint detections.

B.2 Architecture and Training Details

B.2.1 2D Pose Network (2DPoseNet) Architecture and Training Details

Network Architecture: A CNN architecture based on Resnet-101 (K. He et al. 2016a) (up to the filter banks before res5a) is used as the backbone. Since heatmaps are also output, striding is removed from `res5a`. Additionally, the number of features in the `res5a` module are halved, identity skip connections are removed from `res5b` and *res5c*, and the number of features gradually tapered to 15 (heatmaps for 14 joints + root). The heatmaps are upsampled $\times 2$ to be $1/8^{\text{th}}$ the width and height of the input.

The 2D pose network is fully convolutional and it is trained on MPII (Mykhaylo Andriluka et al. 2014) and LSP (Johnson et al. 2010, 2011), with images resized to 368×368 px.

Intermediate Supervision: Additionally, intermediate supervision is employed at `res4b20` and `res5a`, treating the first 15 feature maps of the layers as intermediate heatmaps H.

Training: Caffe (Jia et al. 2014) is used as the framework, with the AdaDelta [2012] solver with a momentum of 0.9 and weight decay rate of 0.005, using a batch size of 7, and L2 Loss everywhere. For the Learning Rate and Loss Weight taper schema, refer to Table B.1.

As shown in Table B.4, the 2D Pose estimation network approaches state of the art results, giving a PCK of **91.2%** and 65.3 AUC on the LSP test set. On the MPII Single Person test set it achieves a PCK of **89.7%** and 61.3 AUC.

Table B.1: Loss weight and learning rate, LR, taper scheme used for *2DPoseNet*. Heatmaps H_{4b20} and H_{5a} are used for intermediate supervision.

Base LR	# Iter	Loss Weights ($\mathbf{w} \times L(H_{xx})$)		
		H	H_{4b20}	H_{5a}
0.050	60k	1.0	0.5	0.5
0.010	60k	1.0	0.1	0.1
0.005	60k	1.0	0.05	0.05
0.001	60k	1.0	0.05	0.05
6.6e-4	60k	1.0	0.005	0.005
0.0001	40k	1.0	0.001	0.001
2.5e-5	40k	1.0	0.0001	0.0001
0.0008	60k	1.0	0.0001	0.0001
0.0001	40k	1.0	0.0001	0.0001
3.3e-5	20k	1.0	0.0001	0.0001

Table B.2: Loss weight and LR taper scheme used for *3DPoseNet*. There is a difference in the number of iterations used when training with Human3.6m or MPI-INF-3DHP alone, v.s. when training with the two in conjunction. Part Labels PL are used only when training with Human3.6m solely. X stands in for $\mathbf{P}^{\mathbf{3D}}$or $\mathbf{O}\mathit{1}^{\mathbf{3D}}$or $\mathbf{O}\mathit{2}^{\mathbf{3D}}$, 3D body joint positions expressed relative to the root, or first and second order kinematic parents.

Base LR	H3.6m/MPI-3DHP Batch = 5 #Epochs	H3.6m+MPI-3DHP Batch = 6 #Epochs	Loss Weights (w $\times$ L(A_{bb})) $X = P/O1/O2$				
			X_{4b5}	X_{4b20}	X	H	PL*
0.05	3 (45k)	2.4 (60k)	50	50	100	0.1	0.05
0.01	1 (15k)	1.2 (30k)	10	10	100	0.05	0.025
0.005	2 (30k)	1.2 (30k)	5	5	100	0.01	0.005
0.001	1 (15k)	0.6 (15k)	1	1	100	0.01	0.005
5e-4	2 (30k)	1.2 (30k)	0.5	0.5	100	0.005	0.001
1e-4	1 (15k)	0.6 (15k)	0.1	0.1	100	0.005	0.001

B.2.2 3D Pose Network (3DPoseNet) Training Details

For training, the solver settings are similar to *2DPoseNet*, and Euclidean Loss is used everywhere. For transfer learning, the learning rate of the transferred layers is scaled down by a factor determined by validation. For fine-tuning in the multi-modal fusion case, the learning rate of the trained network is downscaled by 10,000 with respect to the three new fully-connected layers. For the learning rate and loss weight taper schema for both the main training and multi-modal fusion fine-tuning stages, refer to Tables B.2 and B.3. Different training durations are employed with Human3.6m or MPI-INF-3DHP in isolation, versus when using both in conjunction. This is reflected in the aforementioned tables.

Table B.3: Loss weight and LR taper scheme used for fine-tuning *3DPoseNet* for Multi-modal Fusion scheme.

Base LR	H3.6m/MPI-3DHP Batch = 5 #Epochs	H3.6m+MPI-3DHP Batch = 6 #Epochs	Loss Weights (w × L(A_{bb})) P_{fused}
0.05	(1k)	(2k)	100
0.01	1 (15k)	0.8 (20k)	100
0.005	1 (15k)	0.8 (20k)	100
0.001	1 (15k)	0.8 (20k)	100

Table B.4: *2DPoseNet* on MPII Single Person Pose [2014] dataset and LSP [2010] 2D Pose datasets. * = Trained/Finetuned only on the corresponding training set

	MPII		LSP	
	$PCK_{h0.5}$	AUC	$PCK_{0.2}$	AUC
2DPoseNet				
2DPoseNet	89.6	61.5	**91.2**	65.5
Newell et al. 2016	**90.9***	**62.9***	-	-
Bulat et al. 2016	89.7*	59.6*	90.7	-
S.-E. Wei et al. 2016	88.5	61.4	90.5	65.4
Insafutdinov et al. 2016	88.5	60.8	90.1	**66.1**
Gkioxari et al. 2016	86.1*	57.3*	-	-
Lifshitz et al. 2016	85.0	56.8	84.2	-
Belagiannis et al. 2016	83.9*	55.5*	85.1	-
Pishchulin et al. 2016	82.4	56.5	87.1	63.5
P. Hu et al. 2016	82.4*	51.1*	-	-
Carreira et al. 2016	81.3*	49.1*	72.5*	-

B.3 Domain Adaptation To In The Wild 2D Pose Data

For the domain adaptation experiment, a network branch comprised of $conv_{3\times3,256}$, $conv_{3\times3,128}$, fc_{64} and fc_1 layers is used, along with cross entropy domain classification loss. Domain adaptation is done using the gradient inversion approach of Ganin et al. 2015. The domain adaptation branch is attached after *res*4*b*22 in the network. It was experimentally observed that directly starting out with $\lambda = -1$ performs better than gradually increasing the magnitude of λ with increasing iterations. The network is trained on the Human3.6m training set, with 2D heatmap and part label predictions as auxiliary tasks. Images from MPII Mykhaylo Andriluka et al. 2014 and LSP Johnson et al. 2010, 2011 training sets are used without annotations for learning better generalizable features. The generalizability is improved, as evidenced by the 41.4 3DPCK on MPI-INF-3DHP test set, but does not match up with the 66.5 3DPCK attained using transfer learning.

Appendix C

Occlusion Robust Pose Maps – Additional Details and Results

C.1 ORPM Pose Read-out Process

An algorithmic description of the read-out process is provided in Alg. 1.

C.2 Joint-wise Analysis

Figure C.1 shows joint-wise accuracy comparison of ORPM based inference with LCR-net [2017] on the single person MPI-INF-3DHP test set. For limb joints (elbow, wrist, knee, ankle) LCR-net performs comparably or better than torso-only read-out from ORPMs, but the full read-out performs significantly better.

Figure C.2 shows joint-wise accuracy comparison of the ORPM approach with LCR-net on MuPoTS-3D, the proposed multi-person 3D pose test set. We see that the ORPM based approach obtains a better accuracy for all joint types for most sequences, only performing worse than LCR-net for a select few joint types on certain sequences (TestSeq18,19,20).

C.3 Evaluation on Single-person Test Sets

Evaluation on Human3.6m is in Table C.1, and on MPI-INF-3DHP test set in Table C.2. The location-map formulation developed in Chapter 5 is also compared to the ORPM formulation by training with the same setup, which includes the 2D pretraining on MSCOCO, and the same 3D pose samples between the two formulations.

Table C.3 provides a sequencewise breakdown for the synthetic occlusion experiment on MPI-INF-3DHP test set wherein through randomly placed occlusions $\approx$14% of the joints are occluded. This doesn't account for self-occlusions.

Table C.1: Comparison of results on Human3.6m Ionescu et al. 2014b, for single un-occluded person. Human3.6m, subjects 1,5,6,7,8 used for training. Subjects 9 and 11, all cameras used for testing. Mean Per Joint Postion Error reported in mm

	Direct	**Disc.**	**Eat**	**Greet**	**Phone**	**Pose**	**Purch.**	**Sit.**
Pavlakos et al. 2017	60.9	67.1	61.8	62.8	67.5	58.8	64.4	79.8
[2017]	52.5	63.8	55.4	62.3	71.8	52.6	72.2	86.2
Tome et al. 2017	65.0	73.5	76.8	86.4	86.3	69.0	74.8	110.2
C.-H. Chen et al. 2017	89.9	97.6	90.0	107.9	107.3	93.6	136.1	133.1
Moreno-Noguer 2017	67.5	79.0	76.5	83.1	97.4	74.6	72.0	102.4
Xingyi Zhou et al. 2017	54.8	60.7	58.2	71.4	62.0	53.8	55.6	75.2
Martinez et al. 2017	51.8	56.2	58.1	59.0	69.5	55.2	58.1	74.0
B. Tekin et al. 2017	53.9	62.2	61.5	66.2	80.1	64.6	83.2	70.9
Xiaohan Nie et al. 2017	62.8	69.2	79.6	78.8	80.8	72.5	73.9	96.1
Location-map [2017]	62.6	78.1	63.4	72.5	88.3	63.1	74.8	106.6
LCR-net (Gregory Rogez et al. 2017)	76.2	80.2	75.8	83.3	92.2	79.9	71.7	105.9
Location-map (with setup from Ch 6)	65.52	78.8	64.8	75.0	85.2	66.4	88.1	110.2
ORPM Single-Person	58.2	67.3	61.2	65.7	75.82	62.2	64.6	82.0

	Sit Down	**Smk.**	**Photo**	**Wait**	**Walk**	**Walk Dog**	**Walk Pair**	**Avg.**
Pavlakos et al. 2017	92.9	67.0	72.3	70.0	54.0	71.0	57.6	67.1
[2017]	120.6	66.0	79.8	64.0	48.9	76.8	53.7	68.6
Tome et al. 2017	173.9	84.9	110.7	85.8	71.4	86.3	73.1	88.4
C.-H. Chen et al. 2017	240.1	106.6	139.2	106.2	87.0	114.0	90.5	114.2
Moreno-Noguer 2017	116.7	87.7	100.4	94.6	75.2	87.8	74.9	85.6
Xingyi Zhou et al. 2017	111.6	64.1	65.5	66.0	63.2	51.4	55.3	64.9
Martinez et al. 2017	94.6	62.3	78.4	59.1	49.5	65.1	52.4	62.9
B. Tekin et al. 2017	107.9	70.4	79.4	68.0	52.8	77.8	63.1	70.8
Xiaohan Nie et al. 2017	106.9	88.0	86.9	70.7	71.9	76.5	73.2	79.5
Location-map [2017]	138.7	78.8	93.8	73.9	55.8	82.0	59.6	80.5
LCR-net (Gregory Rogez et al. 2017)	127.1	88.0	105.7	83.7	64.9	86.6	84.0	87.7
Location-map (with setup from Ch 6)	155.9	82.0	95.2	76.8	59.7	94.1	64.3	84.3
ORPM Single-Person	93.0	68.8	84.5	65.1	57.6	72.0	63.6	69.9

Table C.2: Comparison of ORPM formulaiton against the state of the art on single person MPI-INF-3DHP test set. All evaluations use ground-truth bounding box crops around the subject, and *Percentage of Correct Keypoints measure in 3D* (@150mm), and the Area Under the Curve are reported, as described in Chapter 3. Additionally, the Mean Per Joint Position Error (mm) is also reported. Higher PCK and AUC is better, and lower MPJPE is better.

Network	Stand/ Walk	Exercise	Sit On Chair	Crouch/ Reach	On the Floor	Sports	Misc.	Total		
	PCK	PCK	PCK	PCK	PCK	PCK	PCK	PCK	AUC	MPJPE(mm)
Location-maps [2017]	88.8	79.7	76.3	75.3	51.4	89.1	80.9	78.1	42.0	119.2
LCR-net (Gregory Rogez et al. 2017)	70.5	56.3	58.5	69.4	39.6	57.7	57.6	59.7	27.6	158.4
Xingyi Zhou et al. 2017	85.4	71.0	60.7	71.4	37.8	70.9	74.4	69.2	32.5	137.1
D. Mehta et al. 2017a	89.5	77.1	75.3	74.5	52.4	87.1	80.5	77.5	41.1	113.1
ORPM Single-Person (Torso)	75.0	64.8	69.4	69.9	48.9	71.6	62.2	66.4	32.9	141.2
ORPM Single-Person (Full)	83.9	75.4	79.1	77.9	55.7	82.3	74.5	76.2	38.3	120.5
ORPM Multi-Person (Torso)	74.2	65.1	66.8	66.9	46.5	69.5	62.4	65.0	31.8	144.7
ORPM Multi-Person (Full)	82.6	74.3	76.0	73.2	52.1	80.6	74.2	74.1	36.7	125.1

Table C.3: Testing occlusion robustness of ORPMs through synthetic occlusions on MPI-INF-3DHP single person test set. The synthetic occlusions cover about 14% of the evaluated joints overall. The overall *Percentage of Correct Keypoints measure in 3D* (@150mm) is reported, as well as split by occlusion.

	Seq1	Seq2	Seq3	Seq4	Seq5	Seq6	Total
	PCK	PCK	PCK	PCK	PCK	PCK	PCK
Overall							
ORPM Multi-Person	78.7	70.0	71.9	65.2	61.4	60.7	69.0
ORPM Single-Person	80.9	72.8	72.6	65.7	62.5	65.8	71.1
Location-map D. Mehta et al. 2017b	80.1	72.4	72.4	61.5	50.2	69.8	69.4
Location-map (train. setup from Chapter 6)	79.3	74.4	72.2	67.2	55.7	64.6	70.4
Occluded Subset of Joints							
ORPM Multi-Person	73.3	66.5	55.0	56.5	45.1	64.9	62.8
ORPM Single-Person	74.9	63.2	59.0	54.2	48.0	68.4	64.0
Location-map D. Mehta et al. 2017b	61.4	54.5	47.6	36.4	30.5	66.2	53.2
Location-map (train. setup from Chapter 6)	69.6	61.9	49.0	50.8	43.5	63.4	59.2
Un-occluded Subset of Joints							
ORPM Multi-Person	79.9	70.5	73.7	66.2	64.6	59.5	70.0
ORPM Single-Person	82.1	74.0	74.1	67.0	65.3	65.1	72.2
Location-map D. Mehta et al. 2017b	83.9	74.6	75.0	64.4	54.0	70.9	72.1
Location-map (train. setup from Chapter 6)	81.3	76.0	74.6	69.0	58.1	64.8	72.2

Algorithm 1 3D Pose Inference

Given: $\mathcal{P}^{2D}$, $\mathcal{C}^{2D}$, $\mathcal{M}$
for all $i \in (1..m)$ **do**
  if $\mathbf{C}_i^{2D}[k] > thresh, k \in \{pelvis, neck\}$ **then**
    Person i is detected
    for all joints $j \in (1..n)$ **do**
      $rloc = \mathbf{P}_i^{2D}[k]$
      $\mathbf{P}_i[:, j]$ = READLOCMAP(j, rloc)
    end for
    for all limbs $l \in \{arm_l, arm_r, leg_l, leg_r, head\}$ **do**
      for $j =$ GETEXTREMITY$(l); j \notin \{pelvis, neck\}; j = parent(j)$ **do**
        if ISVALIDREADOUTLOC(i, j) **then**
          REFINELIMB(l, $\mathbf{P}_i^{2D}[j]$)
          break
        end if
      end for
    end for
  else
    No person detected
  end if
end for
function GETEXTREMITY(limb l)
  if $l = leg_s$ **then return** $ankle_s$
  else
    if $l = arm_s$ **then return** $wrist_s$
    else return $head$
    end if
  end if
end function
function READLOCMAP(joint j, 2DLocation rloc)
  $rloc = rloc/locMap\_scale\_factor$
  return $\mathbf{M}_j[rloc]$
end function
function REFINELIMB(limb l, 2DLocation rloc)
  for all joints $b \in$ limb l **do**
    $\mathbf{P}_i[:, b]$=READLOCMAP(b, rloc)
  end for
end function
function ISVALIDREADOUTLOC(person i, joint j)
  if $(\mathbf{C}_i^{2D}[j] > 0)$ **then**
    return ISISOLATED(i,j)
  else
    return 0
  end if
end function
function ISISOLATED(person i, joint j)
  $isol = 1$
  for all persons $\bar{i} \in (1..m), \bar{i} \neq i$ **do**
    for all 2DLocations $a \in \rho_{\bar{i}}(j)$ **do**
      if $||a - \mathbf{P}_i^{2D}[j]||_2 < isoThresh$ **then**
        $isol = 0$
        break
      end if
    end for
  end for
  return $isol$
end function

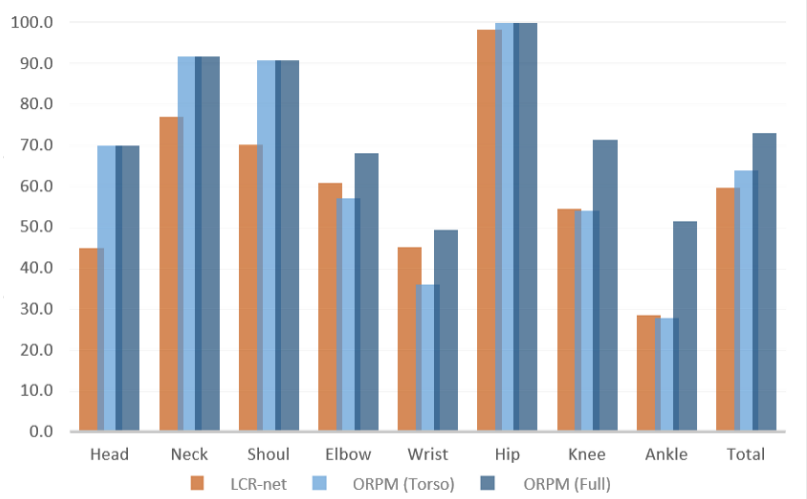

Figure C.1: Joint-wise accuracy comparison of ORPM based inference and LCR-net [2017] on the single person MPI-INF-3DHP test set. 3D Percentage of Correct Keypoints (@150mm) as the vertical axis. LCR-net predictions were mapped to the ground truth bone lengths for fairness of comparison.

ORPM

	Head	Neck	Shoul	Elbow	Wrist	Hip	Knee	Ankle	Total
TestSeq1	96.8	100.0	96.6	78.0	50.9	99.8	81.0	62.3	81.0
TestSeq2	63.9	85.9	68.3	54.9	47.6	75.6	55.3	42.8	59.9
TestSeq3	79.9	91.9	90.5	56.5	46.8	98.9	42.0	30.2	64.4
TestSeq4	73.1	82.7	76.5	56.6	49.3	97.0	50.7	31.4	62.8
TestSeq5	56.5	82.0	79.7	66.2	65.7	85.5	67.3	42.0	68.0
TestSeq6	6.5	32.1	35.5	12.7	10.9	99.4	23.3	10.8	30.3
TestSeq7	66.6	97.8	81.5	47.0	21.0	98.7	63.0	61.9	65.0
TestSeq8	55.2	71.6	65.8	57.2	45.2	96.1	44.4	42.8	59.2
TestSeq9	67.2	84.1	81.5	30.1	25.7	100.0	62.7	73.1	64.1
TestSeq10	98.2	100.0	100.0	63.2	52.1	100.0	95.4	77.8	83.9
TestSeq11	66.0	92.7	84.1	56.0	44.1	89.2	73.8	43.9	67.2
TestSeq12	46.1	73.1	76.2	73.4	66.8	97.1	64.4	40.8	68.3
TestSeq13	58.5	77.9	74.5	48.6	38.5	84.0	68.9	41.3	60.6
TestSeq14	47.5	73.3	69.7	43.8	38.4	79.8	62.1	41.0	56.5
TestSeq15	62.3	91.2	84.7	58.3	42.6	97.1	77.4	52.3	69.9
TestSeq16	72.9	87.8	86.1	82.3	80.1	92.9	81.9	51.9	79.4
TestSeq17	74.4	73.8	78.0	78.1	61.3	96.5	91.0	78.6	79.6
TestSeq18	54.8	73.8	77.1	73.1	44.2	87.6	74.6	41.5	66.1
TestSeq19	44.9	78.4	79.4	55.2	54.1	84.4	77.7	37.9	64.3
TestSeq20	50.8	73.5	71.7	62.5	58.6	73.0	69.0	47.5	63.5

LCR-net

	Head	Neck	Shoul	Elbow	Wrist	Hip	Knee	Ankle	Total
TestSeq1	73.1	81.8	74.4	61.3	41.9	97.0	74.9	47.1	67.7
TestSeq2	54.6	69.3	53.9	43.5	31.7	69.3	48.6	39.3	49.8
TestSeq3	71.2	81.0	56.4	35.8	29.6	95.1	49.1	31.7	53.4
TestSeq4	57.6	80.7	63.2	55.1	48.3	94.7	52.0	31.3	59.1
TestSeq5	68.0	84.9	72.1	70.5	60.5	84.7	65.1	43.2	67.5
TestSeq6	6.3	26.8	18.6	17.1	12.8	84.8	7.6	2.3	22.8
TestSeq7	34.4	66.8	38.8	31.5	22.9	87.6	49.1	25.6	43.7
TestSeq8	56.3	70.0	52.1	44.6	35.4	75.7	46.8	31.7	49.9
TestSeq9	34.9	41.9	34.1	20.0	15.9	45.1	32.8	31.4	31.1
TestSeq10	80.5	97.6	98.0	53.1	31.9	100.0	87.0	87.9	78.1
TestSeq11	23.3	43.0	39.4	61.5	44.1	88.2	57.4	27.4	50.2
TestSeq12	24.2	41.5	39.2	61.1	65.1	97.1	41.2	20.6	51.0
TestSeq13	42.0	62.2	51.5	48.2	37.4	78.5	57.1	36.5	51.6
TestSeq14	36.6	63.2	50.7	39.9	29.2	80.7	57.5	37.4	49.3
TestSeq15	39.4	72.8	59.1	44.6	34.4	91.4	73.6	34.6	56.2
TestSeq16	48.1	67.8	65.0	78.6	68.3	93.1	67.0	35.4	66.5
TestSeq17	44.1	77.7	75.7	68.8	58.9	85.4	59.7	46.8	65.2
TestSeq18	53.7	89.9	83.5	63.6	42.5	89.9	60.8	28.4	62.9
TestSeq19	69.6	80.0	67.2	62.4	53.0	81.1	73.9	50.2	66.1
TestSeq20	70.5	48.8	49.5	67.3	61.9	72.6	64.4	38.0	59.1

Figure C.2: Comparison of ORPM based inference and LCR-net [2017] on MuPoTS-3D, the proposed multi-person test set. Here a joint-wise breakdown of PCK for all 20 sequences is visualized. LCR-net predictions were mapped to the ground truth bone lengths for fairness of comparison.

Appendix D

On Implicit Filter Level Sparsity In Convolutional Neural Networks

Table D.1: Layerwise % filters pruned from BasicNet trained on CIFAR10 and CIFAR100, based on the $|\gamma| < 10^{-3}$ criteria. Also shown are pre-pruning and post-pruning test error, and the % of *convolutional* parameters pruned. C1-C7 indicate Convolution layer 1-7, and the numbers in parantheses indicate the total number of features per layer. Average of 3 runs.

CIFAR10					% Sparsity by γ or % Filters Pruned								% Param	
		Train Loss	Test Loss	Test Err	C1 (64)	C2 (128)	C3 (128)	C4 (256)	C5 (256)	C6 (512)	C7 (512)	Total (1856)	Pruned (4649664)	Pruned Test Err.
Adam	L2: 1e-3	0.29	0.41	13.1	59	57	42	74	76	97	98	83	97	13.5
	L2: 1e-4	0.06	0.43	10.5	44	22	6	45	54	96	95	70	90	10.5
	WD: 2e-4	0.22	0.42	13.4	57	27	9	19	46	77	91	60	83	13.4
	WD: 1e-4	0.07	0.42	11.2	45	4	0	0	14	51	78	40	63	11.2
SGD	L2: 1e-3	0.62	0.64	21.8	86	61	53	46	65	4	0	27	38	21.8
	L2: 5e-4	0.38	0.49	16.3	68	16	9	9	24	0	0	9	13	16.5
	WD: 1e-3	0.61	0.63	21.6	85	60	51	46	66	4	0	27	38	21.6
	WD: 5e-4	0.38	0.46	15.8	69	19	7	7	23	0	0	8	13	16.1

CIFAR100					% Sparsity by γ or % Filters Pruned								% Param	
		Train Loss	Test Loss	Test Err	C1 (64)	C2 (128)	C3 (128)	C4 (256)	C5 (256)	C6 (512)	C7 (512)	Total (1856)	Pruned (4649664)	Pruned Test Err.
Adam	L2: 1e-3	1.06	1.41	39.0	56	47	43	68	72	91	85	76	95	39.3
	L2: 1e-4	0.10	1.98	36.6	41	20	9	33	34	67	55	46	74	36.6
	WD: 2e-4	0.34	1.56	37.3	55	20	3	4	2	16	26	16	27	37.3
	WD: 1e-4	0.08	1.76	36.2	38	4	0	0	0	0	5	3	4	36.2
SGD	L2: 1e-3	1.49	1.78	47.1	82	41	33	29	33	6	18	23	34	47.1
	L2: 5e-4	0.89	1.69	42.1	64	3	3	3	2	0	2	4	4	42.1
	WD: 1e-3	1.49	1.79	47.6	82	43	31	28	33	6	17	23	34	47.6
	WD: 5e-4	0.89	1.69	41.9	66	2	1	4	2	0	1	4	4	41.9

Here additional experiments are provided that show how filter level sparsity manifests under different gradient descent flavours and regularization settings, and that it even manifests with Leaky ReLU.

Table D.2: Layerwise % filters pruned from BasicNet trained on CIFAR100, based on the $|\gamma| < 10^{-3}$ criteria. Also shown are pre-pruning and post-pruning test error. C1-C7 indicate Convolution layer 1-7, and the numbers in parantheses indicate the total number of features per layer. Average of 3 runs.

Adam vs AMSGrad (ReLU)					% Sparsity by γ or % Filters Pruned								
		Train Loss	Test Loss	Test Err	C1 (64)	C2 (128)	C3 (128)	C4 (256)	C5 (256)	C6 (512)	C7 (512)	Total (1856)	Pruned Test Err.
Adam	L2: 1e-3	1.06	1.41	39.0	56	47	43	68	72	91	85	76	39.3
	L2: 1e-4	0.10	1.98	36.6	41	20	9	33	34	67	55	47	36.6
AMSGrad	L2: 1e-2	3.01	2.87	71.9	79	91	91	96	96	98	96	95	71.9
	L2: 1e-4	0.04	1.90	35.6	0	0	0	0	1	25	23	13	35.6
	L2: 1e-6	0.01	3.23	40.2	0	0	0	0	0	0	0	0	40.2

Adam With Leaky ReLU				% Sparsity by γ or % Filters Pruned								
NegSlope=0.01	Train Loss	Test Loss	Test Err	C1 (64)	C2 (128)	C3 (128)	C4 (256)	C5 (256)	C6 (512)	C7 (512)	Total (1856)	Pruned Test Err.
L2: 1e-3	1.07	1.41	39.1	49	40	39	62	61	81	85	70	39.4
L2: 1e-4	0.10	1.99	36.8	33	20	9	31	29	55	53	41	36.8
NegSlope=0.1												
L2: 1e-4	0.14	2.01	37.2	38	30	21	34	31	55	52	43	37.3

The emergence of feature selectivity in Adam in multiple layers is further discusses, along with its implications on the extent of sparsity (Section D.2). Section D.3 considers additional hyperparameters that influence the emergent sparsity. Section D.4 provides specifics for some of the experiments reported in Chapter 7.

D.1 Layer-wise Sparsity in *BasicNet*

Table D.1 shows that with BasicNet on CIFAR-10 and CIFAR-100, Adam shows feature sparsity in both early layers and later layers, while SGD only shows sparsity in the early layers. It is established in the main paper that Adam learns selective features in the later layers which contribute to this additional sparsity.

Sparsity with AMSGrad: In Table D.2 the extent of sparsity of Adam is compared with that of AMSGrad (Reddi et al. 2018). Given that AMSGrad tracks the long term history of squared gradients, the effect of L2 regularization is expected to be dampened in the low gradient regime, and should lead to less sparsity. For BasicNet, on CIFAR-100, with L2 regularization of 10^{-4}, AMSGrad only shows sparsity in the later layers, and overall only 13% of features are inactive. For a comparable test error for Adam, 47% of the features are inactive.

Sparsity with Leaky ReLU: Leaky ReLU is anecdotally [] believed to address the 'dying ReLU' problem by preventing features from being inactivated. The cause of feature level sparsity is believed to be the accidental inactivation of features, which gradients from Leaky ReLU can help revive. Chapter 7 however points at systemic processes underlying the emergence of feature level sparsity, and those would continue to persist even with Leaky ReLU. Though the original definition of feature selectivity does not apply here, it can be modified to make a distinction between data points which produce positive activations for a feature vs. the data points that produce a negative activation. For

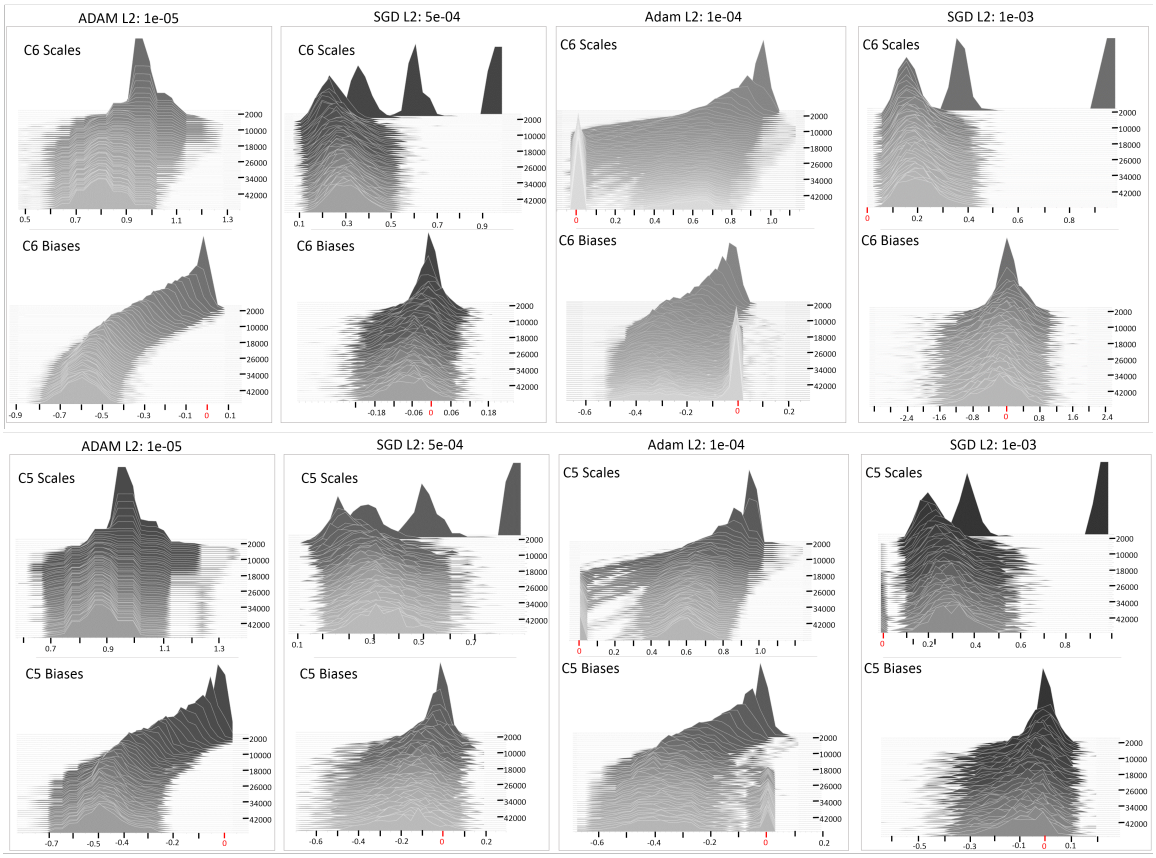

Figure D.1: Emergence of Feature Selectivity with Adam (Layer C6 and C5) The evolution of the learned scales (γ, top row) and biases (β, bottom row) for layer C6 (top) and C5 (bottom) of *BasicNet* for Adam and SGD as training progresses, in both low and high L2 regularization regimes. Adam has distinctly negative biases, while SGD sees both positive and negative biases. For positive scale values, as seen for both Adam and SGD, this translates to greater feature selectivity in the case of Adam, which translates to a higher degree of sparsification when stronger regularization is used.

typical values of the negative slope (0.01 or 0.1) of Leaky ReLU, the more selective features (as per the updated definition) would continue to see lower gradients than the less selective features, and would consequently see relatively higher effect of regularization. For BasicNet trained on CIFAR-100 with Adam, Table D.2 shows that using Leaky ReLU has a minor overall impact on the emergent sparsity. See Section D.3 for more effective ways of reducing filter level sparsity in ReLU networks.

D.2 On Feature Selectivity in Adam

Figure D.1 shows the distribution of the scales (γ) and biases (β) of layers C6 and C5 of *BasicNet*, trained on CIFAR-100. SGD and Adam are considered here, each with a low and high regularization value. For both C6 and C5, Adam learns exclusively negative biases and positive scales, which results in features having a higher degree of selectivity (i.e, activating for only small subsets of the training corpus). In case of SGD, a subset of features learns positive biases, indicating more universal (less selective) features.

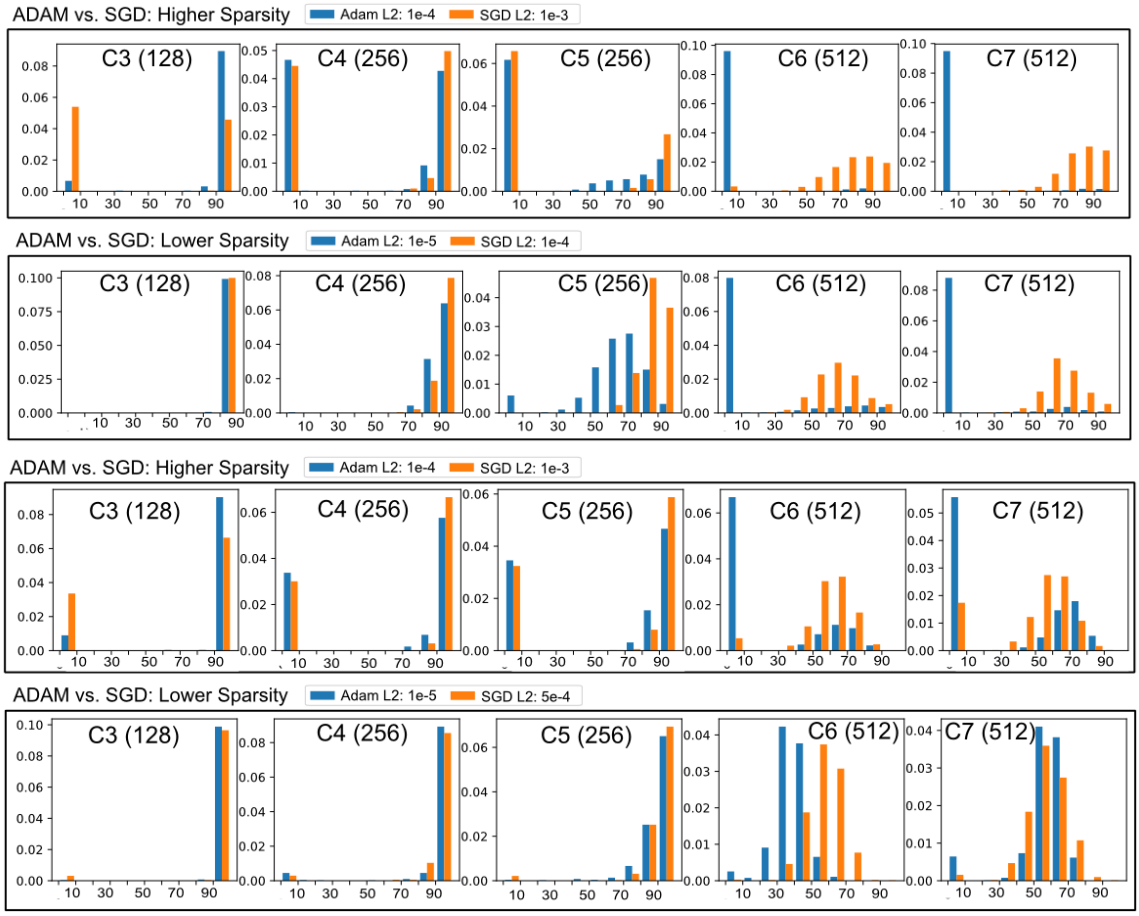

Figure D.2: Layer-wise Feature Selectivity Feature universality for CIFAR 10 (top) and CIFAR-100 (bottom), with Adam and SGD. X-axis shows the universality and Y-axis ($\times 10$) shows the fraction of features with that level of universality. For later layers, Adam tends to learn less universal features than SGD, which get pruned by the regularizer. Please be mindful of the differences in Y-axis scales between plots.

Figure D.2 shows feature selectivity also emerges in the later layers when trained on CIFAR-10, in agreement with the results presented for CIFAR-100 in Fig. 3 of the main paper.

D.3 Effect of Other Hyperparameters on Sparsity

Having shown in Chapter 7 and in Sec. D.2 that feature selectivity results directly from negative bias (β) values when the scale values (γ) are positive, the effect of β initialization value on the resulting sparsity is investigated here. As shown in Table D.3 for BasicNet trained with Adam on CIFAR 100, a slightly negative initialization value of -0.1 does not affect the level of sparsity. However, a positive initialization value of 1.0 results in higher sparsity. This shows that attempting to address the emergent sparsity by changing the initialization of β may be counter productive.

The effect of scaling down the learning rate of γ and β compared to that for the rest of the network (Table D.3) is also investigated. Scaling down the learning rate of γs by a factor of 10 results in

Table D.3: Layerwise % filters pruned from BasicNet trained on CIFAR100, based on the $|\gamma| < 10^{-3}$ criteria. Also shown are pre-pruning and post-pruning test error. C1-C7 indicate Convolution layer 1-7, and the numbers in parantheses indicate the total number of features per layer. The effect of different initializations of βs, as well as the effect of different relative learning rates for γs and βs on the emergent sparsity is studied, when trained with Adam with L2 regularization of 10^{-4}. Average of 3 runs.

				% Sparsity by γ or % Filters Pruned								
	Train Loss	Test Loss	Test Err	C1 (64)	C2 (128)	C3 (128)	C4 (256)	C5 (256)	C6 (512)	C7 (512)	Total (1856)	Pruned Test Err.
Baseline (γ_{init}=1, β_{init}=0)	0.10	1.98	36.6	41	20	9	33	34	67	55	46	36.6
γ_{init}=1, β_{init}=−0.1	0.10	1.98	37.2	44	20	10	34	32	68	54	46	36.5
γ_{init}=1, β_{init}=1.0	0.14	2.04	38.4	47	29	25	36	46	69	61	53	38.4
Different Learning Rate Scaling for β and γ												
LR scale for γ: 0.1	0.08	1.90	35.0	16	6	1	13	20	52	49	33	35.0
LR scale for β: 0.1	0.12	1.98	37.1	42	26	21	41	48	70	55	51	37.1

a significant reduction of sparsity. This can likely be attributed to the decrease in effect of the L2 regularizer in the low gradient regime because it is directly scaled by the learning rate. This shows that tuning the learning of γ can be more effective than Leaky ReLU at controlling the emergent sparsity. On the other hand, scaling down the learning rate of βs by a factor of 10 results in a slight increase in the extent of sparsity.

D.4 Experimental Details

For all experiments, the learned BatchNorm scales (γ) are initialized with a value of 1, and the biases (β) with a value of 0. The reported numbers for all experiments on CIFAR10/100 are averaged over 3 runs. Those on TinyImageNet are averaged over 2 runs, and for ImageNet the results are from 1 run. On CIFAR10/100, VGG-16 follows the same learning rate schedule as *BasicNet*, as detailed in Section 2.1 in the main paper.

For experiments on ObjectNet3D (Xiang et al. 2016) renderings, objects from the following 30 classes are used: aeroplane, bed, bench, bicycle, boat, bookshelf, bus, camera, chair, clock, eyeglasses, fan, flashlight, guitar, headphone, jar, kettle, keyboard, laptop, piano, racket, shoe, sofa, suitcase, teapot, toaster, train, trophy, tub, and wheelchair. The objects are rendered to 64x64 pixel images by randomly sampling (uniformly) the azimuth angle between -180 and 180 degrees, and the elevation between -15 and +45 degrees. The renderings are identical between the cluttered and the plain set, with the backgrounds for the cluttered set taken from the Cubism subset from PeopleArt (Wen et al. 2016) dataset. See Figure D.3. The network structure and training is similar to that for CIFAR10/100, and a batch size of 40 is used.

On TinyImageNet, both VGG-16 and BasicNet follow similar schemes. Using a mini-batch size of 40, the gradient descent method specific base learning rate is used for 250 epochs, and scaled down by 10 for an additional 75 epochs and further scaled down by 10 for an additional 75 epochs, totaling 400 epochs. When the mini-batch size is adjusted, the number of epochs are appropriately adjusted to ensure the same number of iterations.

On ImageNet, the base learning rate for Adam is 1e-4. For *BasicNet*, with a mini-batch size of 64,

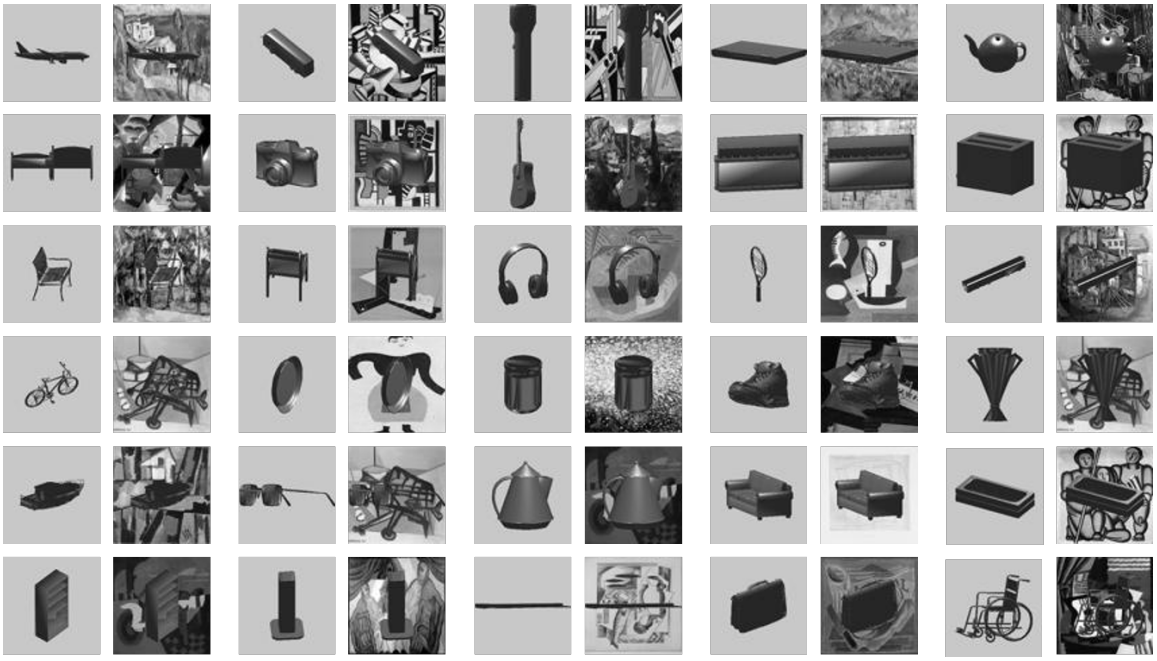

Figure D.3: Unaugmented and augmented renderings of the subset of 30 classes from ObjectNet3D (Xiang et al. 2016) employed to gauge the effect of task difficulty on implicit filter sparsity. The rendered images are 64x64 and obtained by randomly sampling (uniformly) the azimuth angle between -180 and 180 degrees, and the elevation between -15 and +45 degrees. The renderings are identical between the augmented and the unaugmented set and only differ in the background. The background images are grayscale versions of the Cubism subset from PeopleArt (Wen et al. 2016) dataset.

the base learning rate is used for 15 epochs, scaled down by a factor of 10 for another 15 epochs, and further scaled down by a factor of 10 for 10 additional epochs, totaling 40 epochs. The epochs are adjusted with a changing mini-batch size. For VGG-11, with a mini-batch size of 60, the total epochs are 60, with learning rate transitions at epoch 30 and epoch 50. For VGG-16, mini-batch size of 40, the total number of epochs are 50, with learning rate transitions at epoch 20 and 40.

D.5 Sparsity on Tasks Beyond Image Classification

The mechanism underlying the implicit sparsification of filters continues to be at play for tasks other than image classification. Figure D.4 shows a layer-wise breakdown of the emergent sparsity for ResNet-50 trained for multi-person 3D pose estimation as described in Chapter 6.3.

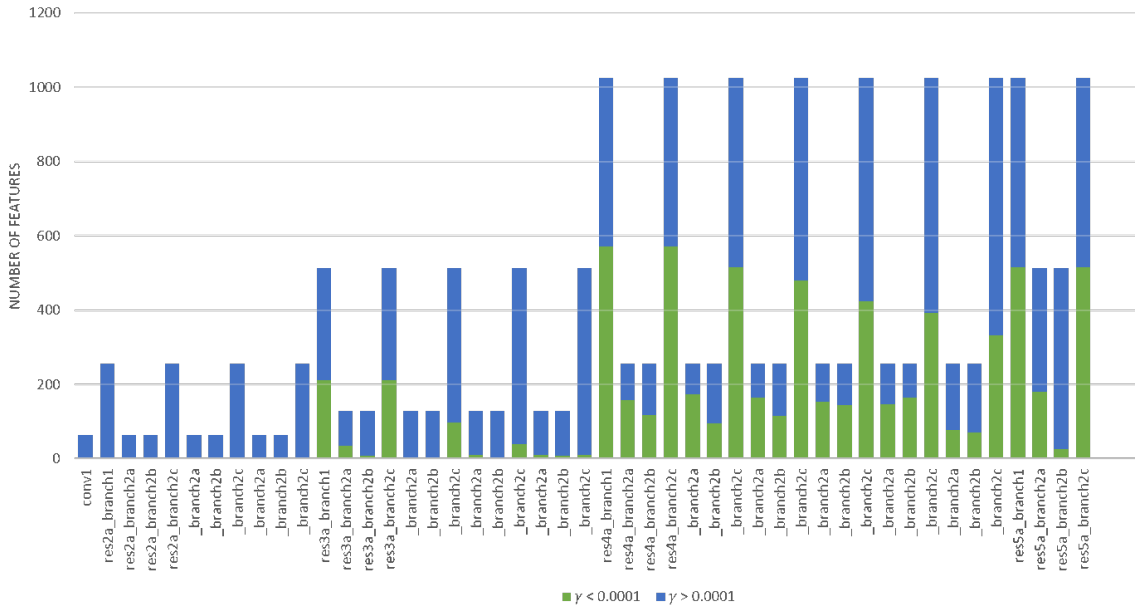

Figure D.4: Visualization of the layerwise sparsity in ResNet-50 trained for the task of multi-person 3D body pose estimation (Chapter 6.3). The network is trained with AdaDelta (Zeiler 2012), and the extent of sparsity is shown using the absolute value of BatchNorm learned scale γ. As with image classification, significant filter sparsity emerges on other tasks as well, when training with adaptive gradient descent methods.

Own Work

Mehta, Dushyant, Kwang In Kim, and Christian Theobalt (2019). "On Implicit Filter Level Sparsity In Convolutional Neural Networks". In: *32nd IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2019)*. IEEE.

Mehta, Dushyant, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt (2017a). "Monocular 3D Human Pose Estimation In The Wild Using Improved CNN Supervision". In: *3D Vision (3DV), 2017 Fifth International Conference on*. IEEE. DOI: 10.1109/3dv.2017.00064. URL: http://gvv.mpi-inf.mpg.de/3dhp_dataset.

Mehta, Dushyant, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Mohamed Elgharib, Pascal Fua, Hans-Peter Seidel, Helge Rhodin, Gerard Pons-Moll, and Christian Theobalt (2020). "XNect: Real-time Multi-person 3D Human Motion Capture with a Single RGB Camera". In: *ACM Transactions on Graphics (Proceedings SIGGRAPH)*. URL: http://gvv.mpi-inf.mpg.de/projects/XNect/.

Mehta, Dushyant, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Srinath Sridhar, Gerard Pons-Moll, and Christian Theobalt (2018). "Single-Shot Multi-Person 3D Pose Estimation From Monocular RGB". In: *3D Vision (3DV), 2018 Sixth International Conference on*. IEEE. URL: http://gvv.mpi-inf.mpg.de/projects/SingleShotMultiPerson/.

Mehta, Dushyant, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt (2017b). "VNect: Real-time 3D Human Pose Estimation with a Single RGB Camera". In: *ACM Transactions on Graphics (Proceedings SIGGRAPH)* 36.4. URL: http://gvv.mpi-inf.mpg.de/projects/VNect/.

References

Agarwal, Ankur and Bill Triggs (2006). "Recovering 3D human pose from monocular images". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 28.1, pp. 44–58.

Agarwal, Sameer, Keir Mierle, et al. (2017). *Ceres Solver*. http://ceres-solver.org.

Akhter, Ijaz and Michael J Black (2015). "Pose-conditioned joint angle limits for 3D human pose reconstruction". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1446–1455.

Alldieck, Thiemo, Marcus Magnor, Bharat Lal Bhatnagar, Christian Theobalt, and Gerard Pons-Moll (June 2019). "Learning to Reconstruct People in Clothing from a Single RGB Camera". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Alldieck, Thiemo, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll (2018a). "Detailed Human Avatars from Monocular Video". In: *3DV*.

— (June 2018b). "Video Based Reconstruction of 3D People Models". In: *CVPR*.

Andriluka, M., U. Iqbal, E. Ensafutdinov, L. Pishchulin, A. Milan, J. Gall, and Schiele B. (2018). "PoseTrack: A Benchmark for Human Pose Estimation and Tracking". In: *CVPR*.

Andriluka, Mykhaylo, Leonid Pishchulin, Peter Gehler, and Bernt Schiele (June 2014). "2D Human Pose Estimation: New Benchmark and State of the Art Analysis". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Andriluka, Mykhaylo, Stefan Roth, and Bernt Schiele (2009). "Pictorial structures revisited: People detection and articulated pose estimation". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1014–1021.

Arora, Sanjeev, Nadav Cohen, and Elad Hazan (2018). "On the Optimization of Deep Networks: Implicit Acceleration by Overparameterization". In: *ICLR*.

ASUS (2011). *Xtion PRO LIVE*. https://www.asus.com/3D-Sensor/Xtion_PRO_LIVE/.

Baak, Andreas, Meinard Müller, Gaurav Bharaj, Hans-Peter Seidel, and Christian Theobalt (2011). "A Data-Driven Approach for Real-Time Full Body Pose Reconstruction from a Depth Camera". In: *IEEE International Conference on Computer Vision (ICCV)*. Bacelona, Spain.

Bagherinezhad, Hessam, Mohammad Rastegari, and Ali Farhadi (2017). "Lcnn: Lookup-based convolutional neural network". In: *Proc. IEEE CVPR*.

Bakken, Rune Havnung and Adrian Hilton (2012). "Real-time Pose Estimation using Tree Structures Built from Skeletonised Volume Sequences." In: *VISAPP (2)*, pp. 181–190.

Balan, Alexandru O, Leonid Sigal, and Michael J Black (2005). "A quantitative evaluation of video-based 3D person tracking". In: *2005 IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*. IEEE, pp. 349–356.

Balan, Alexandru O, Leonid Sigal, Michael J Black, James E Davis, and Horst W Haussecker (2007). "Detailed human shape and pose from images". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8.

Belagiannis, Vasileios and Andrew Zisserman (2016). "Recurrent Human Pose Estimation". In: *arXiv preprint arXiv:1605.02914*.

Bianco, Simone, Remi Cadene, Luigi Celona, and Paolo Napoletano (2018). "Benchmark Analysis of Representative Deep Neural Network Architectures". In: *IEEE Access* 6, pp. 64270–64277.

Bissacco, Alessandro, Ming-Hsuan Yang, and Stefano Soatto (2007). "Fast human pose estimation using appearance and motion via multi-dimensional boosting regression". In: *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 1–8.

Bo, Liefeng and Cristian Sminchisescu (2010). "Twin gaussian processes for structured prediction". In: *International Journal of Computer Vision* 87.1-2, pp. 28–52.

Bogo, Federica, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black (2016). "Keep it SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image". In: *European Conference on Computer Vision (ECCV)*.

Bourdev, Lubomir and Jitendra Malik (2009). "Poselets: Body part detectors trained using 3d human pose annotations". In: *IEEE International Conference on Computer Vision (ICCV)*, pp. 1365–1372.

Brau, Ernesto and Hao Jiang (2016). "3D Human Pose Estimation via Deep Learning from 2D Annotations". In: *International Conference on 3D Vision (3DV)*.

Bregler, Christoph and Jitendra Malik (1998). "Tracking people with twists and exponential maps". In: *Conference on Computer Vision and Pattern Recognition*, pp. 8–15.

Bridgeman, Lewis, Marco Volino, Jean-Yves Guillemaut, and Adrian Hilton (2019). "Multi-person 3D Pose Estimation and Tracking in Sports". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 0–0.

BuciluǍŽ, Cristian, Rich Caruana, and Alexandru Niculescu-Mizil (2006). "Model compression". In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pp. 535–541.

Bulat, Adrian and Georgios Tzimiropoulos (2016). "Human pose Estimation via Convolutional Part Heatmap Regression". In: *European Conference on Computer Vision (ECCV)*.

Cao, Zhe, Tomas Simon, Shih-En Wei, and Yaser Sheikh (2017). "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Carreira, Joao, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik (2016). "Human pose estimation with iterative error feedback". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Caruana, Rich (July 1997). "Multitask Learning". In: *Machine Learning* 28.1, pp. 41–75. ISSN: 1573-0565. DOI: 10.1023/A:1007379606734. URL: https://doi.org/10.1023/A:1007379606734.

Chai, Jinxiang and Jessica K Hodgins (2005). "Performance animation from low-dimensional control signals". In: *ACM Transactions on Graphics (TOG)* 24.3, pp. 686–696.

Chauvin, Yves (1989). "A back-propagation algorithm with optimal use of hidden units". In: *Advances in neural information processing systems*, pp. 519–526.

Chen, Ching-Hang and Deva Ramanan (2017). "3D Human Pose Estimation = 2D Pose Estimation + Matching". In: *CVPR 2017-IEEE Conference on Computer Vision & Pattern Recognition*.

Chen, Wenlin, James Wilson, Stephen Tyree, Kilian Weinberger, and Yixin Chen (2015). "Compressing neural networks with the hashing trick". In: *International Conference on Machine Learning*, pp. 2285–2294.

Chen, Wenzheng, Huan Wang, Yangyan Li, Hao Su, Zhenhua Wang, Changhe Tu, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen (2016). "Synthesizing training images for boosting human 3d pose estimation". In: *3D Vision (3DV), 2016 Fourth International Conference on*. IEEE, pp. 479–488.

Chen, Xianjie and Alan L Yuille (2014). "Articulated pose estimation by a graphical model with image dependent pairwise relations". In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 1736–1744.

Cheng, Yu, Duo Wang, Pan Zhou, and Tao Zhang (2017). "A survey of model compression and acceleration for deep neural networks". In: *arXiv preprint arXiv:1710.09282*.

Chollet, François (2017). "Xception: Deep learning with depthwise separable convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1251–1258.

Dabral, Rishabh, Anurag Mundhada, Uday Kusupati, Safeer Afaque, Abhishek Sharma, and Arjun Jain (2018). "Learning 3d human pose from structure and motion". In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 668–683.

Dalal, Navneet and Bill Triggs (2005). "Histograms of oriented gradients for human detection". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei (2009). "ImageNet: A Large-Scale Hierarchical Image Database". In: *CVPR09*.

Denton, Emily L, Wojciech Zaremba, Joan Bruna, Yann LeCun, and Rob Fergus (2014). "Exploiting linear structure within convolutional networks for efficient evaluation". In: *Advances in neural information processing systems*, pp. 1269–1277.

Deutscher, Jonathan and Ian Reid (2005). "Articulated body motion capture by stochastic search". In: *International Journal of Computer Vision* 61.2, pp. 185–205.

Dinh, Laurent, Razvan Pascanu, Samy Bengio, and Yoshua Bengio (2017). "Sharp Minima Can Generalize For Deep Nets". In: *ICML*.

Dou, Mingsong, Sameh Khamis, Yury Degtyarev, Philip Davidson, Sean Ryan Fanello, Adarsh Kowdle, Sergio Orts Escolano, Christoph Rhemann, David Kim, Jonathan Taylor, et al. (2016). "Fusion4d: Real-time performance capture of challenging scenes". In: *ACM Transactions on Graphics (TOG)* 35.4, p. 114.

Duchi, John, Elad Hazan, and Yoram Singer (2011). "Adaptive subgradient methods for online learning and stochastic optimization". In: *Journal of Machine Learning Research* 12.Jul, pp. 2121–2159.

Elgammal, Ahmed and Chan-Su Lee (2004). "Inferring 3D body pose from silhouettes using activity manifold learning". In: *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*. Vol. 2. IEEE, pp. II–681.

Elhayek, A., E. Aguiar, A. Jain, J. Tompson, L. Pishchulin, M. Andriluka, C. Bregler, B. Schiele, and C. Theobalt (2016). "MARCOnI - ConvNet-based MARker-less Motion Capture in Outdoor and Indoor Scenes". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

Elsken, Thomas, Jan Hendrik Metzen, and Frank Hutter (2018). "Neural Architecture Search: A Survey". In: *arXiv preprint arXiv:1808.05377*.

Fabbri, Matteo, Fabio Lanzi, Simone Calderara, Andrea Palazzi, Roberto Vezzani, and Rita Cucchiara (2018). "Learning to Detect and Track Visible and Occluded Body Joints in a Virtual World". In: *European Conference on Computer Vision (ECCV)*.

Fastovets, Mykyta, Jean-Yves Guillemaut, and Adrian Hilton (2013). "Athlete pose estimation from monocular tv sports footage". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1048–1054.

Felzenszwalb, Pedro F and Daniel P Huttenlocher (2005). "Pictorial structures for object recognition". In: *International Journal of Computer Vision (IJCV)* 61.1, pp. 55–79.

Ferrari, Vittorio, Manuel Marin-Jimenez, and Andrew Zisserman (2009). "Pose search: retrieving people using their pose". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8.

Gall, Juergen, Bodo Rosenhahn, Thomas Brox, and Hans-Peter Seidel (2010). "Optimization and Filtering for Human Motion Capture". In: *International Journal of Computer Vision (IJCV)* 87.1–2, pp. 75–92.

Ganapathi, Varun, Christian Plagemann, Daphne Koller, and Sebastian Thrun (2012). "Real-time human pose tracking from range data". In: *European conference on computer vision*. Springer, pp. 738–751.

Ganin, Yaroslav and Victor Lempitsky (2015). "Unsupervised Domain Adaptation by Backpropagation". In: *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*. ICML'15. Lille, France: JMLR.org, pp. 1180–1189. URL: http://dl.acm.org/citation.cfm?id=3045118.3045244.

Garg, Ravi, Anastasios Roussos, and Lourdes Agapito (2013). "Dense variational reconstruction of non-rigid surfaces from monocular video". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1272–1279.

Geirhos, Robert, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel (2018). "ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness". In: *arXiv preprint arXiv:1811.12231*.

Gilbert, Andrew, Matthew Trumble, Charles Malleson, Adrian Hilton, and John Collomosse (2019). "Fusing visual and inertial sensors with semantics for 3d human pose estimation". In: *International Journal of Computer Vision* 127.4, pp. 381–397.

Girdhar, Rohit, Georgia Gkioxari, Lorenzo Torresani, Manohar Paluri, and Du Tran (2018). "Detect-and-track: Efficient pose estimation in videos". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 350–359.

Gkioxari, Georgia, Bharath Hariharan, Ross Girshick, and Jitendra Malik (2014). "Using k-poselets for detecting people and localizing their keypoints". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3582–3589.

Gkioxari, Georgia, Alexander Toshev, and Navdeep Jaitly (2016). "Chained Predictions Using Convolutional Neural Networks". In: *European Conference on Computer Vision (ECCV)*.

Glorot, Xavier and Yoshua Bengio (2010). "Understanding the difficulty of training deep feedforward neural networks". In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016). *Deep Learning*. http://www.deeplearningbook.org. MIT Press.

Gotardo, Paulo FU and Aleix M Martinez (2011). "Computing smooth time trajectories for camera and deformable shape in structure from motion with occlusion". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33.10, pp. 2051–2065.

Güler, Riza Alp, Natalia Neverova, and Iasonas Kokkinos (2018). "DensePose: Dense Human Pose Estimation in the Wild". In: *CVPR*, pp. 7297–7306. DOI: 10.1109/CVPR.2018.00762. URL: http://openaccess.thecvf.com/content%5C_cvpr%5C_2018/html/Guler%5C_DensePose%5C_Dense%5C_Human%5C_CVPR%5C_2018%5C_paper.html.

Guo, Yiwen, Anbang Yao, and Yurong Chen (2016). "Dynamic network surgery for efficient dnns". In: *Advances In Neural Information Processing Systems*, pp. 1379–1387.

Haeffele, Benjamin D and René Vidal (2015). "Global optimality in tensor factorization, deep learning, and beyond". In: *arXiv preprint arXiv:1506.07540*.

Han, Song, Huizi Mao, and William J Dally (2016). "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding". In: *ICLR*.

Han, Song, Jeff Pool, John Tran, and William Dally (2015). "Learning both weights and connections for efficient neural network". In: *Advances in neural information processing systems*, pp. 1135–1143.

Han, Tengda, Weidi Xie, and Andrew Zisserman (2019). "Video representation learning by dense predictive coding". In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 0–0.

Hanson, Stephen José and Lorien Y Pratt (1989). "Comparing biases for minimal network construction with back-propagation". In: *Advances in neural information processing systems*, pp. 177–185.

Hassan, Mohamed, Vasileios Choutas, Dimitrios Tzionas, and Michael J. Black (Oct. 2019). "Resolving 3D Human Pose Ambiguities with 3D Scene Constraints". In: *International Conference on Computer Vision*. URL: https://prox.is.tue.mpg.de.

Hassibi, Babak, David G Stork, and Gregory J Wolff (1993). "Optimal brain surgeon and general network pruning". In: *Neural Networks, 1993., IEEE International Conference on*. IEEE, pp. 293–299.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016a). "Deep Residual Learning for Image Recognition". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

— (2016b). "Identity mappings in deep residual networks". In: *European conference on computer vision*. Springer, pp. 630–645.

He, Yang, Guoliang Kang, Xuanyi Dong, Yanwei Fu, and Yi Yang (2018). "Soft Filter Pruning for Accelerating Deep Convolutional Neural Networks". In: *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2234–2240.

Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean (2015). "Distilling the knowledge in a neural network". In: *arXiv preprint arXiv:1503.02531*.

Howard, Andrew G, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam (2018). "Mobilenets: Efficient convolutional neural networks for mobile vision applications". In: *Proc. IEEE CVPR*.

Howe, Nicholas R, Michael E Leventon, and William T Freeman (1999). "Bayesian Reconstruction of 3D Human Motion from Single-Camera Video." In: *NIPS*. Vol. 99, pp. 820–6.

Hu, Hengyuan, Rui Peng, Yu-Wing Tai, and Chi-Keung Tang (2016). "Network trimming: A data-driven neuron pruning approach towards efficient deep architectures". In: *arXiv preprint arXiv:1607.03250*.

Hu, Peiyun, Deva Ramanan, Jia Jia, Sen Wu, Xiaohui Wang, Lianhong Cai, and Jie Tang (2016). "Bottom-Up and Top-Down Reasoning with Hierarchical Rectified Gaussians". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Huang, Gao, Shichen Liu, Laurens Van der Maaten, and Kilian Q Weinberger (2018). "Condensenet: An efficient densenet using learned group convolutions". In: *Proc. IEEE CVPR*.

Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger (2017). "Densely connected convolutional networks." In: *CVPR*.

Huang, Yinghao, Manuel Kaufmann, Emre Aksan, Michael J. Black, Otmar Hilliges, and Gerard Pons-Moll (Nov. 2018). "Deep Inertial Poser: Learning to Reconstruct Human Pose from Sparse Inertial Measurements in Real Time". In: *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)* 37. First two authors contributed equally, 185:1–185:15.

Hubara, Itay, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio (2016). "Binarized neural networks". In: *Advances in neural information processing systems*, pp. 4107–4115.

Iandola, Forrest N, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer (2016). "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size". In: *ICLR*.

Innmann, Matthias, Michael Zollhöfer, Matthias Nießner, Christian Theobalt, and Marc Stamminger (Oct. 2016). "VolumeDeform: Real-time Volumetric Non-rigid Reconstruction". In:

Insafutdinov, Eldar, Mykhaylo Andriluka, Leonid Pishchulin, Siyu Tang, Evgeny Levinkov, Bjoern Andres, Bernt Schiele, and Saarland Informatics Campus (2017). "ArtTrack: Articulated multi-person tracking in the wild". In: *Proc. of CVPR*.

Insafutdinov, Eldar, Leonid Pishchulin, Bjoern Andres, Mykhaylo Andriluka, and Bernt Schiele (2016). "DeeperCut: A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model". In: *European Conference on Computer Vision (ECCV)*.

Ioffe, Sergey and Christian Szegedy (2015). "Batch normalization: Accelerating deep network training by reducing internal covariate shift". In: *arXiv preprint arXiv:1502.03167*.

Ionescu, Catalin, Joao Carreira, and Cristian Sminchisescu (2014a). "Iterated second-order label sensitive pooling for 3d human pose estimation". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1661–1668.

Ionescu, Catalin, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu (2014b). "Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 36.7, pp. 1325–1339.

Iqbal, Umar and Juergen Gall (2016). "Multi-person pose estimation with local joint-to-person associations". In: *European Conference on Computer Vision Workshops*. Springer, pp. 627–642.

Iqbal, Umar, Anton Milan, and Juergen Gall (2017). "Posetrack: Joint multi-person pose estimation and tracking". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2011–2020.

Iskakov, Karim, Egor Burkov, Victor Lempitsky, and Yury Malkov (2019). "Learnable Triangulation of Human Pose". In: *International Conference on Computer Vision (ICCV)*.

Jaderberg, Max, Andrea Vedaldi, and Andrew Zisserman (2014). "Speeding up convolutional neural networks with low rank expansions". In: *arXiv preprint arXiv:1405.3866*.

Jain, Arjun, Jonathan Tompson, Mykhaylo Andriluka, Graham W Taylor, and Christoph Bregler (2013). "Learning human pose estimation features with convolutional networks". In: *arXiv preprint arXiv:1312.7302*.

Jain, Arjun, Jonathan Tompson, Yann LeCun, and Christoph Bregler (2014). "Modeep: A deep learning framework using motion features for human pose estimation". In: *Asian Conference on Computer Vision (ACCV)*. Springer, pp. 302–315.

Jia, Yangqing, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell (2014). "Caffe: Convolutional architecture for fast feature embedding". In: *Proceedings of the 22nd ACM International Conference on Multimedia*, pp. 675–678.

Johnson, Sam and Mark Everingham (2010). "Clustered Pose and Nonlinear Appearance Models for Human Pose Estimation". In: *British Machine Vision Conference (BMVC)*. doi:10.5244/C.24.12.

— (2011). "Learning Effective Human Pose Estimation from Inaccurate Annotation". In: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*.

Caffe (2018). https://github.com/jolibrain/caffe.

Joo, Hanbyul, Hao Liu, Lei Tan, Lin Gui, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh (2015). "Panoptic Studio: A Massively Multiview System for Social Motion Capture". In: *ICCV*, pp. 3334–3342.

Joo, Hanbyul, Tomas Simon, and Yaser Sheikh (June 2018). "Total Capture: A 3D Deformation Model for Tracking Faces, Hands, and Bodies". In: *CVPR*.

Kanazawa, Angjoo, Michael J. Black, David W. Jacobs, and Jitendra Malik (2018). "End-to-end Recovery of Human Shape and Pose". In: *CVPR*.

Katircioglu, Isinsu, Bugra Tekin, Mathieu Salzmann, Vincent Lepetit, and Pascal Fua (2018). "Learning latent representations of 3d human pose with deep neural networks". In: *International Journal of Computer Vision* 126.12, pp. 1326–1341.

Keskar, Nitish Shirish, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang (2017a). "On large-batch training for deep learning: Generalization gap and sharp minima". In: *ICLR*.

Keskar, Nitish Shirish and Richard Socher (2017b). "Improving generalization performance by switching from adam to sgd". In: *arXiv preprint arXiv:1712.07628*.

Kim, Yong-Deok, Eunhyeok Park, Sungjoo Yoo, Taelim Choi, Lu Yang, and Dongjun Shin (2015). "Compression of Deep Convolutional Neural Networks for Fast and Low Power Mobile Applications". In: arXiv: 1511.06530 [cs.CV].

Kingma, Diederik P and Jimmy Ba (2014). "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980*.

Krizhevsky, Alex and Geoffrey Hinton (2009). "Learning multiple layers of features from tiny images". In:

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

La Gorce, Martin de, Nikos Paragios, and David J Fleet (2008). "Model-based hand tracking with texture, shading and self-occlusions". In: *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference On*. IEEE, pp. 1–8.

Lassner, Christoph, Javier Romero, Martin Kiefel, Federica Bogo, Michael J. Black, and Peter V. Gehler (July 2017). "Unite the People: Closing the Loop Between 3D and 2D Human Representations". In: *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Lavi, Bahram, Mehdi Fatan Serj, and Ihsan Ullah (2018). "Survey on deep learning techniques for person re-identification task". In: *arXiv preprint arXiv:1807.05284*.

Lavin, Andrew and Scott Gray (2016). "Fast algorithms for convolutional neural networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4013–4021.

Lebedev, Vadim, Yaroslav Ganin, Maksim Rakhuba, Ivan Oseledets, and Victor Lempitsky (2014). *Speeding-up Convolutional Neural Networks Using Fine-tuned CP-Decomposition*. arXiv: 1412.6553 [cs.CV].

Lebedev, Vadim and Victor Lempitsky (2016). "Fast convnets using group-wise brain damage". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2554–2564.

LeCun, Yann, John S Denker, and Sara A Solla (1990). "Optimal brain damage". In: *Advances in neural information processing systems*, pp. 598–605.

Lee, Minsik, Jungchan Cho, Chong-Ho Choi, and Songhwai Oh (2013). "Procrustean normal distribution for non-rigid structure from motion". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1280–1287.

Li, Hao, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf (2017). "Pruning filters for efficient convnets". In: *ICLR*.

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

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

Liang, Xiaodan, Ke Gong, Xiaohui Shen, and Liang Lin (2018). "Look into person: Joint body parsing & pose estimation network and a new benchmark". In: *IEEE transactions on pattern analysis and machine intelligence* 41.4, pp. 871–885.

Lifshitz, Ita, Ethan Fetaya, and Shimon Ullman (2016). "Human Pose Estimation using Deep Consensus Voting". In: *European Conference on Computer Vision (ECCV)*.

Lin, Darryl, Sachin Talathi, and Sreekanth Annapureddy (2016). "Fixed point quantization of deep convolutional networks". In: *International Conference on Machine Learning*, pp. 2849–2858.

Lin, Mude, Liang Lin, Xiaodan Liang, Keze Wang, and Hui Chen (2017). "Recurrent 3D Pose Sequence Machines". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*.

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

Liu, Chenxi, Barret Zoph, Jonathon Shlens, Wei Hua, Li-Jia Li, Li Fei-Fei, Alan Yuille, Jonathan Huang, and Kevin Murphy (2017). "Progressive neural architecture search". In: *arXiv preprint arXiv:1712.00559*.

Liu, Qingju, Teofilo de Campos, Wenwu Wang, Philip Jackson, and Adrian Hilton (2015). "Person tracking using audio and depth cues". In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 22–30.

Liu, Wei, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, and Scott E. Reed (2016). "SSD: Single Shot MultiBox Detector". In: *European Conference on Computer Vision (ECCV)*.

Liu, Zhuang, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumeng Yan, and Changshui Zhang (2017). "Learning efficient convolutional networks through network slimming". In: *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, pp. 2755–2763.

Loper, Matthew M and Michael J Black (2014). "OpenDR: An approximate differentiable renderer". In: *European Conference on Computer Vision*. Springer, pp. 154–169.

Loper, Matthew, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black (2015). "SMPL: A Skinned Multi-Person Linear Model". In: *TOG* 34.6, 248:1–248:16.

Loshchilov, Ilya and Frank Hutter (2017). "Fixing weight decay regularization in adam". In: *arXiv preprint arXiv:1711.05101*.

Luo, Jian-Hao, Jianxin Wu, and Weiyao Lin (2017). "ThiNet: A Filter Level Pruning Method for Deep Neural Network Compression". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5058–5066.

Ma, Ziyang and Enhua Wu (2014). "Real-time and robust hand tracking with a single depth camera". In: *The Visual Computer* 30.10, pp. 1133–1144.

Malleson, Charles, John Collomosse, and Adrian Hilton (2019). "Real-Time Multi-person Motion Capture from Multi-view Video and IMUs". In: *International Journal of Computer Vision*, pp. 1–18.

Malleson, Charles, Andrew Gilbert, Matthew Trumble, John Collomosse, Adrian Hilton, and Marco Volino (2017). "Real-time full-body motion capture from video and imus". In: *2017 International Conference on 3D Vision (3DV)*. IEEE, pp. 449–457.

Marcard, Timo von, Roberto Henschel, Michael Black, Bodo Rosenhahn, and Gerard Pons-Moll (Sept. 2018). "Recovering Accurate 3D Human Pose in The Wild Using IMUs and a Moving Camera". In: *ECCV*.

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

Mehta, Sachin, Mohammad Rastegari, Anat Caspi, Linda Shapiro, and Hannaneh Hajishirzi (2018). "ESPNet: Efficient Spatial Pyramid of Dilated Convolutions for Semantic Segmentation". In: *ECCV*.

Meka, Abhimitra, Michael Zollhöfer, Christian Richardt, and Christian Theobalt (2016). "Live Intrinsic Video". In: *ACM Trans. Graph. (Proc. SIGGRAPH)* 35.4, 109:1–14.

Microsoft Corporation (2013). *Kinect for Xbox One*. http://www.xbox.com/en-US/xbox-one/accessories/kinect.

— (2015). *Kinect SDK*. https://developer.microsoft.com/en-us/windows/kinect.

Moeslund, Thomas B (1999). "The analysis-by-synthesis approach in human motion capture: A review". In: *The 8th Danish conference on pattern recognition and image analysis. Denmark: Copenhagen University*.

Moeslund, Thomas B., Adrian Hilton, and Volker KrÃ$\frac{1}{4}$ger (2006). "A Survey of Advances in Vision-based Human Motion Capture and Analysis". In: *CVIU* 104.2–3, pp. 90–126.

Molchanov, Pavlo, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz (2017). "Pruning convolutional neural networks for resource efficient inference". In:

Monszpart, Aron, Paul Guerrero, Duygu Ceylan, Ersin Yumer, and Niloy J. Mitra (2019). "iMapper: Interaction-guided Scene Mapping from Monocular Videos". In: *ACM SIGGRAPH*.

Moon, Gyeongsik, Juyong Chang, and Kyoung Mu Lee (2019). "Camera Distance-aware Top-down Approach for 3D Multi-person Pose Estimation from a Single RGB Image". In: *The IEEE Conference on International Conference on Computer Vision (ICCV)*.

Morcos, Ari S, David GT Barrett, Neil C Rabinowitz, and Matthew Botvinick (2018). "On the importance of single directions for generalization". In: *arXiv preprint arXiv:1803.06959*.

Moreno-Noguer, F. (2017). "3D Human Pose Estimation from a Single Image via Distance Matrix Regression". In: *CVPR 2017-IEEE Conference on Computer Vision & Pattern Recognition*.

Mori, Greg and Jitendra Malik (2006). "Recovering 3d human body configurations using shape contexts". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 28.7, pp. 1052–1062.

Mozer, Michael C and Paul Smolensky (1989). "Skeletonization: A technique for trimming the fat from a network via relevance assessment". In: *Advances in neural information processing systems*, pp. 107–115.

Newcombe, Richard A., Dieter Fox, and Steven M. Seitz (June 2015). "DynamicFusion: Reconstruction and Tracking of Non-Rigid Scenes in Real-Time". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Newell, Alejandro and Jia Deng (2017). "Associative Embedding: End-to-End Learning for Joint Detection and Grouping". In: *Advances in Neural Information Processing Systems (NIPS)*.

Newell, Alejandro, Kaiyu Yang, and Jia Deng (2016). "Stacked Hourglass Networks for Human Pose Estimation". In: *European Conference on Computer Vision (ECCV)*.

Nibali, Aiden, Zhen He, Stuart Morgan, and Luke Prendergast (2019). "3D Human Pose Estimation with 2D Marginal Heatmaps". In: *WACV*.

Nuke (2015). *The Foundary*. https://www.foundry.com/products/nuke.

Oikonomidis, Iason, Nikolaos Kyriazis, and Antonis A Argyros (2011). "Efficient model-based 3D tracking of hand articulations using Kinect." In: *BmVC*. Vol. 1. 2, p. 3.

Omran, Mohamed, Christop Lassner, Gerard Pons-Moll, Peter Gehler, and Bernt Schiele (2018). "Neural Body Fitting: Unifying Deep Learning and Model Based Human Pose and Shape Estimation". In: *3DV*.

Orts-Escolano, Sergio, Christoph Rhemann, Sean Fanello, Wayne Chang, Adarsh Kowdle, Yury Degtyarev, David Kim, Philip L Davidson, Sameh Khamis, Mingsong Dou, et al. (2016). "Holoportation: Virtual 3D Teleportation in Real-time". In: *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, pp. 741–754.

Paladini, Marco, Alessio Del Bue, João M. F. Xavier, Lourdes Agapito, Marko Stosic, and Marija Dodig (2012). "Optimal Metric Projections for Deformable and Articulated Structure-from-Motion". In: *International Jorunal of Computer Vision (IJCV)* 96.2, pp. 252–276.

Papandreou, George, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy (2017). "Towards Accurate Multi-person Pose Estimation in the Wild". In: *arXiv preprint arXiv:1701.01779*.

Park, Hyun Soo and Yaser Sheikh (2011). "3D reconstruction of a smooth articulated trajectory from a monocular image sequence". In: *International Conference on Computer Vision (ICCV)*, pp. 201–208.

Park, Jongsoo, Sheng Li, Wei Wen, Ping Tak Peter Tang, Hai Li, Yiran Chen, and Pradeep Dubey (2017). "Faster cnns with direct sparse convolutions and guided pruning". In: *ICLR*.

Paszke, Adam, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer (2017). "Automatic differentiation in PyTorch". In: *NIPS-W*.

Pavlakos, Georgios, Xiaowei Zhou, and Kostas Daniilidis (2018a). "Ordinal Depth Supervision for 3D Human Pose Estimation". In: *CVPR*.

Pavlakos, Georgios, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis (2017). "Coarse-to-Fine Volumetric Prediction for Single-Image 3D Human Pose". In: *CVPR 2017-IEEE Conference on Computer Vision & Pattern Recognition*.

Pavlakos, Georgios, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis (2018b). "Learning to Estimate 3D Human Pose and Shape from a Single Color Image". In: *CVPR*.

Pavllo, Dario, Christoph Feichtenhofer, David Grangier, and Michael Auli (2019). "3D human pose estimation in video with temporal convolutions and semi-supervised training". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Pham, Hieu, Melody Y Guan, Barret Zoph, Quoc V Le, and Jeff Dean (2018). "Efficient Neural Architecture Search via Parameter Sharing". In: *arXiv preprint arXiv:1802.03268*.

Pishchulin, Leonid, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter Gehler, and Bernt Schiele (2016). "DeepCut: Joint Subset Partition and Labeling for Multi Person Pose Estimation". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Pishchulin, Leonid, Arjun Jain, Mykhaylo Andriluka, Thorsten Thormählen, and Bernt Schiele (2012). "Articulated people detection and pose estimation: Reshaping the future". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 3178–3185.

Pons-Moll, Gerard, David J Fleet, and Bodo Rosenhahn (2014). "Posebits for monocular human pose estimation". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2337–2344.

Ramakrishna, Varun, Takeo Kanade, and Yaser Sheikh (2012). "Reconstructing 3d human pose from 2d image landmarks". In: *European Conference on Computer Vision*. Springer, pp. 573–586.

Rastegari, Mohammad, Vicente Ordonez, Joseph Redmon, and Ali Farhadi (2016). "Xnor-net: Imagenet classification using binary convolutional neural networks". In: *European Conference on Computer Vision*. Springer, pp. 525–542.

Real, Esteban, Alok Aggarwal, Yanping Huang, and Quoc V Le (2018). "Regularized evolution for image classifier architecture search". In: *arXiv preprint arXiv:1802.01548*.

Reddi, Sashank J, Satyen Kale, and Sanjiv Kumar (2018). "On the convergence of adam and beyond". In: *ICLR*.

Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun (2015). "Faster R-CNN: Towards real-time object detection with region proposal networks". In: *Advances in neural information processing systems*, pp. 91–99.

Rhodin, Helge, Christian Richardt, Dan Casas, Eldar Insafutdinov, Mohammad Shafiei, Hans-Peter Seidel, Bernt Schiele, and Christian Theobalt (2016a). "EgoCap: Egocentric Marker-less Motion Capture with Two Fisheye Cameras". In: *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*.

Rhodin, Helge, Nadia Robertini, Dan Casas, Christian Richardt, Hans-Peter Seidel, and Christian Theobalt (2016b). "General automatic human shape and motion capture using volumetric contour cues". In: *European Conference on Computer Vision (ECCV)*. Springer, pp. 509–526.

Rhodin, Helge, Nadia Robertini, Christian Richardt, Hans-Peter Seidel, and Christian Theobalt (2015). "A Versatile Scene Model With Differentiable Visibility Applied to Generative Pose Estimation". In: *ICCV*.

Robertini, Nadia, Dan Casas, Helge Rhodin, Hans-Peter Seidel, and Christian Theobalt (2016). "Model-based Outdoor Performance Capture". In: *International Conference on Computer Vision (3DV)*.

Rogez, G., P. Weinzaepfel, and C. Schmid (2019). "LCR-Net++: Multi-person 2D and 3D Pose Detection in Natural Images". In: DOI: 10.1109/TPAMI.2019.2892985.

Rogez, Grégory and Cordelia Schmid (2016). "MoCap-guided data augmentation for 3D pose estimation in the wild". In: *Advances in Neural Information Processing Systems*, pp. 3108–3116.

Rogez, Gregory, Philippe Weinzaepfel, and Cordelia Schmid (2017). "LCR-Net: Localization-Classification-Regression for Human Pose". In: *CVPR 2017-IEEE Conference on Computer Vision & Pattern Recognition*.

Rogge, Lorenz, Felix Klose, Michael Stengel, Martin Eisemann, and Marcus Magnor (2014). "Garment replacement in monocular video sequences". In: *ACM Transactions on Graphics (TOG)* 34.1, p. 6.

Romera, Eduardo, José M Alvarez, Luis M Bergasa, and Roberto Arroyo (2018). "Erfnet: Efficient residual factorized convnet for real-time semantic segmentation". In: *IEEE Transactions on Intelligent Transportation Systems* 19.1, pp. 263–272.

Romero, Adriana, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio (2014). "Fitnets: Hints for thin deep nets". In: *ICLR*.

Rosales, Rómer and Stan Sclaroff (2000). "Specialized mappings and the estimation of human body pose from a single image". In: *Human Motion, 2000. Proceedings. Workshop on*. IEEE, pp. 19–24.

— (2006). "Combining generative and discriminative models in a framework for articulated pose estimation". In: *International Journal of Computer Vision* 67.3, pp. 251–276.

Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei (2015). "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision (IJCV)* 115.3, pp. 211–252. DOI: 10.1007/s11263-015-0816-y.

Sandler, Mark, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen (2018). "Mobilenetv2: Inverted residuals and linear bottlenecks". In: *CVPR*. IEEE, pp. 4510–4520.

Sapp, Benjamin and Ben Taskar (2013). "MODEC: Multimodal Decomposable Models for Human Pose Estimation". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Scardapane, Simone, Danilo Comminiello, Amir Hussain, and Aurelio Uncini (2017). "Group sparse regularization for deep neural networks". In: *Neurocomputing* 241, pp. 81–89.

Shotton, Jamie, Toby Sharp, Alex Kipman, Andrew Fitzgibbon, Mark Finocchio, Andrew Blake, Mat Cook, and Richard Moore (2013). "Real-time human pose recognition in parts from single depth images". In: *Communications of the ACM* 56.1, pp. 116–124.

Sidenbladh, Hedvig, Michael J Black, and David J Fleet (2000). "Stochastic tracking of 3D human figures using 2D image motion". In: *European conference on computer vision*. Springer, pp. 702–718.

Sigal, Leonid, Alexandru O Balan, and Michael J Black (2010). "Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion". In: *International Journal of Computer Vision (IJCV)* 87.1-2, pp. 4–27.

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

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

Simon, Tomas, Hanbyul Joo, Iain Matthews, and Yaser Sheikh (2017). "Hand Keypoint Detection in Single Images using Multiview Bootstrapping". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

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

Sminchisescu, Cristian, Atul Kanaujia, and Dimitris Metaxas (2006). "Learning joint top-down and bottom-up processes for 3D visual inference". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1743–1752.

Sminchisescu, Cristian, Atul Kanaujia, and Dimitris N Metaxas (2007). "BM^{3}E: Discriminative Density Propagation for Visual Tracking". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.11, pp. 2030–2044.

Sminchisescu, Cristian and Bill Triggs (2001). "Covariance scaled sampling for monocular 3D body tracking". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Vol. 1. IEEE, pp. I–447.

Srinivas, Suraj and R Venkatesh Babu (2016). "Data-free parameter pruning for deep neural networks". In: *BMVC*.

Srivastava, Rupesh Kumar, Klaus Greff, and Jürgen Schmidhuber (2015). "Highway networks". In: *arXiv preprint arXiv:1505.00387*.

Stanford CS231n (n.d.). *CS231n Convolutional Neural Networks for Visual Recognition*. http://cs231n.github.io/neural-networks-1/.

Starck, Jonathan and Adrian Hilton (2003). "Model-based multiple view reconstruction of people". In: *IEEE International Conference on Computer Vision (ICCV)*, pp. 915–922.

Starck, Jonathan, Atsuto Maki, Shohei Nobuhara, Adrian Hilton, and Takashi Matsuyama (2009). "The multiple-camera 3-d production studio". In: *IEEE Transactions on circuits and systems for video technology* 19.6, pp. 856–869.

Stoll, Carsten, Nils Hasler, Juergen Gall, Hans-Peter Seidel, and Christian Theobalt (2011). "Fast articulated motion tracking using a sums of Gaussians body model". In: *IEEE International Conference on Computer Vision (ICCV)*, pp. 951–958.

Sun, Min and Silvio Savarese (2011). "Articulated part-based model for joint object detection and pose estimation". In: *IEEE International Conference on Computer Vision*. IEEE, pp. 723–730.

Szegedy, Christian, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi (2017). "Inception-v4, inception-resnet and the impact of residual connections on learning." In: *AAAI*. Vol. 4, p. 12.

Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna (2016). "Rethinking the inception architecture for computer vision". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826.

Taycher, Leonid, David Demirdjian, Trevor Darrell, and Gregory Shakhnarovich (2006). "Conditional random people: Tracking humans with crfs and grid filters". In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Vol. 1. IEEE, pp. 222–229.

Taylor, Camillo J (2000). "Reconstruction of articulated objects from point correspondences in a single uncalibrated image". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Vol. 1, pp. 677–684.

Tekin, B., P. Márquez-Neila, M. Salzmann, and P. Fua (2017). "Learning to Fuse 2D and 3D Image Cues for Monocular Body Pose Estimation". In: *IEEE International Conference on Computer Vision (ICCV)*.

Tekin, Bugra, Isinsu Katircioglu, Mathieu Salzmann, Vincent Lepetit, and Pascal Fua (2016a). "Structured Prediction of 3D Human Pose with Deep Neural Networks". In: *British Machine Vision Conference (BMVC)*.

Tekin, Bugra, Artem Rozantsev, Vincent Lepetit, and Pascal Fua (2016b). "Direct Prediction of 3D Body Poses from Motion Compensated Sequences". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

The Captury (2016). http://www.thecaptury.com/.

Theis, Lucas, Iryna Korshunova, Alykhan Tejani, and Ferenc Huszár (2017). "Faster gaze prediction with dense networks and Fisher pruning". In: *ICLR*.

Tiny ImageNet (n.d.). *Tiny ImageNet Visual Recognition Challenge*. https://tiny-imagenet.herokuapp.com/.

Tome, Denis, Chris Russell, and Lourdes Agapito (2017). "Lifting from the deep: Convolutional 3d pose estimation from a single image". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Tompson, Jonathan J, Arjun Jain, Yann LeCun, and Christoph Bregler (2014). "Joint training of a convolutional network and a graphical model for human pose estimation". In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 1799–1807.

Toshev, Alexander and Christian Szegedy (2014). "Deeppose: Human pose estimation via deep neural networks". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1653–1660.

Trumble, Matthew, Andrew Gilbert, Adrian Hilton, and John Collomosse (2016). "Deep convolutional networks for marker-less human pose estimation from multiple views". In: *Proceedings of the 13th European conference on visual media production (CVMP 2016)*, pp. 1–9.

— (2018). "Deep autoencoder for combined human pose estimation and body model upscaling". In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 784–800.

Trumble, Matthew, Andrew Gilbert, Charles Malleson, Adrian Hilton, and John Collomosse (2017). "Total Capture: 3D Human Pose Estimation Fusing Video and Inertial Sensors". In: *Proceedings of 28th British Machine Vision Conference*, pp. 1–13.

Tung, Hsiao-Yu, Hsiao-Wei Tung, Ersin Yumer, and Katerina Fragkiadaki (2017). "Self-supervised Learning of Motion Capture". In: *NeurIPS*, pp. 5242–5252.

Urtasun, Raquel, David J Fleet, and Pascal Fua (2005). "Monocular 3D tracking of the golf swing". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 932–938.

— (2006). "Temporal motion models for monocular and multiview 3d human body tracking". In: *Computer vision and image understanding* 104.2, pp. 157–177.

Varol, Gül, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid (2017). "Learning from Synthetic Humans". In: *CVPR*.

Vasilache, Nicolas, Jeff Johnson, Michael Mathieu, Soumith Chintala, Serkan Piantino, and Yann LeCun (2014). "Fast convolutional nets with fbfft: A GPU performance evaluation". In: *arXiv preprint arXiv:1412.7580*.

Vondrak, Marek, Leonid Sigal, Jessica Hodgins, and Odest Jenkins (2012). "Video-based 3D motion capture through biped control". In: *ACM Transactions On Graphics (TOG)* 31.4, p. 27.

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

Wang, Min, Baoyuan Liu, and Hassan Foroosh (2017). "Factorized Convolutional Neural Networks." In: *ICCV Workshops*, pp. 545–553.

Wei, Shih-En, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh (2016). "Convolutional Pose Machines". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Wei, Xiaolin, Peizhao Zhang, and Jinxiang Chai (2012). "Accurate realtime full-body motion capture using a single depth camera". In: *ACM Transactions on Graphics (TOG)* 31.6, p. 188.

Wen, Wei, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li (2016). "Learning structured sparsity in deep neural networks". In: *Advances in Neural Information Processing Systems*, pp. 2074–2082.

Wen, Wei, Cong Xu, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li (Oct. 2017). "Coordinating Filters for Faster Deep Neural Networks". In: *The IEEE International Conference on Computer Vision (ICCV)*.

Wren, Christopher Richard, Ali Azarbayejani, Trevor Darrell, and Alex Paul Pentland (1997). "Pfinder: real-time tracking of the human body". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 19.7, pp. 780–785.

Wu, Jiaxiang, Cong Leng, Yuhang Wang, Qinghao Hu, and Jian Cheng (2016). "Quantized convolutional neural networks for mobile devices". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4820–4828.

Xiang, Yu, Wonhui Kim, Wei Chen, Jingwei Ji, Christopher Choy, Hao Su, Roozbeh Mottaghi, Leonidas Guibas, and Silvio Savarese (2016). "ObjectNet3D: A Large Scale Database for 3D Object Recognition". In: *European Conference Computer Vision (ECCV)*.

Xiaohan Nie, Bruce, Ping Wei, and Song-Chun Zhu (2017). "Monocular 3D human pose estimation by predicting depth on joints". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3447–3455.

Xie, Saining, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He (2017). "Aggregated residual transformations for deep neural networks". In: *CVPR*. IEEE, pp. 5987–5995.

Xiu, Yuliang, Jiefeng Li, Haoyu Wang, Yinghong Fang, and Cewu Lu (2018). "Pose flow: Efficient online pose tracking". In: *arXiv preprint arXiv:1802.00977*.

Xu, Weipeng, Avishek Chatterjee, Michael Zollhöfer, Helge Rhodin, Dushyant Mehta, Hans-Peter Seidel, and Christian Theobalt (2018). "Monoperfcap: Human performance capture from monocular video". In: *TOG* 37.2, p. 27.

Yang, Fengting and Zihan Zhou (2018). "Recovering 3D Planes from a Single Image via Convolutional Neural Networks". In: *ECCV*, pp. 85–100.

Yang, Wei, Wanli Ouyang, Xiaolong Wang, Jimmy Ren, Hongsheng Li, and Xiaogang Wang (2018). "3d human pose estimation in the wild by adversarial learning". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Vol. 1.

Yang, Zichao, Marcin Moczulski, Misha Denil, Nando de Freitas, Alex Smola, Le Song, and Ziyu Wang (Dec. 2015). "Deep Fried Convnets". In: *The IEEE International Conference on Computer Vision (ICCV)*.

Yasin, Hashim, Umar Iqbal, Björn Krüger, Andreas Weber, and Juergen Gall (2016). "A Dual-Source Approach for 3D Pose Estimation from a Single Image". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ye, Jianbo, Xin Lu, Zhe Lin, and James Z Wang (2018). "Rethinking the Smaller-Norm-Less-Informative Assumption in Channel Pruning of Convolution Layers". In: *ICLR*.

Ye, Mao and Ruigang Yang (2014). "Real-time simultaneous pose and shape estimation for articulated objects using a single depth camera". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2345–2352.

Yu, Ruichi, Ang Li, Chun-Fu Chen, Jui-Hsin Lai, Vlad I. Morariu, Xintong Han, Mingfei Gao, Ching-Yung Lin, and Larry S. Davis (June 2018). "NISP: Pruning Networks Using Neuron

Importance Score Propagation". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yu, Yongkang, Feilinand Yonghao, Zhen Yilin, and Weidong Mohan (2016). "Marker-less 3D Human Motion Capture with Monocular Image Sequence and Height-Maps". In: *European Conference on Computer Vision (ECCV)*.

Zagoruyko, Sergey and Nikos Komodakis (2016). "Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer". In: *ICLR*.

Zanfir, Andrei, Elisabeta Marinoiu, and Cristian Sminchisescu (2018a). "Monocular 3D Pose and Shape Estimation of Multiple People in Natural Scenes–The Importance of Multiple Scene Constraints". In: *CVPR*, pp. 2148–2157.

Zanfir, Andrei, Elisabeta Marinoiu, Mihai Zanfir, Alin-Ionut Popa, and Cristian Sminchisescu (2018b). "Deep Network for the Integrated 3D Sensing of Multiple People in Natural Images". In: *NeurIPS*.

Zeiler, Matthew D (2012). "ADADELTA: an adaptive learning rate method". In: *arXiv preprint arXiv:1212.5701*.

Zhou, Bolei, Yiyou Sun, David Bau, and Antonio Torralba (2018). "Revisiting the Importance of Individual Units in CNNs via Ablation". In: *arXiv preprint arXiv:1806.02891*.

Zhou, Feng and Fernando De la Torre (2014). "Spatio-temporal matching for human detection in video". In: *European Conference on Computer Vision (ECCV)*, pp. 62–77.

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

Zhou, Xiaowei, Menglong Zhu, Spyridon Leonardos, and Kostas Daniilidis (2015b). "Sparse Representation for 3D Shape Estimation: A Convex Relaxation Approach". In: *arXiv preprint arXiv:1509.04309*.

Zhou, Xiaowei, Menglong Zhu, Spyridon Leonardos, Kosta Derpanis, and Kostas Daniilidis (2015c). "Sparseness Meets Deepness: 3D Human Pose Estimation from Monocular Video". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhou, Xingyi, Qixing Huang, Xiao Sun, Xiangyang Xue, and Yichen Wei (2017). "Towards 3D Human Pose Estimation in the Wild: A Weakly-Supervised Approach". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 398–407.

Zhou, Xingyi, Xiao Sun, Wei Zhang, Shuang Liang, and Yichen Wei (2016). "Deep Kinematic Pose Regression". In: *ECCV Workshop on Geometry Meets Deep Learning*.

Zhu, Yingying, Mark Cox, and Simon Lucey (2011). "3D motion reconstruction for real-world camera motion". In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, pp. 1–8.

Zhuang, Zhuangwei, Mingkui Tan, Bohan Zhuang, Jing Liu, Yong Guo, Qingyao Wu, Junzhou Huang, and Jinhui Zhu (2018). *Discrimination-aware Channel Pruning for Deep Neural Networks*. Ed. by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett. URL: http://papers.nips.cc/paper/7367-discrimination-aware-channel-pruning-for-deep-neural-networks.pdf.

Zollhöfer, Michael, Matthias Nießner, Shahram Izadi, Christoph Rhemann, Christopher Zach, Matthew Fisher, Chenglei Wu, Andrew Fitzgibbon, Charles Loop, Christian Theobalt, and Marc Stamminger (2014). "Real-time Non-rigid Reconstruction using an RGB-D Camera". In: *ACM Transactions on Graphics (TOG)* 33.4.

Zoph, Barret and Quoc V Le (2016). "Neural architecture search with reinforcement learning". In: *arXiv preprint arXiv:1611.01578*.