Vision for Social Robots:
Human Perception and Pose Estimation

Thesis by
Matteo Ruggero Ronchi

In Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2020
Defended December 20th, 2019

Matteo Ruggero Ronchi
ORCID: 0000-0002-4277-3314

ACKNOWLEDGEMENTS

Pursuing a PhD is such a great example of how life is full of wonderful contradictions. While it is meant to be one's quintessential solitary journey into scientific discovery, it has given me the opportunity and joy to encounter so many people and share so many moments that will have a lasting impact on my life.

I would like to start by thanking my advisor Pietro Perona. Working with you was my main motivation for pursuing a PhD at Caltech and I will forever cherish the many lessons you taught me, as they have made me a better scientist and a better man. Thank you for giving me the opportunity to follow my research interests, also independently, and for teaching me when I was right or wrong, helping me understand why. I will always hold dear your guidance.

I am also deeply grateful to all the members of my committee. Yisong Yue, for strengthening my foundations and emphasizing the need for sound theoretical arguments behind any practical application. Aaron Ames, for encouraging me to aim high and think big, even beyond the bounds of my main field of expertise. Jeremie Papon, for being supportive throughout and beyond my collaboration at Disney and for your thoughtful and caring advice. Katie Bouman, for jumping in so enthusiastically at a later stage of my Thesis, and for your inspiration and example as a scientist.

Marco Gori, Michelangelo Diligenti and Duccio Papini, your early mentorship during my years of college in Siena has imprinted on me the joy, passion and curiosity about science which propelled my engine throughout the long journey of my PhD. For that, I am forever in debt.

I cannot thank enough all the Computational Vision lab mates I had throughout the years. Each of you has challenged me to improve and set the highest standard of a scientist, colleague, and friend.

A special mention goes to Bo for being the first one to welcome me in the lab and a constant source of inspiration and knowledge. Xavi, for navigating me through the hurdles of my first publication and passing along the Italian torch. Oisin, for sharing your exceptional curiosity, insight and intuition, and for your mentorship. I am very proud of our work together and will cherish collaborating and learning with you as one of the most enjoyable and fun times of my PhD.

Ron, Eyrun, David, Grant and Joe, “the Blind Beavers” with whom I spent most of my PhD years, I am grateful for getting to know you and for developing friendships that go way beyond the walls of the lab, from our Gradiator victories, all the way to beers on the lawn, and spiked salsa and margaritas at Amigos!

Serim, Mason, Gabi, Cristina and Sara, I have fond memories of all the moments spent together, as they made my lab experience a better one. Toni and Neehar, I will always remember with pleasure getting lost in arguments, most with no purpose, during our lunch breaks outside Moore building.

I deeply thank all the department admins and Caltech personnel that have been patient with me and helped me figure out things a thousand times. In particular, Maria Lopez for making me feel like at home since my very first visit, Caroline Murphy for the day in, day out support, Tess Legaspi for our colorful conversations each time I had to bring in the blue cards when I didn’t sign up for courses in time, and Stella for our daily early morning chats and our almost successful attempt at growing an avocado tree in lab.

Many other people have made my experience at Caltech unique and unforgettable, and I take great pride in being able to call myself their friend. Thanks for sharing the joys and sorrows of these years.

Thanks to my mentor Brian for your constant presence, for always listening, and for giving me a safe space to express all the colors of my personality on and off a theater stage. You taught me to play without being afraid of the unknown, and to be comfortable without knowing all the answers. And you let me sing in a musical. Yeah.

To all my theater and improv friends, I enjoyed so much sharing the stage together. In particular, thanks to Utkarsh for being a great companion in so many endeavors, and Milan for *yes-anding* the idea of starting the Implicit group at Caltech, as improv has added so much to my life.

Wendy, our relationship has meant a lot to me over these years, and always will. Chatting with you week in, week out has made me realize the importance of not being afraid of ones own vulnerabilities.

Harry, I have enjoyed all the ups and downs of backpacking together in the Sierras and being your housemate. From the peaceful garden we grew at Ardendale, to the narrowly escaped explosion on Washington, your presence has meant so much.

Ali, Katie, Sofi, Taso, Scott, Cory and Riley thanks for being there for all the Superbowl, Friendsgiving, and Birthday celebrations. You really were my family away from family and have given me so much during the years.

Maria, thanks for being such an honest friend and fearless tango partner, no matter how many times I stepped on your toes. I have enjoyed improvising dance steps with you.

Marcello thanks for being my captain and friend, and lending me your helping hand to lift me up when knocked down, on the soccer field and beyond.

Arnold and Arturo, being with you made time fly by so quickly. I am grateful for our deep friendship. You've allowed me to be myself and put up with my competitive spirit at every single game, from Ludo to the soccer field.

Alessandro, thanks for your contagious enthusiasm since the very first day we met on a random Friday happy hour at the Rathskeller. You've encouraged me to smoothen up my angles, leading by example with your inspiring and kind nature. I appreciate our genuine friendship and the ways in which it has grown.

Joe you've been a constant throughout all aspects of my life at Caltech. From the long hours spent at the whiteboard in lab and coding in the Annenberg tree house, to arguing about the conundrums of academic research and philosophizing about random things. From the thrill of improvising, to making home made ice-cream and pesto, and sharing the passion of a well kept and fruitful garden. Thanks for being the friend I can always rely on.

Pablo, throughout the years you have been a well of inspiration for basically everything. Thanks for always giving all of yourself for any and all endeavors without even one moment of hesitation, as only a true friend can do. You taught me not to feel satisfied until I understand every single bit of something, all the way to the smallest screw. I couldn't have done it without you.

To my dearest friends of youth, Andrea, Gabbro and Giada from Rome, and Herve and Giovanni from my times in Siena, we proved that friendships can transcend time and space.

I wish to thank my family for everything. Papà, I am happy to have followed your footsteps and feel so fortunate for being able to share this accomplishment with you, as you were not able to share it with your dad. You've always encouraged me and challenged me in ways that have allowed me to grow and become who I am today.

Mamma, I constantly feel your love even across the world. Knowing that you are proud of me lifts me up and keeps me going even in the hardest moments. I too am proud of being a researcher like you.

Finally, Holly I don't have enough words to thank you, but hopefully these three can capture them all: I love you! You shared the laughs of my best days and the hardship of the most difficult moments. Throughout all of our journey together, you have yearned knowing me and have always supported and loved me for who I am. You are the most precious gift I received from my PhD at Caltech.

ABSTRACT

IN order to extract the underlying meaning from a scene captured from the surrounding world in a single still image, *social robots* will need to learn the human ability to detect different objects, understand their arrangement and relationships relative both to their own parts and to each other, and infer the dynamics under which they are evolving. Furthermore, they will need to develop and hold a notion of context to allow assigning different meanings (semantics) to the same visual configuration (syntax) of a scene.

The underlying thread of this Thesis is the investigation of new ways for enabling interactions between social robots and humans, by advancing the visual perception capabilities of robots when they process images and videos in which *humans are the main focus of attention*.

First, we analyze the general problem of scene understanding, as social robots moving through the world need to be able to interpret scenes without having been assigned a specific preset goal. Throughout this line of research, *i)* we observe that human actions and interactions which can be visually discriminated from an image follow a very heavy-tailed distribution; *ii)* we develop an algorithm that can obtain a spatial understanding of a scene by only using cues arising from the effect of perspective on a picture of a person's face; and *iii)* we define a novel taxonomy of errors for the task of estimating the 2D body pose of people in images to better explain the behavior of algorithms and highlight their underlying causes of error.

Second, we focus on the specific task of 3D human pose and motion estimation from monocular 2D images using weakly supervised training data, as accurately predicting human pose will open up the possibility of richer interactions between humans and social robots. We show that when 3D ground-truth data is only available in small quantities, or not at all, it is possible to leverage knowledge about the physical properties of the human body, along with additional constraints related to alternative types of supervisory signals, to learn models that can regress the full 3D pose of the human body and predict its motions from monocular 2D images.

Taken in its entirety, the intent of this Thesis is to highlight the importance of, and provide novel methodologies for, social robots' ability to interpret their surrounding environment, learn in a way that is robust to low data availability, and generalize previously observed behaviors to unknown situations in a similar way to humans.

PUBLISHED CONTENT AND CONTRIBUTIONS

M. R. Ronchi participated in designing the project, developing the method, running the experiments and writing the manuscript for all of the following papers:

M. R. Ronchi, O. Mac Aodha and P. Perona *"How Much Does Multi-View Self-Supervision Help 3D Pose Estimation?"*

URL: http://www.vision.caltech.edu/~mronchi/projects/MultiView3DPose

M. R. Ronchi, O. Mac Aodha, R. Eng and P. Perona *"It's all Relative: Monocular 3D Human Pose Estimation from Weakly Supervised Data."* 29th British Machine Vision Conference (2018, Newcastle, England)

URL: http://www.vision.caltech.edu/~mronchi/projects/RelativePose

M. R. Ronchi and P. Perona *"Benchmarking and Error Diagnosis in Multi-Instance Pose Estimation."* 16th International Conference on Computer Vision (2017, Venice, Italy).

DOI: 10.1109/ICCV.2017.48

URL: http://www.vision.caltech.edu/~mronchi/projects/PoseErrorDiagnosis

M. R. Ronchi, J. S. Kim and Y. Yue *"A Rotation Invariant Latent Factor Model for Moveme Discovery from Static Poses."* IEEE 16th International Conference on Data Mining (2016, Barcelona, Spain).

DOI: 10.1109/ICDM.2016.0156

URL: http://www.vision.caltech.edu/~mronchi/projects/RotationInvariantMovemes

M. R. Ronchi and P. Perona *"Describing Common Human Visual Actions in Images."* 26th British Machine Vision Conference (2015, Swansea, Wales).

DOI: 10.5244/C.29.52

URL: http://www.vision.caltech.edu/~mronchi/projects/Cocoa

X. P. Burgos-Artizzu, **M. R. Ronchi** and P. Perona *"Distance Estimation of an Unknown Person from a Portrait."* 13th European Conference on Computer Vision (2014, Zurich, Switzerland).

DOI: 10.1007/978-3-319-10590-1_21

URL: http://www.vision.caltech.edu/~mronchi/projects/FaceDistancePortrait

CONTENTS

Acknowledgements . . . iii
Abstract . . . vii
Published Content and Contributions . . . viii
Contents . . . ix
List of Figures . . . xii
List of Tables . . . xv

Introduction **2**
Chapter I: Flavors of Robots . . . 2
Chapter II: (Computer) Vision for Social Robots . . . 6
Chapter III: Thesis Statement . . . 11

I Scene Understanding **15**
Chapter IV: Describing Common Human Visual Actions . . . 16
4.1 Introduction . . . 17
4.2 Previous Work . . . 19
4.3 Framework . . . 21
4.4 Dataset Collection . . . 22
4.4.1 Visual VerbNet . . . 23
4.4.2 Image and Subject Selection . . . 24
4.4.3 Interactions Annotations . . . 24
4.4.4 Visual Actions Annotations . . . 26
4.5 Analysis . . . 27
4.6 Discussion and Conclusions . . . 32
Chapter V: Distance Estimation of an Unknown Person from a Portrait . . . 37
5.1 Introduction . . . 38
5.2 Related Work . . . 40
5.3 Caltech Multi-Distance Portraits Dataset . . . 41
5.3.1 Annotating CMDP . . . 42
5.4 Problem Formulation . . . 43
5.5 Method . . . 44
5.5.1 Facial Landmarks . . . 45
5.5.2 Proposed Approach . . . 46
5.6 Results . . . 47
5.6.1 Re-ordering Task . . . 48
5.6.2 Regression Task . . . 49
5.6.3 Physiognomy Interpretation . . . 51

5.7 Conclusions 54
Chapter VI: Benchmarking and Error Diagnosis in Multi-Instance Pose Estimation 57
6.1 Introduction 58
6.2 Related Work 59
6.2.1 Error Diagnosis 59
6.2.2 Evaluation Framework 60
6.2.3 Human Pose and Skeleton Color Coding 61
6.2.4 Algorithms 62
6.3 Multi-Instance Pose Estimation Errors 64
6.3.1 Localization Errors 64
6.3.2 Scoring Errors 72
6.3.3 Background False Positives and False Negatives 74
6.4 Sensitivity to Occlusion, Crowding, and Size 75
6.5 Discussion and Recommendations 78

II Recovering 3D Human Pose and Motion from Images and Videos using Weak Supervision 85

Chapter VII: A Rotation Invariant Latent Factor Model for Moveme Discovery from Static Poses 86
7.1 Introduction 87
7.2 Related Work 90
7.3 Models 91
7.3.1 Basic Notation and Framework 91
7.3.2 Baselines 92
7.3.3 Rotation-Invariant Latent Factor Model 93
7.3.4 Training Details 97
7.4 Experiments and Empirical Results 98
7.4.1 Dataset and Additional Annotations 99
7.4.2 Activity Recognition 100
7.4.3 Action Dynamics Inference & Manifold Traversal 101
7.4.4 Moveme Visualization 102
7.4.5 Angle Recovery 104
7.4.6 Generalization Behaviour 105
7.4.7 Manifold Visualization 105
7.5 Conclusion and Future Directions 107
Chapter VIII: It's all Relative: Monocular 3D Human Pose Estimation from Weakly Supervised Data 113
8.1 Introduction 114
8.2 Related Work 115
8.3 Method 118
8.3.1 Supervised 3D Pose Estimation 118
8.3.2 3D Pose Estimation with Relative Constraints 118
8.3.3 Implementation Details 120

8.4 Human Relative Depth Annotation Performance 121
8.5 3D Pose Estimation Results 125
8.5.1 Human3.6M Dataset 125
8.5.2 Leeds Sports Pose Dataset 130
8.6 Conclusion 130
Chapter IX: How Much Does Multi-View Self-Supervision Help 3D Pose Estimation? 137
9.1 Introduction 138
9.2 Related Work 140
9.3 Method 143
9.3.1 Time Contrastive Network Encoder 145
9.3.2 Pose Decoder 146
9.3.3 Viewpoint Decoder 147
9.3.4 Implementation Details 149
9.4 Experiments 150
9.4.1 Ego versus Camera-Centric 3D Pose Finetuning 151
9.4.2 Impact of Viewpoints During Training 152
9.4.3 Image Selection During Training 153
9.4.4 Identity Invariant Features 154
9.4.5 Embedding Dimensionality 156
9.4.6 Impact of the Pose Decoder 157
9.5 Discussion and Recommendations 160
9.6 Conclusion 161

Conclusions **169**
Chapter X: Summary of Thesis and Contributions 169
Chapter XI: Future Steps Towards Social Robots 171
Bibliography 175

LIST OF FIGURES

Number *Page*

1.1 The evolution of human-made tools . . . 3
1.2 The multi-faceted capabilities of Social Robots . . . 5
4.1 Example annotation from the COCO-a dataset . . . 17
4.2 Steps in the collection of the COCO-a dataset . . . 22
4.3 Visual VerbNet (VVN) . . . 23
4.4 Statistics from the COCO-a dataset . . . 24
4.5 Quality evaluation of the COCO-a annotations . . . 25
4.6 Distribution of the visual actions based on object category . . . 27
4.7 Analysis of the visual actions performed by people and the most common visual action in the COCO-a dataset . . . 29
4.8 Histogram of counts for objects and visual actions in the COCO-a dataset . . . 30
4.9 Heavy tail analysis of the visual actions . . . 31
4.10 Sample images and rare annotations contained in the COCO-a dataset 33
5.1 Example annotations from the Caltech Multi-Distance Portraits (CMDP) dataset . . . 38
5.2 Face landmark formats . . . 43
5.3 Impact of physiognomy when estimating the distance of a person from a portrait . . . 44
5.4 Performance of the face landmark estimation algorithm . . . 45
5.5 Distribution of errors by individual landmark . . . 46
5.6 Performance analysis on the re-ordering task . . . 48
5.7 Input landmarks' discriminative power on the re-ordering and regression tasks . . . 49
5.8 Performance analysis on the regression task . . . 50
5.9 Ablation study for the regression task . . . 50
5.10 Output of the regression algorithm for estimating the distance of an unknown person from a single portrait image . . . 51
5.11 Average per subject bias in the distance estimation task . . . 52
5.12 Visual example of how physiognomy biases distance estimation . . . 52

5.13 Correlation between the physiognomy of a person and the bias in its distance estimation . . . 53
6.1 Coarse to fine error analysis of multi-instance person keypoint estimation algorithms. . . . 58
6.2 Similarity measure between a keypoint detection and its ground-truth location . . . 61
6.3 COCO keypoints format for 2D human pose and skeleton color coding 62
6.4 Taxonomy of keypoint localization errors . . . 65
6.5 Example of instance scoring error . . . 66
6.6 Analysis of localization errors and their impact on performance . . . 67
6.7 Example of the progressive correction of keypoint localization errors 70
6.8 Analysis of scoring errors and their impact on performance . . . 73
6.9 Analysis of background errors (FP, FN) and their impact on performance 74
6.10 Sample images from the proposed benchmarks of the COCO dataset . 75
6.11 Performance and error sensitivity to occlusion and crowding . . . 76
6.12 Amount of training data in each of the suggested benchmarks for the COCO dataset . . . 77
6.13 Performance and error sensitivity to size . . . 78
7.1 Rotation invariant moveme discovery . . . 87
7.2 Moveme representation in the 2D image plane . . . 92
7.3 Training pipeline of the proposed method LFA-3D . . . 96
7.4 Example annotation for the orientation angle collected on the LSP dataset . . . 99
7.5 Performance analysis on the activity recognition task . . . 100
7.6 Performance analysis on the action dynamics inference task . . . 101
7.7 Time evolution visualization for three learned movemes . . . 103
7.8 Performance analysis on the viewpoint-angle estimation task and generalization analysis . . . 104
7.9 T-SNE visualization of the manifold of human poses for all the input images in the LSP dataset . . . 106
7.10 Activation heatmap over all the LSP dataset images for two learned movemes . . . 107
8.1 Monocular 3D pose estimation from weak supervision . . . 114
8.2 Trade-off between training data and 3D pose reconstruction error . . 115
8.3 Neural network architecture adopted for 3D pose regression . . . 121
8.4 Bias in the annotations due to camera perspective . . . 122

8.5 Human performance analysis on the relative keypoint depth annotation task . . . 123
8.6 Example of the relative keypoint depth annotations collected for the LSP dataset . . . 124
8.7 Performance analysis on the 3D pose estimation task . . . 127
8.8 Inference time 3D pose predictions on the Human3.6M dataset . . . 129
8.9 Inference time 3D pose predictions on the LSP dataset . . . 131
9.1 Multi-view self-supervision for the task of 3D human pose estimation 138
9.2 Encoder-decoder pipeline for monocular 3D human pose estimation . 144
9.3 Amount of supervision versus reconstruction error trade-off curve for Ego versus Camera-Centric pose representation . . . 151
9.4 Generalization of multi-view self-supervision to new viewpoints . . . 152
9.5 Distribution of pairwise distances in the learned embedding space . . 154
9.6 Quantitative and qualitative analysis of the invariance to identity of the learned embedding . . . 155
9.7 Sensitivity of the data versus error trade-off curve to the embedding dimension . . . 156
9.8 Impact of the pose decoder architecture on the amount of data versus reconstruction error trade-off curve . . . 157
9.9 Learning rate sensitivity for different pose decoder architectures . . . 158
9.10 Inference time ego-centric 3D pose predictions on the Human3.6M dataset . . . 159
11.1 Stereo and ego-centric video capturing rig . . . 172
11.2 Video capture setup for collecting a dataset of ego-centric videos of actions and social interactions . . . 172
11.3 Dynamic model of the leg of the robot Cassie . . . 173
11.4 Playing “Simon Says” in the viewpoint-invariant embedding . . . 174

LIST OF TABLES

Number *Page*

4.1 State-of-the-art datasets for single-frame action recognition 20
4.2 List of visual actions in COCO-a grouped by category 28
4.3 List of visual adverbs in COCO-a grouped by category 29
5.1 Subject variety in the CMDP dataset 41
6.1 The 2016 COCO keypoints challenge leaderboard 63
6.2 Performance improvement due to the optimal rescoring of detections 73
7.1 Example of the chronological re-ordering of a sequence of images using the LFA-3D method 102
8.1 Ablation study of our model on the Human3.6M dataset 126
8.2 3D pose reconstruction errors on the Human3.6M dataset 128

Introduction

Chapter 1

FLAVORS OF ROBOTS

From Tools to Robots

We live in a world full of robots, and it has been like this for more than a while.

Since the inception of our species, *Homo Sapiens* have been building tools for survival, as a form of protection, or for gathering plants and hunting [1, 2]. Just a few tens of thousands of years later, in the transition from hunter-gatherers to a food-cultivating species [3–6], tools evolved into more complex, refined and specialized utensils to serve the new set of skills needed for proliferation.

Only relatively recently, when survival to the next day was not a rare privilege any longer, have our tools taken a completely different nature, driven by our intellectual growth and curiosity, rather than by specific daily needs [7, 8]. New powerful tools also meant having more time for leisurous activities, facilitating long and energy consuming tasks like crop plowing or irrigation [9], and enabling us to take interest in more abstract and complex tasks, such as calculating the positions of astronomical objects [10].

It is in this context that the term *automaton* was first adopted.

Automaton is the Latin translation of the Greek word αὐτό-ματον, meaning self-acting, the conjuction of the two words *autos*/"self" and *matos*/ "thinking, animated or willing". Although the word itself has ancient roots, it appeared for the first time in the late 16th century [11], in a book recollecting some of the original writings by the Greek mathematician Hero of Alexandria who extensively studied hydraulics, pneumatics and mechanics.

It is unclear exactly when the fascination with anthropomorphic automatons started, but there are a few very famous examples throughout the past centuries. The first known humanoid automaton is the "Automa Cavaliere", constructed by Leonardo da Vinci around the year 1495 [12, 13]. With the outer appearance of a Germanic knight, it was powered by a complex core of mechanical devices allowing it to stand, sit, raise its visor and independently manoeuver its arms.

The "Mechanical Turk" [15] is arguably the most famous and controversial automa-

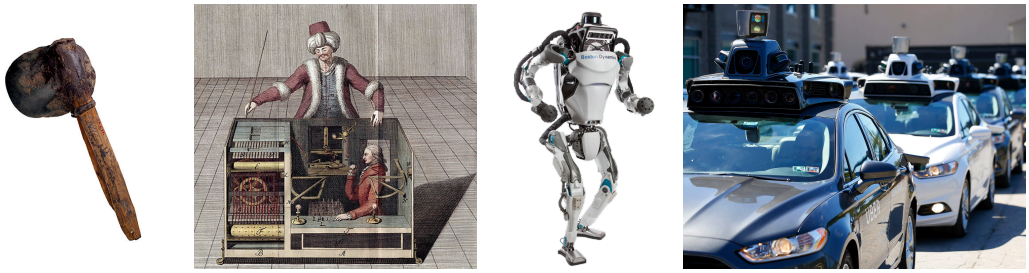


Figure 1.1: **The evolution of human-made tools.** As we evolved from *Homo Sapiens* to *Homo Deus* [14], our tools also matured: from the early days of building spears for hunting, to the Mechanical Turk, all the way to modern day robots and self-driving cars.

ton. It was a well constructed machine based on a mechanical illusion allowing a real person to hide inside of it and play games of chess. The Turk was designed by Wolfgang von Kempelen in 1770 to impress the Empress Maria Theresa of Austria and had a very successfull career[1] spanning almost a century until it was destroyed by a fire in 1854, along with its well-kept secret.

Another construction discussed in *The Golden Age of Automata* [16], is "The Flute Player" by Innocenzo Manzetti [17, 18]. The life-size man-shaped automaton was composed of internal levers, connecting rods and compressed air tubes, allowing its lips and fingers to move on a flute and perform 12 different arias according to a program recorded on a cylinder similar to those used in player pianos [19].

It was only in 1920 that the word *robot*, as we commonly adopt it today, was introduced in the play "R.U.R - Rossum's Universal Robots" by Karel Capek [20]. The term is derived from the Czech *robotnik*/"forced worker" and *robota*/"labor" and is an accomplishement of Capek's older brother, painter and writer Josef [21].

Since then, starting with the technological innovations of the industrial revolution, robots kept expanding into almost all aspects of society and popular culture. Nowadays, we find robots in fiction literature [22–25] and movies [26–29], as well as at the core of our industrial and economic infrastructure.

Today, the main challenge in robotics is not any longer whether we can build a robot to perform a specific task, but rather how small, efficient or similar to a human can we build it to perform such a task.

The practical and economic advantages of robots are such that now, and even more so in the near future, *we must learn to live in a world full of social robots*.

[1] Allegedly, the Turk was operated by many great chess masters, such as Johann Allgaier, Boncourt, Aaron Alexandre, William Lewis, Jacques Mouret, and William Schlumberger, allowing it to win most of the games it played during demonstrations around Europe and the Americas.

Social Robots

As technology keeps improving and becomes more and more pervasive, not only for industrial applications, but also in our private lives, robots are ready to take the next leap forward.

In this Thesis, we are interested in *Social*[2] *Robots*, and address some of the advancements that are required to enable their large scale deployment.

A social robot is a robot [30] capable of performing a variety of tasks in the context of a social interaction with a human agent [31]. Such tasks can vary from very specific ones, such as opening the door of a car for a person, to more open-ended ones, such as walking someone's dog. At the same time, the nature of the social interaction could require simple avoidance - who would want a 300 pound robot to run into them? - or encompass attention, complex expressive communication (verbal and non-verbal), collaborative or competitive behaviour, or even emotional response.

While intuitive, the above definition is still somewhat vague. Here we would like to suggest two criteria for marking the distinction between any robot and a social robot:

1. **Sensory perception:** A social robot must be equipped with an artificial perceptual system similar to the one that humans use to interact with each other. The different components of such a system should correspond to all five senses of the body: vision, sound, touch, taste, and smell.

2. **Social perception:** A social robot must be able to use its perception system to support interpersonal interactions, allow it to form impressions about other people as sovereign personalities, and to elaborate and test inferences based on such perceptions.

Facial expressions, tone of voice, hand gestures, and body posture or movement are just a few examples of nonverbal communication between people. Associating the look of a person sweating with nervousness rather than with sport fatigue is an example of social inference supported by the ability to detect smell, body parts and movements through olfactory and visual perceptions. While sensory perception can, to some extent, be learned and taught [32, 33], social perception can only be inferred. As a result, it is fundamental for social robots to be able to learn without

[2]The world social is derived from the Latin expression *socii*/"allies" named after the Italian Socii states, historical allies of the Roman Republic.

Figure 1.2: **The multi-faceted capabilities of Social Robots.** Social Robots are able to perform a variety of tasks in the context of social interactions with human agents. Pictures taken from a scene of the movie "I, Robot" [29].

pre-knowledge and be able to interpret the environment and generalize previously observed behaviours to unknown situations in a similar way that humans are able to.

Our goal is to push forward the transition from designing and building robots as a tool/*something* for purely functional, predefined and programmable tasks, to an entity/*someone* that is able to match our social and intellectual skills for accomplishing a common goal [31, 34].

To sustain such transition, it is fundamental for social robots to support the perceptive abilities that are at the basis of human intelligence and evolution [35, 36]. Using artificial intelligence methods to build robots with the ability to perceive is still an open endeavor that needs to be achieved in order for them to become truly social entities.

Chapter 2

(COMPUTER) VISION FOR SOCIAL ROBOTS

Visuomotor skills are fundamental for intelligence [37, 38] and there is a substantial corpus of research investigating how to build robots that can move like humans. This will inevitably help perception since vision and movement are deeply intertwined.

However, the main focus of this Thesis is on enhancing one of the perceptive channels of social robots, specifically visual perception.

In the human brain, neurons devoted to visual processing number in the hundreds of millions and take up about 30 percent of the cortex, compared with 8 percent for touch and just 3 percent for hearing. Humans dedicate about half of their brain power to visual system processes, and vision is by all means the most effective and direct interface with the world in which we live.

As a result, if we wish to have robots roaming around our cities and seamlessly interacting with us at coffee shops or in our homes as social companions or assistants, we need to make sure that their visual abilities are on par with the tasks they need to accomplish.

As a contribution to clarifying the challenge ahead of us, we'd like to put forward the following statement of work:

In order to extract the underlying meaning from a scene captured from the surrounding world in a single still image, social robots will need to learn the human ability to detect different objects, understand their arrangement and relationships relative both to their own parts and to each other, and infer the dynamics under which they are evolving. Furthermore, social robots will need to develop and hold a notion of context to allow assigning different meanings (semantics) to the same visual configuration (syntax) of a scene.

Computer Vision can help provide the means by which the above statement of work can be pursued and accomplished.

Object Detection: Understanding *what objects are where* [39] provides one of the most basic pieces of information needed by social robots.

Most of the early object detection algorithms were based on handcrafted local image feature descriptors - designed to be invariant to transformations such as scaling or rotation [40, 41] - which were selectively compared to parts of the image in the hope of finding a successful match. The first successful algorithm of this type is the Viola-Jones detector for human faces [42]. Improvements came in the form of more intricate feature descriptors, capable of capturing more informative patterns from the image, such as the histograms of oriented gradients, normalized gradients magnitude and color channels [43–45]. Deformable Part Models [46], defined by a global object filter together with a set of parts filters and associated deformation models between those parts, were the most effective of these methods for traditional object detection.

Object detection evolved in a new direction and at unprecedented speed with *i)* the resurgence of Convolutional Neural Networks [47], which provided a new effective representation for objects that could be extracted directly from images without having to be hand-designed, and *ii)* the availability of datasets with a large number of images [48–50], which allowed to capture the variability in appearance, providing robustness and statistical significance to the learned representations.

Detection algorithms are now almost exclusively based on Convolutional Neural Networks and grouped into two families: "two-stage architectures" [51–54] based on a coarse to fine process in which a classifier is applied to a sparse set of candidate object locations, and "one-stage architectures" [55–57], in which a classifier is applied over a regular dense sampling of all possible object locations in the image.

Part Estimation: Understanding the structure of objects in the world and their part anatomy is the only way in which a social robot can be able to interact with them once it established their nature (class/category).

Within the context of inanimate objects, there is significant overlap within the research for part estimation and object detection. However, in the case of entities for which highly non-rigid motions are very typical, such as humans, the task assumes a significantly different nature because of self-occlusion and the fact that those motions can significantly alter the appearance of the entity. As a result, different types of algorithms have been successful.

The concept that a small number of simple pixel locations (keypoints) tracking the main joints of the human body was sufficient to evoke compelling impressions of the entire human body performing complicated actions such as walking, running or

dancing was first hypothesized (shown) by Johansson [58].

Most of the early work in localizing parts of articulated objects (inanimated and animated) is inspired from the seminal work on Pictorial Structures [59], in which the parts of an object are represented by a fixed template and arranged in a deformable configuration. However, when the type of articulation can be very non-rigid, such as in the specific case of human parts estimation, the appearance of the template parts themselves can change significantly, so it is only with the introduction of more sophisticated graph based deformable part models [46, 60–63] that these approaches started working more reliably.

The focus on parts based methods remained very central until deep learning became more pervasive and there was availability of large datasets for human pose estimation [49, 64, 65]. Completely different methodologies were introduced in which part templates were no longer defined, but instead the precise pixel locations of keypoints could be directly regressed from the input image [66].

Now most work is defined in two approaches, top-down [67–69] which starts from a human detection and then detects its parts, or bottom up [70, 71] which detects parts of the body and then groups them into instances.

With the availability of datasets of different nature, the task of part estimation has also evolved. Now synthetic and motion capture datasets [72, 73] started enabling the task of looking for parts in 3D space and not in the image plane any longer. Similarly, other areas of interest look at estimating the 3D shape of the body now that 3D shapes of human body have been parametrized with new dataset [74].

Dynamics: Given a scene, being able to predict possible ways in which it might evolve will allow social robots to act and interact within the world with intention. The ability to cope with moving and changing objects, changing illumination, and changing viewpoints is essential to perform several tasks. If you're not able to predict the possible outcomes of a certain action, you cannot choose one based on whatever your goal might be.

Note that given the intrinsic ambiguities of certain situations, not even humans can fully resolve the problem of inferring the dynamic evolution of a scene from a static snapshot. However, from a very early age [75], we develop an intuition that allows us to predict the evolution of all the possible outcomes.

There are four possible cases under which a scene might be captured by a camera [76]: *i)* Stationary camera, stationary objects, *ii)* Stationary camera, moving objects, *iii)*

Moving camera, stationary objects, and *iv)* Moving camera, moving objects. Each one of these cases presents unique challenges and thus requires different approaches.

For example, one may study the intrinsic properties of the physical entities in a scene and try predicting their dynamics evolution. A tower of material blocks in a scene may be determined to be in unstable equilibrium and the final stable configuration accurately predicted based purely on the laws of physics [77, 78].

Alternatively, it could be possible to predict the way in which an image will evolve based on the current attributes of its pixels (a generative framework for future video prediction). This approach typically requires modeling the complex distribution of all possible future pixel configurations compatible with their initial state [79–81].

Yet another approach specific to the field of robotics attempts learning a model that can predict the result of the interaction of an agent with the world, in terms of changes in the visual input caused by a robotic control manipulation [82–84].

Ultimately, in the context of human motion, the goal is being able to predict the dynamics of a human body consistent with its current position and intrinsic anatomical contraints. Recent approaches tackle this problem by using a recurrent network starting from an image at the current state [85, 86].

Context: Context is the set of circumstances that form the setting for an event, statement, or idea, and in terms of which it can be fully understood and assessed. As obvious from its definition, context is an over-arching concept that can be applied to all of the previous work.

For social robots, understanding context of the environment implies the ability of understanding that they cannot just consider their main focus of attention when deciding what actions to take, but they also have to take into consideration all the inferential evidence from the surrounding world. For instance, if the robot witnesses a shooting scene, it must be able to discriminate the contextual value of that happening on a stage theater versus happening in a public street.

Computer Vision can help a social robot to validate and determine the context in which it is operating, or it can aid updating and learning a new context.

In current research, context is modeled as a conditioning variable which is used for providing answers to certain problems, and can assume one of two natures: *i) spatial*, in the sense of what does the world look like around the part of the scene we

are focusing on; *ii) temporal*, in the sense of what has happened immediately before what is being considered in the current scene.

The usefulness of context is captured in the famous COCO (common objects in context) dataset [49]. This is particularly useful when paired with algorithms such as Convolutional Neural Networks. In fact, CNNs have the ability of considering spatial context when extracting features for an image portion, due to the fact that their receptive fields are able to look at the entire picture at higher levels of the architecture, and exploit any available global information. Intuitively, larger context regions capture global spatial configurations of objects, while smaller context regions focus on the local part appearance.

Manually designed context representations, such as multiple bounding boxes or image crops, have been typically used to provide more precise contextual information about a scene. This has been very helpful for both object detection [87, 88] and human pose estimation [89–91]. However, more recently there has been a focus on studying methodologies that do not pre-define context, but model it as a form of visual attention in which the network can choose which portions of the image to focus on [92].

In the field of dynamics, context is typically considered as a small window of time around the current time, and 3D Convolutional Neural Networks [93] are used to make predictions over space and time. However, the length of the temporal context window is very small since these 3D convolutions require dense sampling in time to work accurately. More recently, for capturing longer term supportive information, a long-term set of feature banks [94] context is typically used by dynamic models and can be accessed whenever needed to aid the current recognition task.

Chapter 3

THESIS STATEMENT

The underlying thread of this Thesis is the investigation of new ways for enabling social interactions between robots and humans by advancing the perceptual capabilities of social robots when they process images and videos where *humans are the main focus of attention*.

The investigations carried out throughout this work address the following questions at multiple levels of abstraction:

- What information can be captured from a static scene that best helps inferring its context?
- Is it possible to recover information from the real world that is lost in the projection to monocular images?
- How can one extract information from images and videos on how humans are positioned and move?
- Can one develop algorithms that work accurately and have good generalization ability, even for tasks in which there is little high-quality training data?

We tackled the above topics along different dimensions: *i)* by collecting *datasets* for both well-known and novel tasks; *ii)* by introducing new *metrics* that capture the many sources of error that impact the performance of methods performing a certain task; *iii)* by designing *algorithms* that enable tasks that were previously thought impossible, and by solving known tasks based on less constraining assumptions.

The Thesis is articulated in two main sections, organized in a top-down fashion: in Part I, we analyze the high-level, general problem of scene understanding; in Part II, we focus on improving estimates of 3D pose and motion of humans captured in monocular two-dimensional images and videos when there is a limited amount of supervision available.

Part I

The aim of scene understanding entails the ability to provide a description of a scene depicted in an image as a whole without the need of a specific task or goal being specified. This comes very easily and natural for humans: even when images are just shown for a brief moment, we are able to elaborate informative general-purpose high-level scene descriptions useful for carrying out a diverse set of tasks also unrelated to the scene's individual elements.

As social robots move through the world and are presented with images of scenes dynamically changing, they will need to be able to interpret them without having been assigned a specific preset goal.

This requires many different skills. For one, understanding a scene as a whole is to be able to reason about the entities present in the image and the way in which they are positioned and interact with each other. Further, it implies having a correct spatial perception of the scene and being able to evaluate depths and the relative position of the objects in the scene. Finally, it requires distinguishing the number of people in the image, their positions (both anatomically and with respect to others) while they may be performing complicated actions.

In the following chapters, we look at this problem from three points of view independent from each other, yet all very relevant to the overarching goal of gaining understanding of a scene that can be useful for a variety of down-stream tasks.

Chapter 4: We answer the question of what should a task-independent image description look like. We propose a "graph of interactions" which, unlike previously adopted representations (such as captions or bags of objects), captures in an unambiguous way what are the *entities* in an image, their *attributes* and *relative positioning*, as well as the *action relationships* between them.

Chapter 5: We show that it is possible to obtain a spatial understanding of a scene in terms of the metric distance from a subject of interaction by only using cues arising from the effect of perspective on a two-dimensional monocular picture of the person's face. Furthermore, neither the subject's specific identity attributes, nor the camera calibration parameters are found to be necessary.

Chapter 6: We define a novel taxonomy of errors for analyzing the performance of algorithms on the task of estimating the two-dimensional body pose of people in monocular images. Our error analysis shows that characteristics of the portrayed

people substantially affect the performance and that current benchmarks do not appropriately capture the variability of images in real world scenarios.

Part II

One of the three fundamental pillars of modern Computer Vision is the unprecedented availability of large datasets [48, 95, 96] capturing all the aspects of variability for a specific task of interest.[1]

However, it is unrealistic to attempt to collect datasets capturing the variability of all the social tasks that robots will be expected to tackle. Besides being a very long tail distribution, the correct answers often cannot even be well defined, in the sense of a pre-defined sequence of steps constituting the "proper" behavior. In fact, open-ended tasks can be solved in many different ways and even humans might complete them in several different, but equally acceptable, ways.

As a result, we need to develop learning algorithms that are robust to low data availability and display generalization abilities that are at least comparable to those of humans.

In this Thesis, we have tackled this problem from the angle of estimating the 3D pose and motion of humans from monocular images and videos. The application of supervised machine learning to 3D pose estimation in real world images is currently hampered by the lack of varied training images with corresponding 3D poses. The majority of current 3D pose estimation algorithms are trained using data that has either been collected in carefully controlled studio settings or has been generated artificially, which hampers their applicability to real-life tasks.

Each of the three chapters in this part of the Thesis tackles the above problem with a slightly different approach, yet they all share the basic goal of improving the ability of Computer Vision algorithms to work in situations for which it is expensive (time and resource-wise) and complicated (multitude of environments) to capture the required supervision needed for applying a supervised Computer Vision methodology.

Chapter 7: We design a new algorithm that learns a set of three-dimensional basis poses of the human body that can be used to characterize the manifold of primitive human motions. Our technique uses only two-dimensional keypoints from images

[1] The other two are the resurgence of Convolutional Neural Networks [47, 97] along with the development and mass deployment of Graphics Processing Units.

depicting the actions containing the reconstructed motions, taken from various known camera pan angles at unknown instants of time.

Chapter 8: We define a novel loss function for training a neural network on the task of three-dimensional human pose estimation from monocular images, using only sparse and easily obtainable relative depth annotations. Despite the greatly simplified set of assumptions, our method shows performance comparable to that of fully supervised state-of-the-art algorithms.

Chapter 9: We propose a self-supervised methodology for learning a viewpoint-invariant embedding of human poses from video sources. Our approach is simple and inexpensive as the only required training data are synchronized videos from multiple uncalibrated cameras. We explore the data efficiency, accuracy, and viewpoint invariance of the learned embedding applied to the task of regressing three-dimensional human pose from monocular images.

Finally, in **Chapter 10** we summarize the contributions and novel observations supported by the research conducted throughout this Thesis, and in **Chapter 11** we outline a few promising directions of investigation which we are planning on pursuing, such as discussing the framework in which we believe our work can be used for controlling the visuomotor skills of a robot.

Part I

Scene Understanding

Chapter 4

DESCRIBING COMMON HUMAN VISUAL ACTIONS

The contents of this chapter are adapted from the peer-reviewed publication:

M. R. Ronchi and P. Perona *"Describing Common Human Visual Actions in Images."* 26th British Machine Vision Conference (2015, Swansea, Wales).
DOI: 10.5244/C.29.52
URL: http://www.vision.caltech.edu/~mronchi/projects/Cocoa

WHICH common human actions and interactions are recognizable in monocular still images? Which involve objects and/or other people? How many actions does a person performing at a time? We address these questions by exploring the actions and interactions that are detectable in the images of the COCO dataset. We make two main contributions. First, a list of 140 common 'visual actions' obtained by analyzing the largest on-line verb lexicon currently available for English (VerbNet) and human sentences used to describe images in COCO. Second, a complete set of annotations for those 'visual actions' composed of subject-object and associated verb, which we call COCO-a (a for 'actions'). COCO-a is larger than existing action datasets in terms of number instances of actions, and is unique because it is data-driven, rather than experimenter-biased. Other unique features are that it is exhaustive, and that all subjects and objects are localized. A statistical analysis of the accuracy of our annotations and of each action, interaction and subject-object combination is provided.

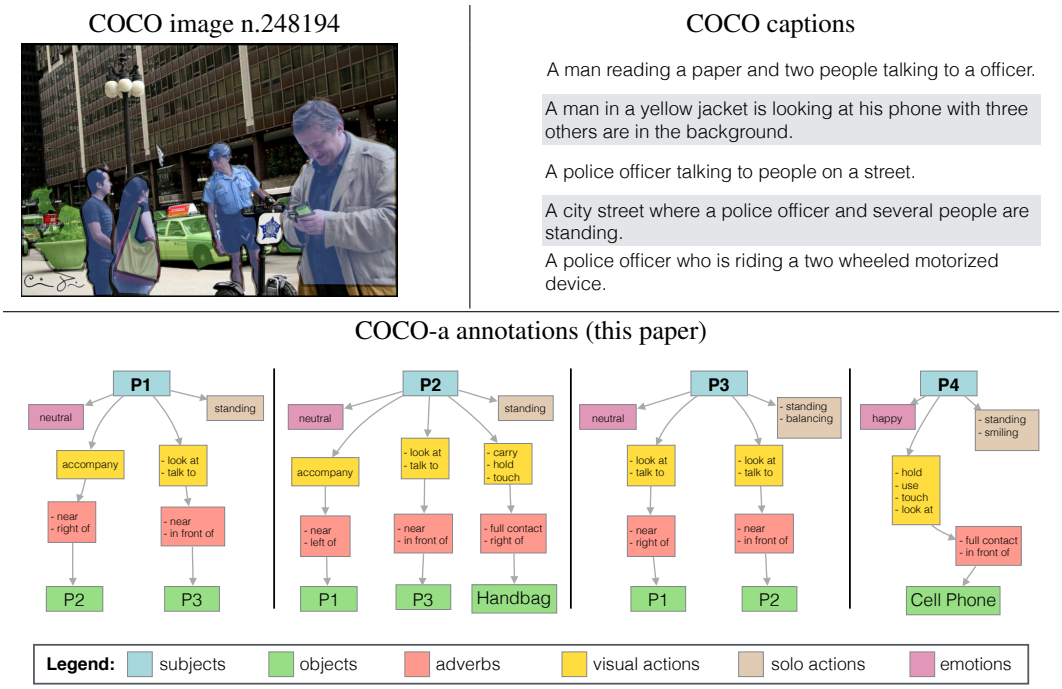


Figure 4.1: **Example annotation from the COCO-a dataset**. (Top) COCO image with its corresponding captions. (Bottom) COCO-a annotations. The annotations are organized by subject: each person in the image (P1–P4, from left to right) can be represented in turn as a subject (blue node) or an object (green node). Each subject and subject-object pair is associated to states and actions (colored boxes containing the collected annotations). Each action is associated to one of the 140 visual actions defined in our Visual VerbNet (VVN), Section 4.4.1

4.1 Introduction

Vision, according to Marr, is "to know what is where by looking." This is a felicitous definition, but there is more to scene understanding than 'what' and 'where': there are also 'who', 'whom', 'when' and 'how'. Besides recognizing objects and estimating shape and location, we wish to detect agents, understand their actions and plans, estimate what and whom they are interacting with, reason about cause and effect, and predict what will happen next.

The idea that actions are an important component of 'scene understanding' in computer vision dates back at least to the '80s [1, 2]. In order to detect actions alongside objects, the relationships between those objects needs to be discovered. For each action, the roles of 'subject' (active agent) and 'object' (passive - whether thing or person) have to be identified. This information may be expressed as a 'semantic network' [3], which is the first useful output of a vision system for scene

understanding.[1] Further steps in scene understanding include assessing causality and predicting intents and future events. It may be argued that producing a full-fledged semantic network for the entire scene may not be necessary in answering questions about the image, as in the Visual Turing Test [4], or in producing output in natural language form. One of the goals of the present study is to ground this debate in data and make the discussion more empirical and less philosophical.

Three main challenges face us in approaching scene understanding. (1) Deciding the nature of the representation that needs to be produced (e.g. there is still disagreement on whether actions should be viewed as arcs or nodes in the semantic network). (2) Designing algorithms that will analyze the image and produce the desired representation. (3) Learning – most of the algorithms that are involved have a considerable number of free parameters. In the way of each one of these steps is a dearth of annotated data. The ideal dataset to guide our next steps has four desiderata: (a) it is *representative* of the pictures we collect every day; (b) it is *richly and accurately annotated* with the type of information we would like our systems to know about; (c) it is *not biased* by a particular approach to scene understanding, rather it is collected and annotated independently of any specific computational approach; (d) it is *large*, containing sufficient data to train the large numbers of parameters that are present in today's algorithms. Current datasets do not measure up to one or more of these criteria. Our goal is to fill this gap. In the present study, we focus on actions that may be detected from single images (rather than video). We explore the visual actions that are present in the recently collected COCO image dataset [5]. The COCO dataset is large, finely annotated and focused on 81 commonly occurring objects and their typical surroundings.

By studying the visual actions in COCO, we make two main contributions:

1. An **unbiased method for estimating actions**, where the data tells us which actions occur, rather than starting from an arbitrary list of actions and collecting images that represent them. We are thus able to explore the type, number and frequency of the actions that occur in common images. The outcome of this analysis is **Visual VerbNet (VVN)** listing the 140 common actions that are visually detectable in images.

[1] While there is broad agreement that the knowledge produced by a 'scene understanding' algorithm should take the form of a graph, the exact definition and the name of this graph have not yet been settled. Other popular choices are 'parse network', 'knowledge graph', 'scene graph'.

2. A **large and well annotated dataset of actions** on the current best image dataset for visual recognition, with rich annotations including all the actions performed by each person in the dataset, and the people and objects that are involved in each action, subject's posture and emotion, and high level visual cues such as mutual position and distance, Figure 4.1.

4.2 Previous Work

Human action recognition has been an important research topic in Computer Vision since the late 80's, and was mainly based on motion/video datasets. Nagel and his collaborators analyzed the German language to detect verbs that refer to actions in urban traffic scenes. They found 119 verbs referring to 67 distinct actions [6, 7], a complete description of actions in a well-defined environment of practical relevance. Early work on human action detection focused on detecting actions as spatio-temporal patterns [8, 9]. Datasets collected in the early 2000s reflect this interest. A popular example is the *KTH* dataset [10] containing videos of people performing 6 actions (no interaction with objects and other people). Laptev and collaborators [11] collected the *Hollywood* dataset culling video of 12 human actions from commercial movies, thus mitigating experimenter bias from acting and filming.

Exploring actions in still images [12] is very valuable given the prevalence and convenience of still pictures. It presents additional challenges – detecting humans, and computing their pose, is more difficult than in video, and the direction of motion is not available making some actions ambiguous (e.g. picking up versus putting down a pen on a desk). State-of-the-art datasets are summarized in Table 4.1.

Everingham and collaborators annotated the PASCAL dataset with 10 actions [13] as a part of the PASCAL-VOC competition. The dataset contains images from multiple sources. The dataset is annotated for objects, and contains a point location for human bodies. Fei-Fei and collaborators collected the Stanford 40 Action Dataset with images of humans performing 40 actions [14]. All images were obtained from Google, Bing, and Flickr. The person performing the action is identified by a bounding box, but objects are not localized. There are 9532 images in total and between 180 and 300 images per action class. Le et al., in their 89 Actions Dataset [15], selected all the images in PASCAL representing a human action and assembled a dataset of 2038 images, which they manually annotated with a verb. The dataset contains 19 objects and 36 verbs, which are combined to form 89 actions.

Dataset	Images	Actions	Per Image Statistics				
			Subjects	Objects	Interactions	Actions	Adverbs
Pascal [13]	9100	10	1	1	x	1	x
Stanford 40 [14]	9532	40	1	1	x	1	x
89 Actions [15]	2038	89	1	1	x	1	x
TUHOI [16]	10805	2974	1.8	-	x	4.8	x
Our work	10000	140	2.2	5.2	5.8	11.1	9.6

Table 4.1: **State-of-the-art datasets for single-frame action recognition.** We indicate with 'x' quantities that are not annotated, with '-' statistics that are not reported. The meaning of Interactions and Adverbs is explained in Section 4.4.

The COCO dataset has been annotated with five captions per image [5], which provides information on actions. These annotations have many good properties: they are data-driven and unbiased; easy and inexpensive to collect; and intuitive and familiar for human interpretation. However, from the point of view of training algorithms for action recognition, there are significant drawbacks: captions don't specify where things are in the image; captions focus typically on one action (which is a very incomplete description of the image); natural language is ambiguous and still difficult to analyze automatically. For these reasons, the COCO captions are not sufficient to inform research on action recognition. The closest work to our own is a dataset called TUHOI [16], which extends the ImageNet [17] annotations by providing descriptions of the actions contained in the images. However, the action annotations are obtained as free-typed verbs, which introduces many ambiguities (such as synonyms) and does not control the specificity of the verbs and on their being visually discriminable – more on this in the next Section.

In the present study, we make a number of steps forward. First, we derive the actions from the data rather than imposing a pre-defined set. Second, we collect data in the form of semantic networks, in which active entities and all the objects they are interacting with are represented as connected nodes. Each agent-object pair is labelled with the set of relevant actions; each agent is also labelled with 'solo' actions such as posture and motion. Emotional state of the agent, relative location and distance at which interactions occur are also recorded. The advantages of this representation over natural language captions can be seen in Figure 4.1.

4.3 Framework

It is important to keep the distinction straight between 'verbs' and 'actions'. Verbs are words and actions are states and events. According to the dictionary, a verb is "a word used to describe an action, state, or occurrence". By contrast, an action is "the fact or process of doing something". Thus *verbs* are words that are used to denote *actions*. Unfortunately, the correspondence between verbs and actions is not one-to-one. For example, the verb *spread* may denote the action of spreading jam on a toast using a knife, or may describe the action carried out by a group of people who part ways simultaneously. Same word, different actions. Conversely, *to spread* (in the culinary sense) becomes *to butter* when what is being spread is butter. Two words for the same action. Furthermore, some actions may be denoted by a single word, *surf* or *golf*, and others may require a few words, *play tennis* and *ride a bicycle*. For simplicity we will call 'verb' all the expressions that describe actions, whether single or multi-worded.

Actions are not equal in length and complexity. It has been pointed out that one may distinguish between 'movemes', 'actions', and 'activities' [18, 19] depending on structure, complexity, and duration. For example: *reach* is a moveme (a brief target-directed ballistic motion), *drink from a glass* is an action (a concatenation of movemes: reach the glass, grasp its stem, lift the glass to the lips, etc.), while *dine* is an activity (a stochastic concatenation of actions taking place over a stretch of time). Here, we do not distinguish between movemes, actions and activities because in still images the extent in time and complexity is not directly observable.

We call '**visual action**' an action, state, or occurrence that has a unique and unambiguous visual connotation, making it detectable and classifiable; i.e., *lay down* is a visual action, while *relax* is not. A visual action may be discriminable only from video data, 'multi-frame visual action' such as *open* and *close*, or from monocular still images, 'single-frame visual action' (simply 'visual action' throughout the rest of this dissertation), such as *stand*, *eat* and *play tennis*. In order to label visual actions, we will use the verbs that come readily to mind to a native English speaker, a concept akin to *entry-level categorization* for objects [20]. Based on this criterion, sometimes we prefer more general visual actions (e.g. *play tennis*) rather than the sports domain specific ones such as *volley* or *serve*, and *drink* rather than more specific 'movemes' such as *lift a glass to the lips*), other times more specific ones (e.g. *shaking hands* instead of more generally *greet*).

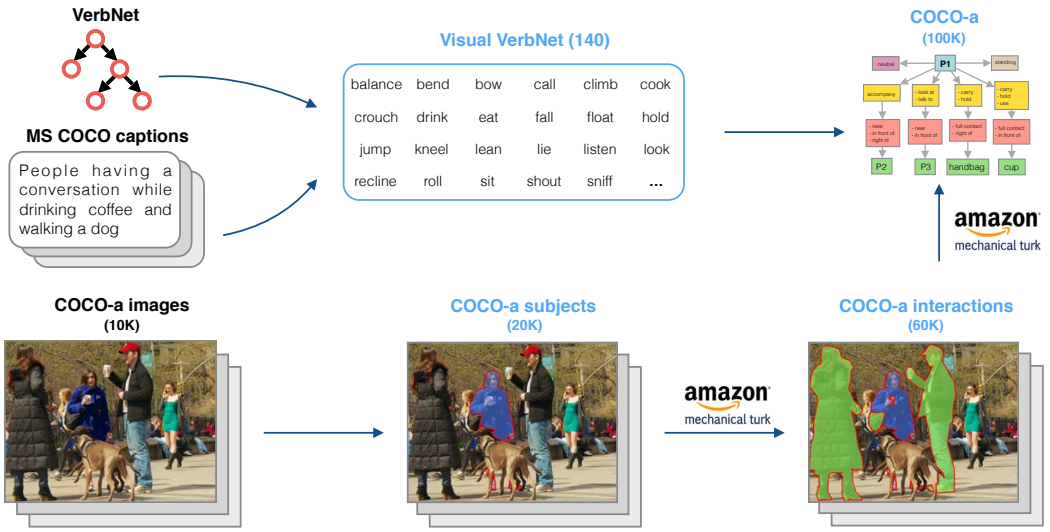


Figure 4.2: **Steps in the collection of the COCO-a dataset.** From VerbNet and COCO captions, we extracted a list of 140 visual actions. Persons that are annotated in the COCO images were considered as potential 'subjects' of actions, and AMT workers annotated all the objects they interact with, and assigned the corresponding visual actions. Titles in light blue indicate the components of the dataset, and their size is shown in parenthesis. COCO image n.118697 is used in this visualization.

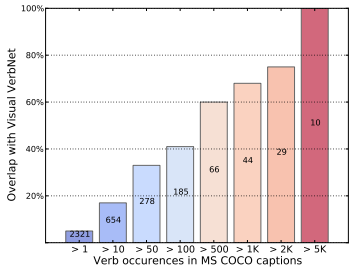
While taxonomization has been adopted as an adequate means of organizing object categories (e.g. animal → mammal → dog → dalmatian), and shallow taxonomies are available for verbs in VerbNet [21], we are not interested in fine-grained categorization for the time being, thus there are no taxonomies in our set of visual actions.

4.4 Dataset Collection

Our goal is to collect an unbiased dataset with a large amount of meaningful and detectable interactions involving human agents as subjects. Our focus is on humans given the large variety of actions they perform and great availability of data, but we will consider extending our collection to other agents and objects in the future.
We put together a process, exemplified in Figure 4.2, consisting of four steps:

1. Obtain the list of common visual actions that are observed in everyday images, Section 4.4.1.
2. Identify the people who are carrying out actions (the subjects), Section 4.4.2.
3. For each subject, identify the objects that he/she is interacting with, Section 4.4.3.
4. For each subject-object pair, identify the relevant actions, Section 4.4.4.

accompany	chew	exchange	jump	pay	punch	sing	swim
avoid	clap	fall	kick	perch	push	sit	talk
balance	clear	feed	kill	pet	put	**skate**	taste
bend (pose)	climb	fight	kiss	photograph	reach	**ski**	teach
bend (something)	cook	fill	kneel	pinch	read	slap	**throw**
be with	crouch	float	laugh	play	recline	sleep	tickle
bite	cry	fly	lay	play baseball	remove	smile	touch
blow	cut	follow	lean	play basketball	repair	sniff	use
bow	dance	get	lick	play frisbee	ride	snowboard	walk
break	devour	give	lie	play instrument	roll	spill	wash
brush	dine	groan	lift	play soccer	row	spray	wear
build	disassemble	groom	**light**	play tennis	run	spread	whistle
bump	draw	hang	listen	poke	sail	squat	wink
call	dress	help	look	pose	separate	squeeze	write
caress	drink	hit	massage	pour	shake hands	stand	
carry	drive	hold	meet	precede	shout	steal	
catch	drop	hug	mix	prepare	show	straddle	
chase	eat	hunt	**paint**	pull	signal	surf	

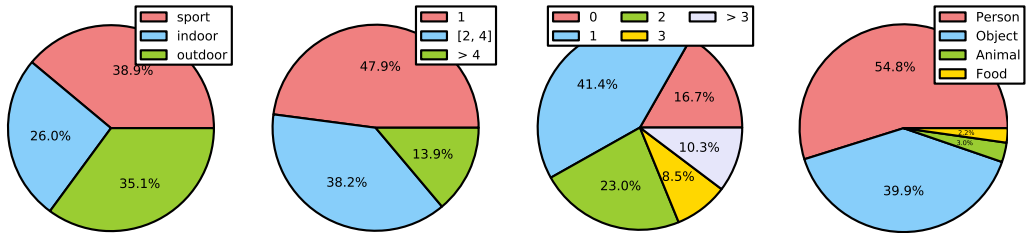


Not Visual Actions				Mutli-frame Visual Actions			Single-Frame Visual Actions		Synonyms				Domain-Specific
attempt	engage	practice	share	approach	leave	start	cover	tie	adjust	gather	paddle	stare	bowl
board	enjoy	prop	stick	block	miss	step	face	wrap	attach	grab	pass	stuff	grind
celebrate	extend	race	stretch	close	move	stop	line		color	hand	pick	take	park
check	feature	reflect	top	come	open	turn	load		crowd	lead	place	toss	pitch
compete	include	relax	travel	cross	raise		slide		display	leap	say	watch	set
contain	learn	rest	try	enter	return		sprinkle		dock	make	see	wave	swing
decorate	live	seem	wait	flip	seat		stack		handle	mount	slice		tow
double	perform	shape		head	shake		surround		fix	observe	speak		

Figure 4.3: **Visual VerbNet (VVN).** (Top-Left) List of the 140 visual actions that constitute VVN – bold ones were added after the comparison with COCO captions. (Top-Right) There is 60% overlap for the 66 verbs in VVN (of the total 2321 in COCO captions) with > 500 occurrences. (Bottom) Verbs with > 100 occurrences in the COCO captions not contained in VVN, organized in categories.

4.4.1 Visual VerbNet

To obtain the list of the entry-level visual actions, we examined VerbNet [21] (containing > 8000 verbs organized in about 300 classes) and selected all the verbs that refer to visually identifiable actions. Our criteria of selection is that we would expect a 6–8 year old child to be able to easily distinguish visually between them. This criterion led us to group synonyms and quasi-synonyms (*speak* and *talk*, *give* and *hand*, etc.) and to eliminate verbs that were domain-specific (*volley*, *serve*, etc.) or rare (*cover*, *sprinkle*, etc.). To be sure that we were not missing any important actions, we also analyzed the verbs in the captions of the images containing humans in the COCO dataset, and discarded verbs not referring to human actions, without a clear visual connotation, or synonyms. This resulted in adding six additional verbs to our list for a total of 140 visual actions, shown in Figure 4.3-(top-left). Figure 4.3-(top-right) explores the overlap of VVN with the verbs in COCO captions. The overlap is high for verbs that have many occurrences, and verbs that appear in the COCO captions but not in VVN do not denote a visual action, are synonyms of a visual action contained already in VVN, or refer to actions that are either very domain-specific or highly unusual, as shown in the table in Figure 4.3-(bottom). The process we followed ensured an unbiased selection of visual actions. Furthermore, we asked Amazon Mechanical Turk (AMT) workers for feedback on the completeness of this list and, given their scant response, we believe that VVN is



(a) (b) (c) (d)

Figure 4.4: **Statistics from the COCO-a dataset.** The distribution of (a) the type of locations for all the images contained in the dataset, (b) the number of subjects portrayed per image, and (c) the number of objects that a subject is interacting with. (d) The distribution of the category of the interacting objects.

very close to complete, and does not need extension unless specific domain action recognition is required. The goal of VVN is not to impose a strict ontology on the annotations that will be collected, but rather to set a starting point for a systematic analysis of actions in images and limit the effect of the many ambiguities that are present in natural language.

4.4.2 Image and Subject Selection

Different actions usually occur in different environments, so in order to balance the content of our dataset, we selected an approximately equal number of images of three types of scenes: sports, outdoors, and indoors, as shown in Figure 4.4-(a). Furthermore, the type of actions performed is strongly correlated to the number of people interacting. As a result, we selected images of various complexity containing only a single subject, small groups (2-4 subjects) and crowds (>4 subjects), Figure 4.4-(b). From the selected images, all the people whose pixel area is larger than 1600 pixels are defined as 'subjects'. All the people in an image, regardless of size, are still considered as possible objects of an interaction. The result of this preliminary image analysis is an intermediate dataset containing about two subjects per image, indicated as 'COCO-a subjects' in Figure 4.2.

4.4.3 Interactions Annotations

For each subject, we collected labels describing all the objects that he/she is interacting with. In order to do so, we presented annotators with images such as in Figure 4.5-(left), containing only one 'subject' (the person highlighted in blue) and a multitude of potential 'objects' (hightlighted in white), and asked them to either (1)

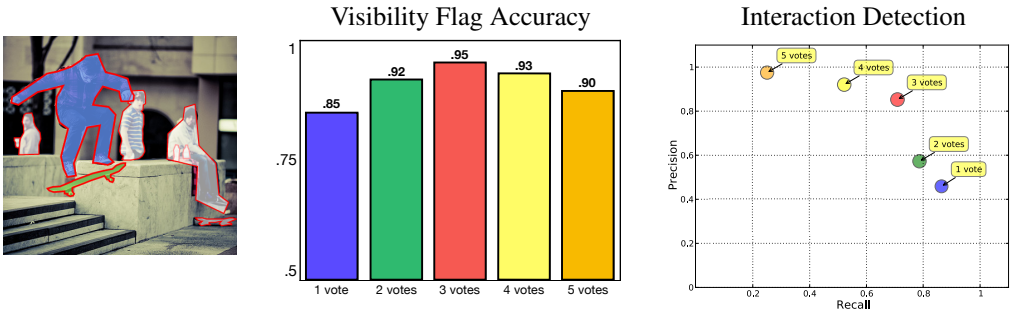


Figure 4.5: **Quality evaluation of the COCO-a annotations.** (Left) Example image presented to the AMT workers with the 'subject' highlighted in blue and all the 'objects' in white. When an object is labeled as part of an interaction, it is highlighted in green. Annotators' accuracy for the task of 'subject' visibility flag classification (Center) and Precision and Recall curve for the task of 'subject-object' interaction detection (Right). We measure the performance of annotators by consolidating the answers of multiple annotators using a number of votes ranging from 1 to 5, and comparing the obtained label to the annotation by the authors on a subset of images.

flag the subject if it was mostly occluded or invisible; or (2) click on all the objects (including other people) that he/she is interacting with.

Deciding if a person is occluded and if it is interacting with an object (or other person) is somewhat subjective, so we asked 5 workers to analyze each image and combined their responses requiring a number of votes ranging from 1 to 5. The loosest strategy ('1 vote') required only one out of the five annotators to say that a 'subject' was occluded or that a 'subject-object' pair was interacting for that label to be assigned, while the most strict strategy ('5 votes') required complete agreement between all workers. In order to assess the quality of the responses and determine the best method for combining them into a unique label, we also collected ground-truth answers from one of the authors on a subset of the images. 1) We measured the accuracy (percentage agreement) between the author's answer and the combined annotations in assessing the visibility flag of the 'subject' in the image. 2) We framed the annotation of a 'subject-object' interaction pair as a detection problem, and defined the following quantities: *True Positives* - interactions identified by both the author and the annotators; *False Positives* - interactions disregarded by the author, but identified by the annotators; *False Negatives* - interactions identified by the author and missed by the annotators.

Based on the above definitions, it is possible to measure the Precision and Recall trade-off for all possible combination strategies. We found that requiring three votes yielded the highest flag agreement of about 95%, Figure 4.5-(center), and the

best trade-off between Precision and Recall, Figure 4.5-(right). After discarding the flagged subjects and consolidating the interaction annotations, we obtained an average of 5.8 interactions per image, which constitute the 'COCO-a interactions' dataset, Figure 4.2.

As shown in Figure 4.4-(c), about 1/5 of the subjects is performing 'solo' visual actions (not interacting with any objects), while 2/5 is involved in an interaction with a single object, and 2/5 interact with two or more objects (Figure 4.1 contains examples of subjects interacting with two and three objects). Figure 4.5-(d) suggests that our dataset is human-centric, since more than half of the interactions happen with other people.

4.4.4 Visual Actions Annotations

In the final step of our process, we labelled all the subject-object interactions in the COCO-a interactions dataset with the visual actions in VVN. Workers were presented with an image containing a single highlighted interaction, as visualized in Figure 4.5-(left), and asked to select all the visual actions describing it.

We organized the set of visual actions in VVN into 8 groups – '*posture/motion*', '*solo actions*', '*contact actions*', '*actions with objects*', '*social actions*', '*nutrition actions*', '*communication actions*', '*perception actions*'. This was based on two simple rules: (a) actions in the same group share some important property, for instance being performed 'solo' (*cry*, *clap hands*), with 'objects' (*cut*, *ride a bicycle*), or being 'social' in the sense of requiring the involvement of other people (*shake hands*, *feed*); (b) actions in the same group tend to be mutually exclusive, i.e. a person can be drinking or eating at a certain moment, but not both, while 'visual actions' of different categories may co-occur, i.e. one may be eating while standing.

Furthermore, we included in our study 3 'adverb' categories: '*emotion*' of the subject.[2], '*location*' and '*relative distance*' of the object with respect to the subject[3] This allowed us to obtain a rich set of annotations for all the actions that a subject is performing which completely describe his/her state, a property that is novel with respect to existing datasets and favours the construction of semantic networks

[2]Despite the disagreement on the fact that humans might have basic discrete emotions [22, 23], we adopt Ekman's 6 basic emotions [24] for this study as we are interested in a high level description of subject's emotional state.

[3]Some actions require a special mutual position: *sit* implies that the sitter is above, as well as in contact with, the thing that he sits on. Other actions require proximity and no contact.

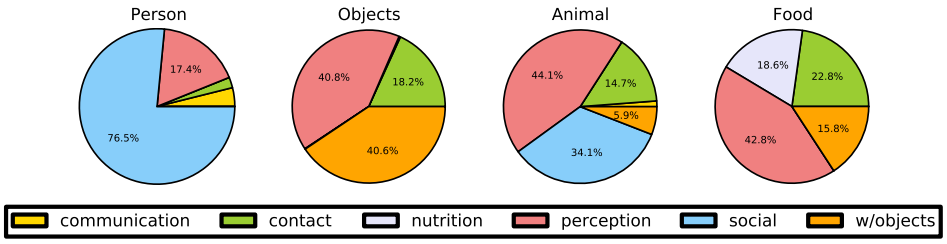


Figure 4.6: **Distribution of the visual actions based on object category.** Fraction of visual actions that belong to each macro category (excluding posture and solo actions) when subjects interact with People, Animals, Objects or Food.

centered on the subject. Tables 4.2 and 4.3 contain a break down of the visual actions and adverbs into the categories that were presented to the AMT workers.

We asked three annotators to select all the visual actions and adverbs that describe each subject-object interaction pair. In some cases, annotators interpreted interactions differently, but still correctly, so we return all the visual actions collected for each interaction along with the value of agreement between the annotators, rather than forcing a deterministic, but arbitrary, ground truth. Depending on the application that will be using our data, it will be possible to consider visual actions on which all the annotators agree or only a subset of them. The average number of visual action annotations provided per image for an agreement of 1, 2 or all 3 annotators is 19.2, 11.1, and 6.1, respectively.

This constitutes the content of the COCO-a dataset in its final form.

4.5 Analysis

Figure 4.1 allows a first qualitative analysis of the COCO-a dataset. Compared with COCO captions, COCO-a annotations contain additional information by providing: *i)* a complete account of all the subjects, objects and actions contained in an image; *ii)* an unambiguous and machine-friendly form; and *iii)* the specific localization in the image for each subject and object. Statistics of the information that the COCO-a dataset annotations capture and convey for each image is summarized in Table 4.1.

In Figure 4.6, we see the most frequent types of actions carried out when subjects interact with four specific object categories: other people, animals, inanimate objects (such as a handbag or a chair), and food. For interactions with people, the visual actions belong mostly to the categories '*social*' and '*perception*'. When subjects interact with animals, the visual actions are similar to those with people, except

Visual Actions		
Posture / Motion (23)	Communication (6)	Contact (22)
balance, bend, bow, climb, crouch, fall, float, fly, hang, jump, kneel, lean, lie, perch, recline, roll, run, sit, squat, stand, straddle, swim, walk	call, shout, signal, talk, whistle, wink	avoid, bit, bump, caress, hit, hold, hug, kick, kiss, lick, lift, massage, pet, pinch, poke, pull, punch, push, reach, slap, squeeze, tickle
Social (24*)	Perception (5)	Nutrition (7)
accompany, be with, chase, dance, dine, dress, feed, fight, follow, give, groom, help, hunt, kill, meet, pay, shake hands, teach, play baseball, play basketball, play frisbee, play soccer, play tennis, precede	listen, look, sniff, taste, touch	chew, cook, devour, drink, eat, prepare, spread
Solo (24*)	With objects (34)	
blow, clap, cry, draw, groan, laugh, paint, photograph, play, play baseball, play basketball, play frisbee, play soccer, play tennis, play instrument, pose, sing, sleep, smile, write, skate, ski, snowboard, surf	bend, break, brush, build, carry, catch, clear, cut, disassemble, drive, drop, exchange, fill, get, lay, light, mix, pour, read, remove, repair, ride, row, sail, separate, show, spill, spray, steal, put, throw, use, wash, wear	

Table 4.2: **List of visual actions in COCO-a grouped by category.** The complete list of visual actions contained in Visual VerbNet. Visual actions in one category are usually mutually exclusive, while visual actions of different categories may co-occur. (*) There are five visual actions (*play baseball*, *play basketball*, *play frisbee*, *play soccer*, *play tennis*) that are considered both '*social*' and '*solo*' types of actions.

Adverbs		
Emotion (6)	Relative Location (6)	Relative Distance (4)
anger	above	far
disgust	behind	full contact
fear	below	light contact
happiness	in front	near
sadness	left	
surprise	right	

Table 4.3: **List of visual adverbs in COCO-a grouped by category.** The complete list of high level visual cues collected, describing the subjects (emotion) and localization of the interaction (relative location and distance).

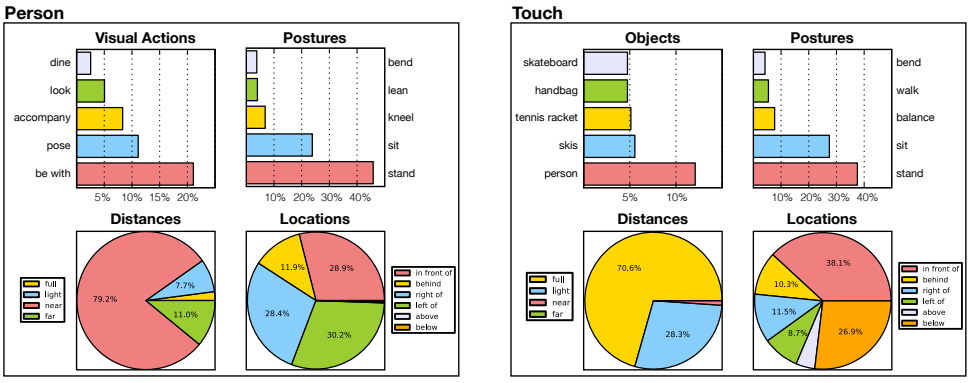


Figure 4.7: **Analysis of the visual actions performed by people and the most common visual action in the COCO-a dataset.** (Left) Top visual actions, postures, distances and relative locations of person/person interactions. (Right) Objects, postures, distances and locations that are most commonly associated with the visual action 'touch'.

there are fewer '*social*' actions and more '*perception*' actions. Person and animal are the only types of objects for which the '*communication*' visual actions are used at all. When people interact with objects, the visual actions used to describe those interactions are mainly from the categories '*with objects*' and '*perception*'. As expected, food items are the only ones for which visual actions of '*nutrition*' category are selected.

Figure 4.8-(top) shows the full lists of objects that people interact with, and the number of interactions. There are 29 objects with more than 100 interactions in the images contained in the COCO-a dataset.

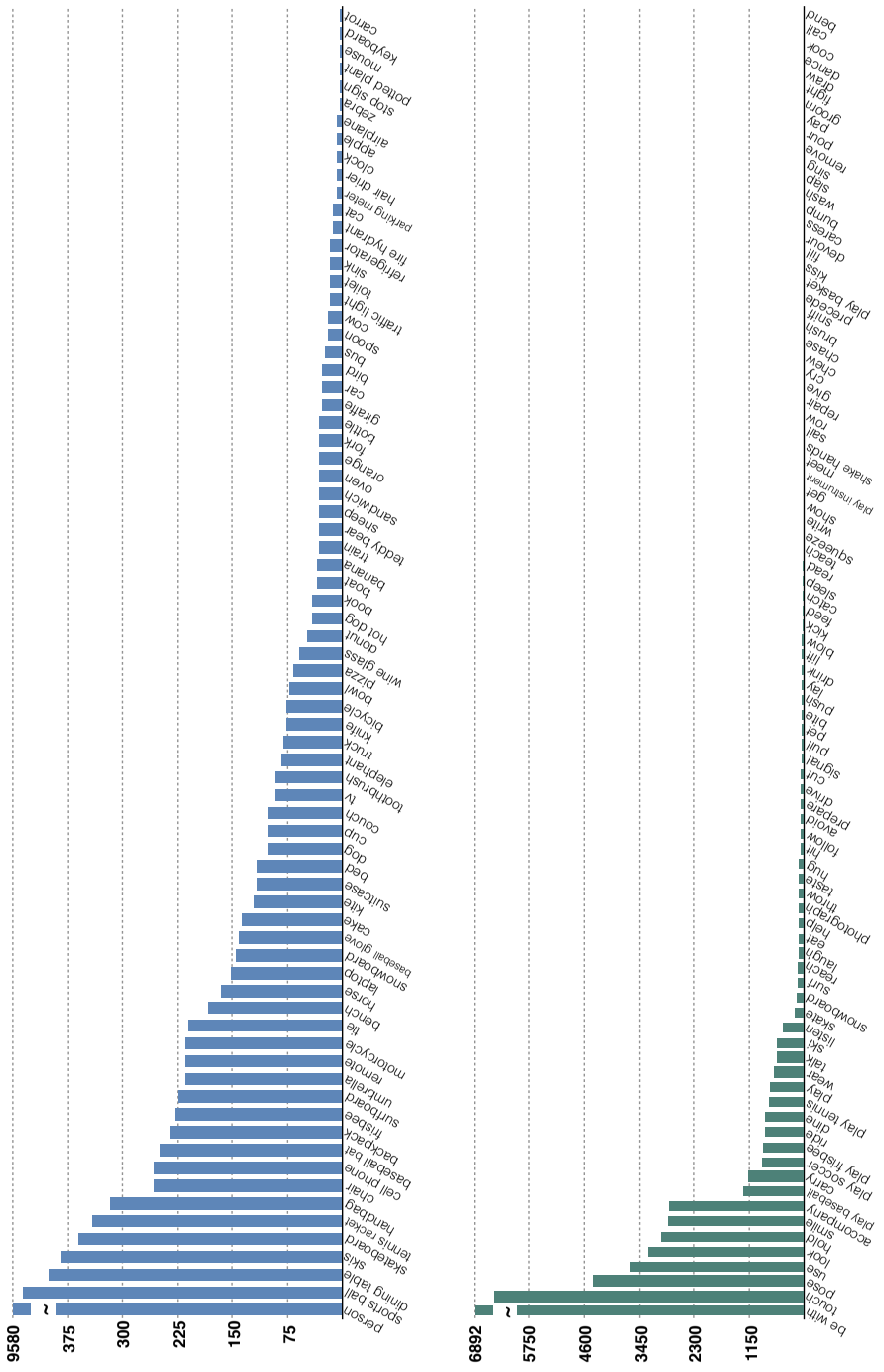


Figure 4.8: **Histogram of counts for objects and visual actions in the COCO-a dataset.** The frequency distribution of interacting objects obtained (Top) and visual actions (Bottom) that people perform in the COCO-a dataset, as obtained through the annotation process. The distributions are long-tailed with a fairly steep slope, and there are only 29 objects and 31 visual actions having more than 100 occurrences.

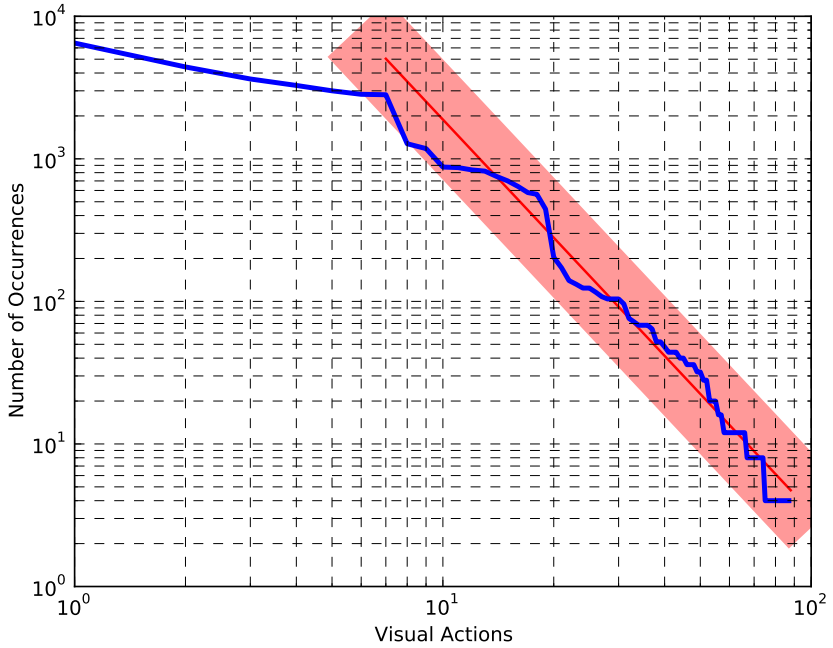


Figure 4.9: **Heavy tail analysis of the visual actions.** The plot in log-log scale of the list of visual actions against the number of occurrences.

The human-centric nature of the dataset is confirmed by the fact that the most frequent object of interaction is other persons, an order of magnitude more than any other objects. Since our dataset contains an equal number of sports, outdoor and indoor scenes, the list of objects is heterogeneous and contains objects that can be found in all environments.

In Figure 4.8-(bottom), we show the complete list of visual actions annotated from the images and their occurrences. There are only 31 visual actions that have more than 100 occurrences, with 90% of the actions having less than 2000 occurrences and covering about 27% of the total count of visual actions. The distribution of the visual actions' counts follows a heavy tail distribution, to which we fit a line, shown in Figure 4.9, with slope $\alpha \sim -3$. This leads to the observation that COCO dataset is sufficient for a thorough representation and study of about 20 to 30 visual actions. However, we are considering methods to bias our image selection process in order to obtain more samples of the actions contained in the tail.

The most frequent visual action in our dataset is '*be with*'. This is a very particular visual action as annotators use it to specify when people belong to the same group. Common images often contain multiple people involved in different group actions, and this annotation can provide insights in learning concepts such as the difference

between proximity and interaction – i.e. two people back to back are probably not part of the same group although spatially close.

The COCO-a dataset contains a rich set of annotations. We provide two examples of the information that can be extracted and explored, for an object and a visual action contained in the dataset. Figure 4.7-(left) describes interactions between people. We list the most frequent visual actions that people perform together (*be in the same group*, *pose* for pictures, *accompany* each other, etc.), postures that are held (stand, sit, kneel, etc.), distances of interaction (people mainly interact near each other, or from far away if they are playing some sports together) and locations (people are located about equally in front or to each other's sides, more rarely behind and almost never above or below each other). A similar analysis can be carried out for the visual action *touch*, Figure 4.7-(right). The most frequently touched object are other people, sports and wearable items. People touch things mainly when they are standing or sitting (for instance a chair or a table in front of them). As expected, the distribution of locations is very skewed, as people are almost always in full or in light contact when touching an object and never far away from it. The location of objects shows us that people in images usually touch things in front (as comes natural in the action of grasping something) or below of them (such as a chair or bench when sitting).

To explore the expressive power of our annotations, we decided to query rare types of interactions and visualize the images retrieved. Figure 4.10 shows the result of querying our dataset for visual actions with rare emotion, posture, position or location combinations. The format of the annotations allows to query for images by specifying at the same time multiple properties of the interactions and their combinations, making them well suited for the training of image retrieval systems.

4.6 Discussion and Conclusions

By a combined analysis of VerbNet and COCO captions, we were able to compile a list of the main 140 visual actions that take place in common scenes. Our list, which we call Visual VerbNet (VVN), attempts to include all actions that are visually discriminable. It avoids verb synonyms, actions that are specific to particular domains, and fine-grained actions. Unlike previous work, Visual VerbNet is not the result of experimenter's choices; rather, it is derived from linguistic analysis (VerbNet) and an existing large dataset of everyday scenes (COCO captions). Our novel dataset, COCO-a, consists of the VVN actions contained in 10,000 COCO images, representative of a wide variety of scenes and situations in which 81 common objects are an-

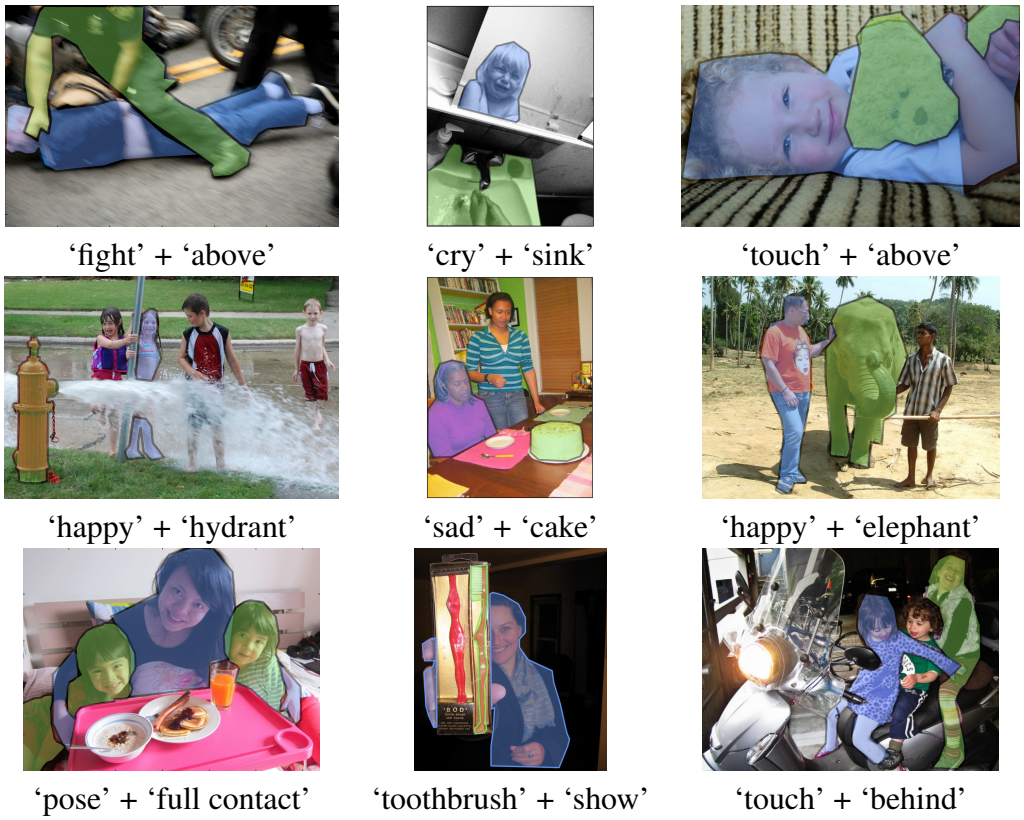


Figure 4.10: **Sample images and rare annotations contained in the COCO-a dataset.** Sample images returned as a result of querying our dataset for visual actions with rare emotion, posture, position or location combinations. Subjects are in blue.

notated with pixel precision segmentations. A key aspect of our annotations is that they are complete. First, each person in each image is identified as a possible subject, or active agent of some action. Second, for each agent, the set of objects that he/she is interacting with is identified. Third, for each agent-object pair (and each single agent), all the possible interactions involving that pair are identified, along with high level visual cues, such as emotion and posture, spatial relationship and distance. The analysis of our annotations suggests that our collection of images ought to be augmented with an eye to increasing representation for the VVN actions that are less frequent in COCO. We hope that our dataset will provide researchers with a starting point for conceptualizing actions in images: which representations are most suitable, which algorithms should be used. We also hope that it will provide an ambitious benchmark on which to train and test algorithms. Amongst applications that are enabled by this dataset are building visual Q&A systems [4, 25], more sophisticated image retrieval systems, and automated analysis of actions in images of social media.

References

[1] H.-H. Nagel, "From image sequences towards conceptual descriptions", *Image and vision computing*, vol. 6, no. 2, pp. 59–74, 1988 (cit. on p. 17).

[2] H.-H. Nagel, "A vision of 'vision and language' comprises action: An example from road traffic", *Artificial Intelligence Review*, vol. 8, no. 2-3, pp. 189–214, 1994 (cit. on p. 17).

[3] S. Russell and P. Norvig, *Artificial intelligence, a modern approach*. Prentice-Hall, Egnlewood Cliffs, 1995 (cit. on p. 17).

[4] D. Geman, S. Geman, N. Hallonquist, and L. Younes, "A visual turing test for computer vision system", *Proceedings of the National Academy of Sciences (PNAS)*, 2015 (cit. on pp. 18, 33).

[5] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context", in *Computer Vision–ECCV 2014*, Springer, 2014, pp. 740–755 (cit. on pp. 18, 20).

[6] U. C. von Seelen, "Ein formalismus zur beschreibung von bewegungsverben mit hilfe von trajektorien", PhD thesis, Diplomarbeit, Fakultaet fuer Informatik der Universitaet Karlsruhe, 1988 (cit. on p. 19).

[7] D. Koller, N. Heinze, and H. Nagel, "Algorithmic characterization of vehicle trajectories from image sequences by motion verbs", in *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, IEEE, 1991, pp. 90–95 (cit. on p. 19).

[8] R. Polana and R. C. Nelson, "Recognition of motion from temporal texture", in *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92., 1992 IEEE Computer Society Conference on*, IEEE, 1992, pp. 129–134 (cit. on p. 19).

[9] K. Rohr, "Towards model-based recognition of human movements in image sequences", *CVGIP: Image understanding*, vol. 59, no. 1, pp. 94–115, 1994 (cit. on p. 19).

[10] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local svm approach", in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, IEEE, vol. 3, 2004, pp. 32–36 (cit. on p. 19).

[11] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies", in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, IEEE, 2008, pp. 1–8 (cit. on p. 19).

[12] G. Guo and A. Lai, "A survey on still image based human action recognition", *Pattern Recognition*, vol. 47, no. 10, pp. 3343–3361, 2014 (cit. on p. 19).

[13] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, *The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results*, http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html (cit. on pp. 19, 20).

[14] B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. Guibas, and L. Fei-Fei, "Human action recognition by learning bases of action attributes and parts", in *Computer Vision (ICCV), 2011 IEEE International Conference on*, IEEE, 2011, pp. 1331–1338 (cit. on pp. 19, 20).

[15] D.-T. Le, J. R. Uijlings, and R. Bernardi, "Exploiting language models for visual recognition.", in *EMNLP*, 2013, pp. 769–779 (cit. on pp. 19, 20).

[16] D.-T. Le, J. Uijlings, and R. Bernardi, "Tuhoi: Trento universal human object interaction dataset", *V&L Net 2014*, p. 17, 2014 (cit. on p. 20).

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database", in *2009 IEEE conference on computer vision and pattern recognition*, Ieee, 2009, pp. 248–255 (cit. on p. 20).

[18] C. Bregler, "Learning and recognizing human dynamics in video sequences", in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 1997, pp. 568–574 (cit. on p. 21).

[19] D. J. Anderson and P. Perona, "Toward a science of computational ethology", *Neuron*, vol. 84, no. 1, pp. 18–31, 2014 (cit. on p. 21).

[20] S. Palmer, *Vision science: Photons to phenomenology*. MIT Press, 1999 (cit. on p. 21).

[21] K. Kipper, A. Korhonen, N. Ryant, and M. Palmer, "A large-scale classification of english verbs", *Language Resources and Evaluation*, vol. 42, no. 1, pp. 21–40, 2008 (cit. on pp. 22, 23).

[22] A. Ortony and T. J. Turner, "What's basic about basic emotions?", *Psychological review*, vol. 97, no. 3, p. 315, 1990 (cit. on p. 26).

[23] S. Du, Y. Tao, and A. M. Martinez, "Compound facial expressions of emotion", *Proceedings of the National Academy of Sciences*, vol. 111, no. 15, E1454–E1462, 2014 (cit. on p. 26).

[24] P. Ekman, "An argument for basic emotions", *Cognition & emotion*, vol. 6, no. 3-4, pp. 169–200, 1992 (cit. on p. 26).

[25] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh, "VQA: visual question answering", *CoRR*, vol. abs/1505.00468, 2015. [Online]. Available: `http://arxiv.org/abs/1505.00468` (cit. on p. 33).

Chapter 5

DISTANCE ESTIMATION OF AN UNKNOWN PERSON FROM A PORTRAIT

The contents of this chapter are adapted from the peer-reviewed publication:

X. P. Burgos-Artizzu, **M. R. Ronchi** and P. Perona *"Distance Estimation of an Unknown Person from a Portrait."* 13th European Conference on Computer Vision (2014, Zurich, Switzerland).

DOI: 10.1007/978-3-319-10590-1_218

URL: http://www.vision.caltech.edu/~mronchi/projects/FaceDistancePortrait

We propose the first automated method for estimating distance from frontal pictures of unknown faces. Camera calibration is not necessary, nor is the reconstruction of a 3D representation of the shape of the head. Our method is based on estimating automatically the position of face and head landmarks in the image, and then using a regressor to estimate distance from such measurements. We explored which landmarks are more important for this task, and collected and annotated a dataset of frontal portraits of 53 individuals spanning a number of attributes (sex, age, race, hair), each photographed from seven distances. We find that our proposed method outperforms humans performing the same task. We observe that different physiognomies will bias systematically the estimate of distance, i.e. some people look closer than others.

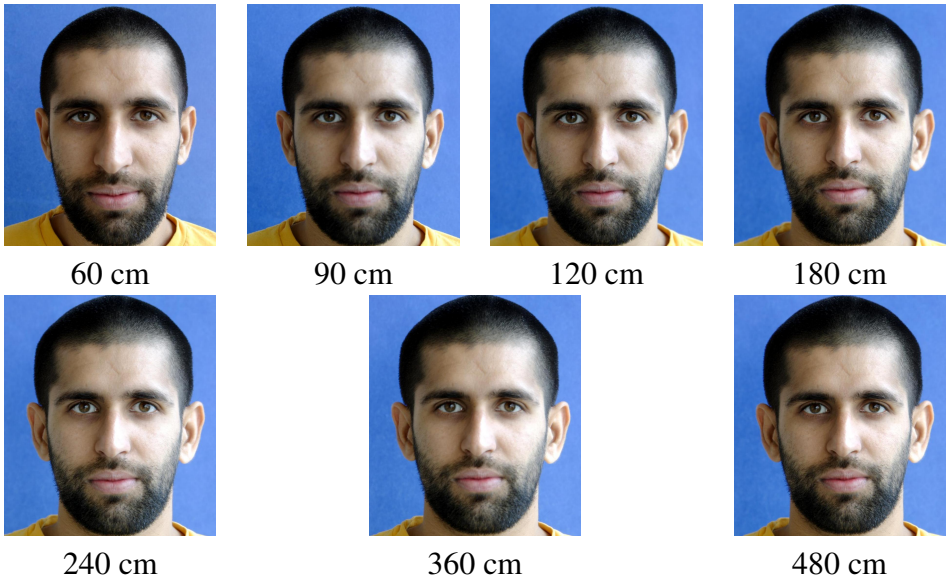


Figure 5.1: **Example annotations from the Caltech Multi-Distance Portraits (CMDP) dataset**. Portrait pictures of a subject taken from 7 different distances ranging between 60 cm (top-left image) and 480 cm (bottom-right image). The effect of perspective, improperly called 'perspective distortion', is clearly noticeable. In portraits taken from a closer distance, the nose and mouth appear bigger, the ears are partially occluded by the cheeks, and the face appears longer. This systematic deformation in the image plane is related to distance. We explore whether, and how accurately, the distance from which the portrait was taken may be estimated from the image when both the person and the camera are unknown.

5.1 Introduction

Consider a standard portrait of a person – either painted or photographed. Can one estimate the distance between the camera (or the eye of the painter) and the face of the sitter? Can one do so accurately even when the camera and the sitter are unknown? These questions are not just academic – we have four applications in mind. First, faces are present in most consumer pictures; if faces could provide a cue to distance, this would be useful for scene analysis. Second, psychologists have pointed out that the distance from which a portrait is captured affects its emotional valence [1]; therefore, estimating this distance from a given picture would provide a cue to automate the assessment of its emotional valence. Third, estimating the distance from which master paintings were produced will provide art historians with useful information on art practices throughout the ages [2]. The fourth potential application is forensics: inconsistency in the distance from which faces were photographed may help reveal photographic forgeries [3].

The most informative visual cues for distance are stereoscopic disparity [4], motion parallax [5],[6] and structured lighting [7, 8]. However, we are interested in the case of a static monocular brightness picture, such as a painting hanging in a museum or a photograph in a newspaper, where none of these cues is available.

The most reliable remaining cue is object familiarity [9]; however, there are several obstacles to a straightforward use of this cue. First, if the camera is unknown one does not have calibration parameters, which rules out straightforward use of the distance of known points, such as the distance between the pupils. Second, when the sitter is unknown only statistical knowledge of his/hers 3D shape is available. However, it is known that one image of a constellation of at least five 3D points whose mutual position is known is sufficient both for camera calibration and pose computation [10], and therefore one would expect that some useful signal is available, see Figure 5.1.

In this work, we study the feasibility and accuracy of automatically estimating the distance of a person from a camera, using a single 2D frontal portrait image without requiring any prior knowledge on the camera used or the person being photographed. Our approach is to first automatically detect facial features and then estimate distance from their mutual positions in the image. Our main contributions are:

1. A novel approach for estimating the camera-head distance from a single 2D portrait photograph when both the camera and the sitter are unknown. Our method yields useful signal and outperforms humans by 16%, see Figure 5.6.

2. The introduction of a new dataset of portraits, *Caltech Multi-Distance Portraits* (**CMDP**), composed of 53 subjects belonging to both sexes, a variety of ages, ethnic backgrounds and physiognomies. Each subject was photographed from seven different distances and each portrait manually labeled with 55 keypoints over the head and face. The dataset is available online.

3. In-depth analysis and discussion of the feasibility of the proposed approach. We study two different variants of the task and analyze what are the most important input visual cues. We compare our method's performance using both machine estimated and ground-truth landmarks against the performance of human observers. Interestingly, we found that the main source of error for both humans and our method is the variability of physiognomies.

5.2 Related Work

Estimating the pose of a human head from an image was explored in [11, 12]. The literature focuses on the estimation of the three degrees of freedom (DOF) - yaw, pitch and roll - under the assumption that the human head can be modeled as a disembodied rigid object. Knowledge of the intrinsic camera parameters or depth information is required.

Psychophysics experiments [13, 14] show that human face recognition performance can be impaired by perspective transformation. As one might expect, the severity of this deficit depends on the difference between the amount of 'perspective distortion' at the learning and testing phases. They also established that both global perspective information and local image similarity features such as ears, eyes, mouth or nose play a fundamental role in this task. Their conclusion is that perspective distortion impairs face recognition, similarly to other visual cues such as lighting and head orientation. This poses the question of whether perspective distortion or, equivalently, distance may be estimated.

Psychologists [1] observed that portrait photographs taken from within personal space elicit lower investments in an economic trust game and lower ratings of social traits such as strength, attractiveness or trustworthiness. These findings could not be explained by width-to-height ratio, explicit knowledge of the camera distance or typicality of the presented faces, thus suggesting the existence of a facial cue influencing social judgments as a function of interpersonal distance. They suggest that there is an "optimal distance" at which portraits should be taken. This idea of choosing the optimal viewpoint and distance to subject is also known to be of great importance in traditional portraiture [2].

To our knowledge, Flores et al. [15] are the first to propose a method that recovers camera distance from a single image of a previously unseen subject. Their work is based on the Efficient Perspective n-Point algorithm (EPnP) [16], a non-iterative solution to the perspective n-point problem for pose estimation of a calibrated camera given n 3D-to-2D point correspondences. The main difference with our work is that this approach is based on explicit computation of 3D information; therefore it requires 3D models of heads. We argue that this is an unnecessary complication. Moreover, Flores et al.'s method is not fully automated and requires hand-annotated landmarks on the test image.

In contrast with all prior work, we propose to train and test in image space without the need for 3D head features or pose information. Furthermore, no calibration or knowledge of camera parameters is needed. Finally, thanks to the recent improvement of automatic facial landmark estimation [17–21], our method is fully automated; it uses automatically estimated landmarks instead of manual annotations.

5.3 Caltech Multi-Distance Portraits Dataset

We collected a novel dataset, the *Caltech Multi-Distance Portraits* (**CMDP**). This collection is made of high quality frontal portraits of 53 individuals against a blue background imaged from seven distances spanning the typical range of distances between photographer and subject: 60, 90, 120, 180, 240, 360, 480 cm, see Figure 5.1. For distances exceeding 5m, perspective projection approaches a parallel projection (the depth of a face is about 10cm), therefore no samples beyond 480 cm were needed. Participants were selected among both genders, different ages and a variety of ethnicities, physiognomies, hair and facial hair styles, to make the dataset as heterogeneous and representative as possible.

Table 5.1: **Subject variety in the CMDP dataset.** Diversity of the subjects contained in the *Caltech Multi-Distance Portraits* dataset. Individuals may belong to multiple categories.

Category	Num. of Subjects	Percentage
African-American	4	7.5%
Asian	5	9.4%
Caucasian	36	68.2%
Latino	8	15%
Female	7	13.2%
Male	46	86.8%
With Facial Hair	13	24.5%
With Occlusions	11	20.7%

Pictures were collected with a Canon Rebel Xti DSLR camera mounting a 28-300mm L-series Canon zoom lens. Participants standing in front of a blue background were instructed to remain still and maintain a neutral expression. The photographer used a monopod to support the camera-lens assembly. The monopod was adjusted so that the height of the center of the lens would correspond to the bridge of the nose, between the eyes. Markings on the ground indicated seven distances. After taking each picture, the photographer moved the foot of the monopod to the next marking,

adjusted the zoom to fill the frame of the picture with the face, and took the next picture. This procedure resulted in seven pictures (one per distance) being taken within 15-20 seconds. Images were then cropped and resampled to a common format. The lens was calibrated at different zoom settings to verify the amount of barrel distortion, which was found to be very small at all settings, and thus left uncorrected. Lens calibration was then discarded and not used further in our experiments.

As the camera approaches the subject, the relationship between the size of the picture and that of the main parts of the face changes, Figure 5.1. It is important to clarify that this 'perspective distortion' is not a lens error (this was verified, as explained in the previous paragraph): it arises from the projection of the three dimensional world into a two dimensional image and is easily observable with our own eyes. We could have used any other lens, including one with fixed focal length, or a pinhole camera, and there would have been no difference in the amount of 'perspective distortion' measured at a given distance (that is, assuming that the lens has no internal flaws or distortion). Using a lens with a shorter focal length and wider field of view will result in a coarser pixel sampling of the face, but the perspective geometry and proportions would only depend on distance, or, equivalently, on the visual angle subtended by the face. Regardless of the lens used, crops of two images taken from the same distance would be identical, apart from sampling resolution. We used a zoom lens to obtain maximum pixel resolution at all distances.

5.3.1 Annotating CMDP

All images in the dataset were manually annotated with 55 facial landmarks distributed over and along the face and head contour, see Figure 5.2-(a). The location of our landmarks is very different from landmark positions typically used in the literature, more focused towards the center and bottom of the face, as for example Multi-pie [22] format, Figure 5.2-(b). We purposely wanted to have landmarks around the head contour (in green) and all around the face (in red), to sample a larger area of the face.

The dataset was annotated by three different people. Portraits from the same subject were always labeled by the same annotator, to minimize the variance in the location of landmarks between pictures at different distances. To check consistency of annotations, we doubly annotated several images from different subjects. Annotators are very consistent, showing an average disagreement between them less than 3% of the interocular distance, and not varying much across distances, see Figure 5.4.

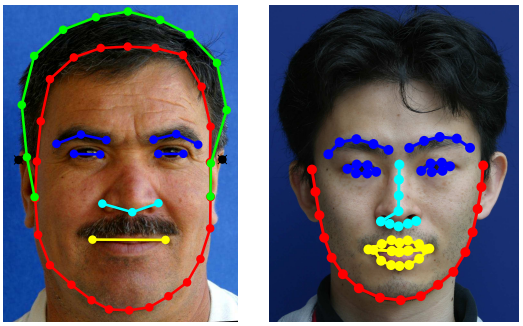


Figure 5.2: **Face landmark formats.** Our suggested 55 face landmarks (left) compared to the 68 ones contained in the Multi-pie format [22] (right). With landmarks around the hair line and top of the forehead, our landmarks allow to test whether these regions provide useful signal, despite their intrinsic variability.

5.4 Problem Formulation

The goal of this investigation is to estimate the camera-head distance from a single 2D portrait photograph when both the camera and the sitter are unknown. From this initial problem formulation, we derive two different tasks:

1. Sorting the seven images belonging to a single previously unseen subject according to their distance.
2. Estimating the distance from which a single image of a previously unseen subject was taken.

While the difference between the two might seem subtle, it affects the entire procedure. Firstly, from a machine learning point of view, the former is a classification task, while the latter is a pure regression problem, meaning that feature normalization schemes and error metrics will be different in each case.

Secondly, pure regression is a much harder task. Since the person has never been seen before, it is difficult to account for his/her physiognomy. For example, a person with a round face or a big nose will often appear closer than a squared face with a small nose, see Figure 5.3.

In fact, while humans are able to perform the first task rather accurately, see Figure 5.6, they are completely unable to perform the second task. Part of the reason is the well-known fact that humans are better at relative judgments, rather than estimating absolute values. Another reason may be that having access to several pictures of the same subject allows to ignore physiognomy and focus on the important signal. For real-life applications, the regression task is far more relevant; we use the ordering task exclusively to benchmark our method against human performance and guide our thoughts.

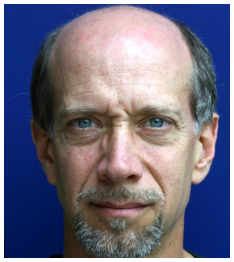



Figure 5.3: **Impact of physiognomy when estimating the distance of a person from a portrait.** Estimating the relative distance of previously unseen subjects is a difficult task. Consider these portraits: their physiognomy confuses human annotators, which have a tendency to pick the left image (d=240cm) as the closest one, while the right hand side one (d=180cm) was closer.

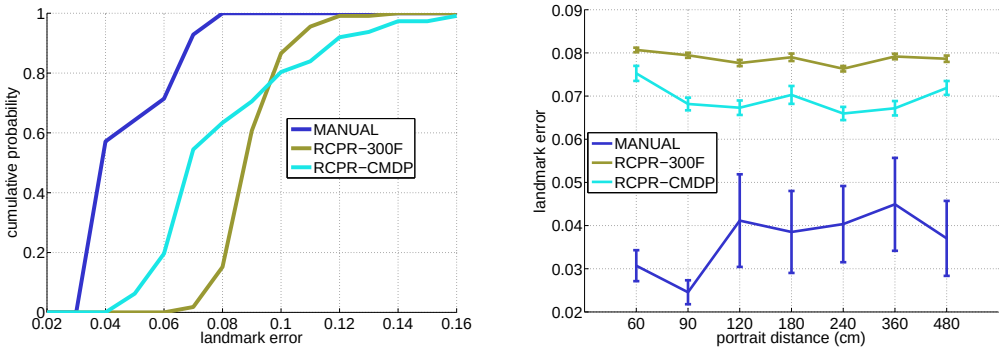
Error metrics: In the re-ordering problem, we measure for each portrait the probability of being correctly classified into its distance category (from 1 to 7). In the regression task, we measure both the Pearson correlation coefficient (Corr) and the coefficient of determination (R^2) between prediction and ground-truth distance on all 7 images of the test subject:

$$Corr(sbj) = \frac{COV(gt(sbj), pred(sbj))}{std(gt(sbj)) * std(pred(sbj))}, \quad R^2(sbj) = 1 - \frac{(gt(sbj) - pred(sbj))^2}{\left(gt(sbj) - \overline{gt}\right)^2}, \tag{5.1}$$

where the terms *gt(sbj)* and *pred(sbj)* are, respectively, the ground-truth and predicted distances of each picture belonging to the subject being evaluated, and $\overline{gt}$ is the average of all ground-truth distances.

5.5 Method

We use the position of the face's landmarks to capture the 2D shape of the face and therefore measure how much it changes with distance. Input landmarks can be both the result of manual annotations or the output of a landmark estimation algorithm. After computing the facial landmarks, we apply a supervised learning approach, see Section 5.5.2. A subset of the subjects in the dataset are used to train a regressor capable of mapping the shape of their face at different distances to the ground-truth distances. Then, the performance of the learned regressor is evaluated on the remaining subjects in the dataset according to each of the tasks defined in the previous Section.



(a) Cumulative probability of errors (b) Error as a function of face distance

Figure 5.4: **Performance of the face landmark estimation algorithm.** Human annotators are very consistent, showing a low disagreement (3% average), and not varying much across distances. Training RCPR using images from CMDP achieves good average performance except for its high number of failures (16%), struggling in close-range images. RCPR trained on 300-Faces yields slightly worse average performance with a lower number of failures (4%). A failure is an average error above 10%, as in [20].

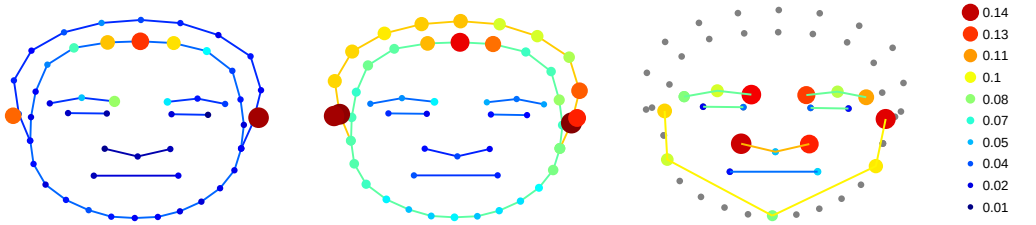
5.5.1 Facial Landmarks

Encouraged by the recent success of facial landmark estimation approaches, we decided to benchmark its feasibility for this task. We use *Random Cascaded Pose Regression (RCPR)* [20], due to its performance, speed and availability of code.

We trained RCPR on 70% of the individuals in our CMDP dataset (259 images in total), with the same parameters as in the original publication. When applied to the remaining 30% of our dataset, RCPR yields an average landmark error of 6.9% and a 16% failure rate, see Figure 5.4-(a). Errors are measured as the average landmark distance to ground-truth, and normalized as a percentage with respect to interocular distance. A failure is an average error above 10%, as in [20].

We also trained RCPR on the more exhaustive 300-Faces-in-the wild dataset [21] which contains more than 2K faces taken from previously existing datasets and re-annotated following Multi-Pie 68 landmarks [22] convention, see Figure 5.2-(b). To compare its results on our test images, we only evaluate it on the 22 set of landmarks our convention shares with Multi-Pie format. This version of RCPR, applied to the same 30% subset of subjects achieves an average error of 7.8%, but with a much lower failure rate (4%), see Figure 5.4-(a).

Both RCPR versions are still far from human performance, struggling slightly more with faces from both distance extremes, less common in face recognition datasets, see Figure 5.4-(b). The distribution of errors by landmarks, shown in Figure 5.5,



(a) Human errors (b) RCPR-CMDP errors (c) RCPR-300F errors

Figure 5.5: **Distribution of errors by individual landmark.** (blue=low average error, red=high average error). (a) Humans concentrate their disagreement on the forehead (telling where it ends is somewhat subjective) and ears (which can be occluded by hair and excessive distortion). (b) RCPR trained on our faces struggles with hair and face contours. (c) RCPR trained on 300W-Faces struggles the most with the eyebrows and chin. Gray points signify non-presence of the landmark due to use of the Multi-Pie convention.

reveals that RCPR trained on CMDP struggles particularly with the head contour and the ears due to their inherent variability, while RCPR trained on 300F struggles with the nose and eyebrows. However, both versions still have a low number of failure cases and therefore these issues affect only slightly the final performance of distance estimation when compared with using ground-truth landmarks.

5.5.2 Proposed Approach

After collecting the facial landmarks, we use them as input to learn a regressor that maps face shapes to their ground-truth distances. More specifically, shape $\mathcal{S}$ is represented as a series of P landmark locations $\mathcal{S} = [(x_p, y_p)|p \in 1..P \wedge x, y \in \mathbb{R}]$. For each subject $i \in 1..N$, we dispose of seven different shape vectors S_d^i associated with each one of the different distance images $d \in 1..7$. The goal is to learn a robust mapping from each one of the seven shapes to their respective distance: $f : \mathcal{S}_d^i \mapsto \mathbb{R}$.

Shape vector normalization: Due to the heterogeneity of face physiognomies contained in our dataset, a prior normalization step of the face shapes is crucial to learn a robust mapping. First, we standardize all portraits using the individual shape vectors $\mathcal{S}_d^i$, cropping the image around the face, and removing scale and rotation variations. Then, we propose two different normalization schemes for each one of the tasks defined previously.

In the re-ordering task, we can compute the average subject face shape across all seven distances ($\overline{\mathcal{S}}^i = \frac{1}{7}\sum_{d=1}^{7}\mathcal{S}_d^i$) and use it to normalize each shape, subtracting the mean from the landmark's position ($\mathcal{S}'^i_d = \mathcal{S}_d^i - \overline{\mathcal{S}}^i$). This filters out the variations in the shape of the face due to the physiognomy of the individual, leaving only the changes due to perspective distortion.

In the case of the regression task, at test time we only have access to one shape $\mathcal{S}_d^i$ at a time. During training, however, we can compute the average shape for each distance ($\overline{\mathcal{S}}_d = \frac{1}{N}\sum_{i=1}^{N}\mathcal{S}_d^i$). These average shapes can then be used to codify the current shape as the concatenation of the differences between $\mathcal{S}_d^i$ and each one of the d average faces $\overline{\mathcal{S}}_d$: ($\mathcal{S'}_d^i = \left\langle \mathcal{S}_d^i - \overline{\mathcal{S}}_{d=1}, .., \mathcal{S}_d^i - \overline{\mathcal{S}}_{d=7} \right\rangle$).

The effect of each one of these normalization schemes on performance is presented in Figure 5.9-(b), compared also to no normalization at all. Each step improves performance significantly. It is evident that being able to average out with respect to the subject's shape makes a big difference, even compared to our distance normalization scheme.

Inverse distance: In practice, inverse distance is preferred to avoid the saturation of the signal after a certain value of distance (i.e. the difference in the measured distortion becomes negligible with respect to the change in distance).

Learning algorithm: We train a multivariate linear regressor to learn the mapping from the normalized shapes $\mathcal{S'}_d^i$ onto the inverse distance of a face as a weighted linear combination of the P landmark locations: ($\sum_{p=1}^{P}\mathbf{w}_p\mathcal{S'}_d^i(x_p, y_p)$). We tried several other methods, but none improved results compared to using a simple linear regression. We suspect this may be due to the the relatively small number of training examples and a further investigation should be conducted in future work.

Regression vs. classification: For the classification task, we sort the values the regressor outputs for each of the 7 images belonging to the same subject and compare it against ground-truth distance ordering.

5.6 Results

We now discuss the results of our method on the re-ordering and pure regression tasks. We benchmark three variants of our method depending on the nature of the input landmarks: using 1) Ground-truth landmarks (MANUAL), 2) Landmarks from RCPR trained on our CMDP images (RCPR-CMDP) and 3) Landmarks from RCPR trained on 300-Faces in the wild (RCPR-300F).

All reported results are obtained using 70% of the subjects for training and the remaining 30% for testing (the same train/test set as that used to train RCPR-CMDP), except in Figure 5.9 where cross-validation runs are used. Variance is shown as standard errors. In Section 5.6.3, we show examples of how subject physiognomy affects performance.

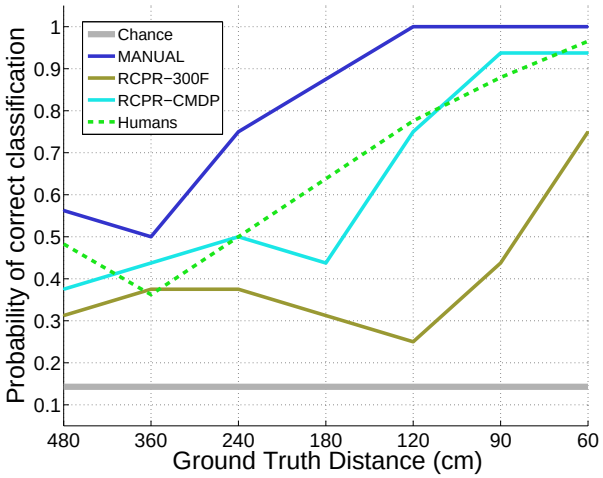


Figure 5.6: **Performance analysis on the re-ordering task.** We measured the probability of correctly ordering portraits of a subject according to their distance. Our methods using manual landmarks outperforms humans by 16%, while using RCPR-CMDP performance is virtually identical (lower by 3%).

5.6.1 Re-ordering Task

Figure 5.6 shows the performance on the re-ordering task. Apart from the three variants of our method, we also plot the result obtained by humans asked to perform the exact same task. We developed a specific GUI and asked a group of 5 people of different levels of computer vision expertise to sort a random permutation of all 7 pictures of a subject based on their conveyed distance. Each person annotated at least 10 different subjects (70 images in total).

The ground-truth landmarks based variant (MANUAL) outperforms human performance by 16%, while the automatic based ones (RCPR-CMDP and RCPR-300F) are slightly behind by 3% and 25% respectively. Closer faces appear to be much easier to classify than distant ones because of their unusual and disproportioned geometry. This has been confirmed by the human subjects of the study, stating their difficulty in telling apart images in the middle distance-range.

We find these results very encouraging. Our best variant outperforms human capabilities in the classification task, correctly reordering an average of 81% of the faces when random chance is merely 15%. The same method using machine estimated landmarks still classifies correctly 62% of the images, and could very likely be improved just by increasing the availability of training examples.

Figure 5.7-(Top) shows which landmarks are most discriminative for the re-ordering task using both MANUAL and RCPR-CMDP input. We measure how well each landmark group (head contour, face contour, eyes, nose, mouth) compares to best performance when only that particular group is used. For both MANUAL and RCPR-CMDP, best results are achieved using the head contour and the nose, while the eyes seem to be the least useful.

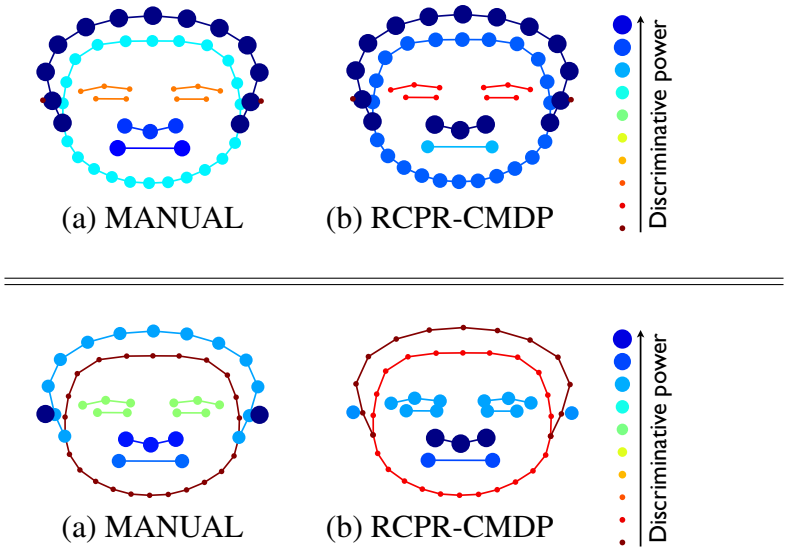


Figure 5.7: **Input landmarks' discriminative power on the re-ordering and regression tasks.** The discriminative power of each landmark - ranging from most discriminative (big blue dot) to least discriminative (small red dot) - is measured in terms of how well does our method perform when incorporating only a subset of all the possible landmarks as input to the learning. (Top) In the re-ordering task, the most useful landmarks are the face and head contours and the nose. (Bottom) In the regression task, the most useful groups of landmarks are the nose and ears. We observe that for both tasks there is significant overlap in the group of significative landmarks for both the MANUAL and RCPR-CMDP methods.

5.6.2 Regression Task

Figure 5.8 shows the results on the regression task. There is a strong correlation between ground-truth distances and predictions of our method. MANUAL achieves 75% correlation with a coefficient of determination of $R^2 = .5$, while RCPR-CMDP and RCPR-300F achieve 65% and 45% correlation, and $R^2 = .48$ and .46, respectively. All variants seem to struggle more with the larger distances, as noticeable from the higher variance and greater distance to the ground truth. This is an expected result considering the lower effect of perspective differences between two images taken from afar.

As expected, directly estimating the distance of an unknown face proved to be a harder task. Nonetheless, a correlation of 75% with ground-truth indicates that the method is learning well. Furthermore, increasing the amount of training data results in a peek of correlation up to 85%, see Figure 5.9-(a), with no apparent saturation, suggesting that with more training data available, final performance could be close to that desired for real-life applications. Figure 5.10 shows an example output of our algorithm for a subject whose predicted distances are close to the ground truth.

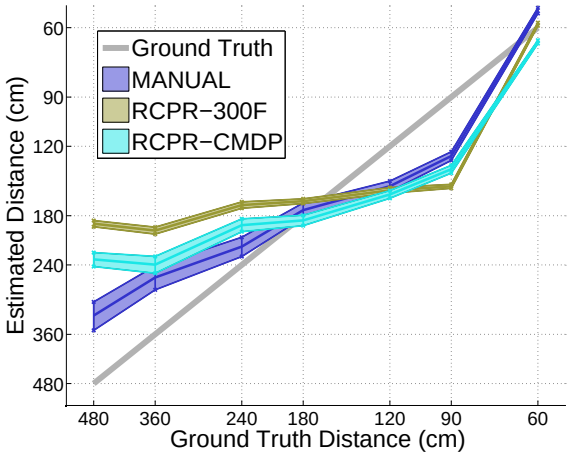
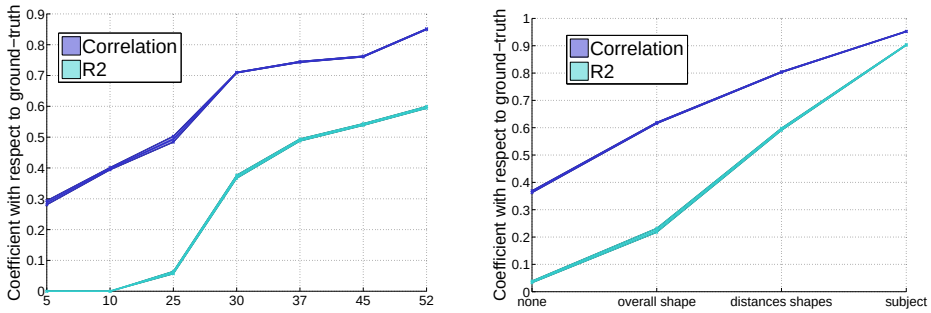


Figure 5.8: **Performance analysis on the regression task.** We measured the discrepancy between the estimated and ground-truth distance of a person from a single portrait. Using MANUAL landmarks achieves 75% correlation with a coefficient of determination of $R^2 = .5$, while RCPR-CMDP and RCPR-300F achieve 65% and 45% correlation, and $R^2 = .48$ and .46, respectively.



(a) Subjects used for training (b) Normalization scheme

Figure 5.9: **Ablation study for the regression task.** (a) The impact of increasing the number of training subjects on the regression task using MANUAL landmarks. With each added subject, the performance continues to grow with no saturation. (b) The impact of using the different normalization approaches presented in Section 5.5.2 with 52 training subjects in a leave-one-out cross validation scheme. Normalizing the shape of a subject's face using his own average shape across all distances achieves best performance.

Overall, our experiments suggest that the distance of a face may be estimated from an uncalibrated 2D portrait.

Figure 5.7-(Bottom) shows which landmarks are most discriminative for the regression task using both MANUAL and RCPR-CMDP input. We measure how well each landmark group (head contour, face contour, eyes, nose, mouth) compares to best performance when only that particular group is used. For both MANUAL and RCPR-CMDP, the most discriminative group is once again the nose. This finding agrees with human annotators, which consistently reported during the re-ordering psychophysics experiments the use of the deformation in a subject's nose as their main visual cue for the task.

Looking at Figure 5.7-(Top) and Figure 5.7-(Bottom) together is very informative.

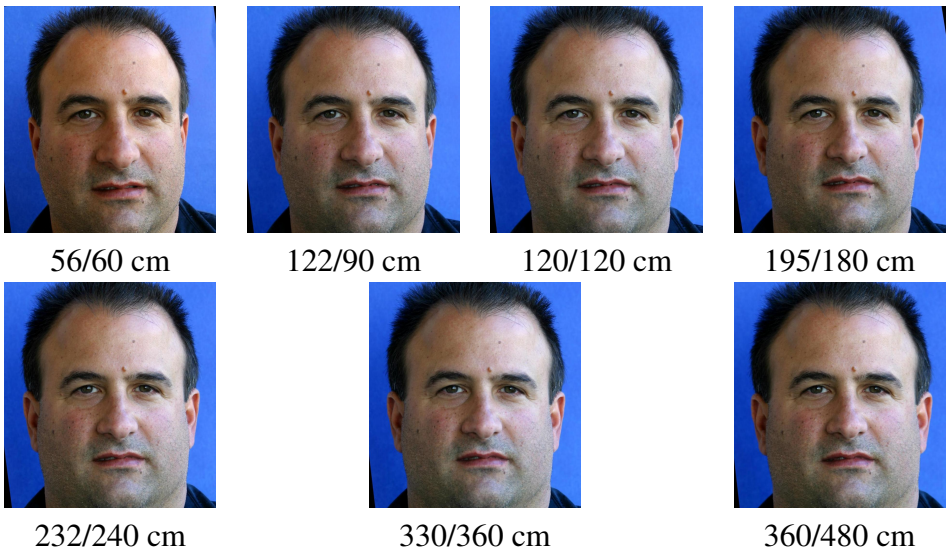


Figure 5.10: **Output of the regression algorithm for estimating the distance of an unknown person from a single portrait image.** The input images of the subject are ordered from top-left (closest) to bottom-right (furthest). The prediction of our method and the ground-truth distance are respectively listed below every picture.

They show that, as we suspected, head and facial contours are extremely important, which explains why RCPR using our landmark convention works far better that RCPR using Multi-Pie convention, which does not have landmarks around head contour. The differences between both figures tell us what parts of the face vary the most across individuals, defining most important cues for physiognomy. Take the facial contour for instance: it switches from most useful in re-ordering task to least useful in regression. This is natural; if one knows the shape of a subject's face (as in the re-ordering task), it can be very useful to watch how it gets deformed by perspective. However, when not being able to tell physiognomy and perspective changes apart (as in the regression task), information about the landmarks becomes uninformative.

5.6.3 Physiognomy Interpretation

A final interesting observation regards physiognomy. Throughout all of the experiments, we observed that physiognomy of people turned out to be one of the key factors for performance, both for human observers and for our algorithm.

In fact, some people appear to be systematically closer than others exclusively due to the shape of their face, Figure 5.3, and this systematically biases the estimate of their distance to the camera.

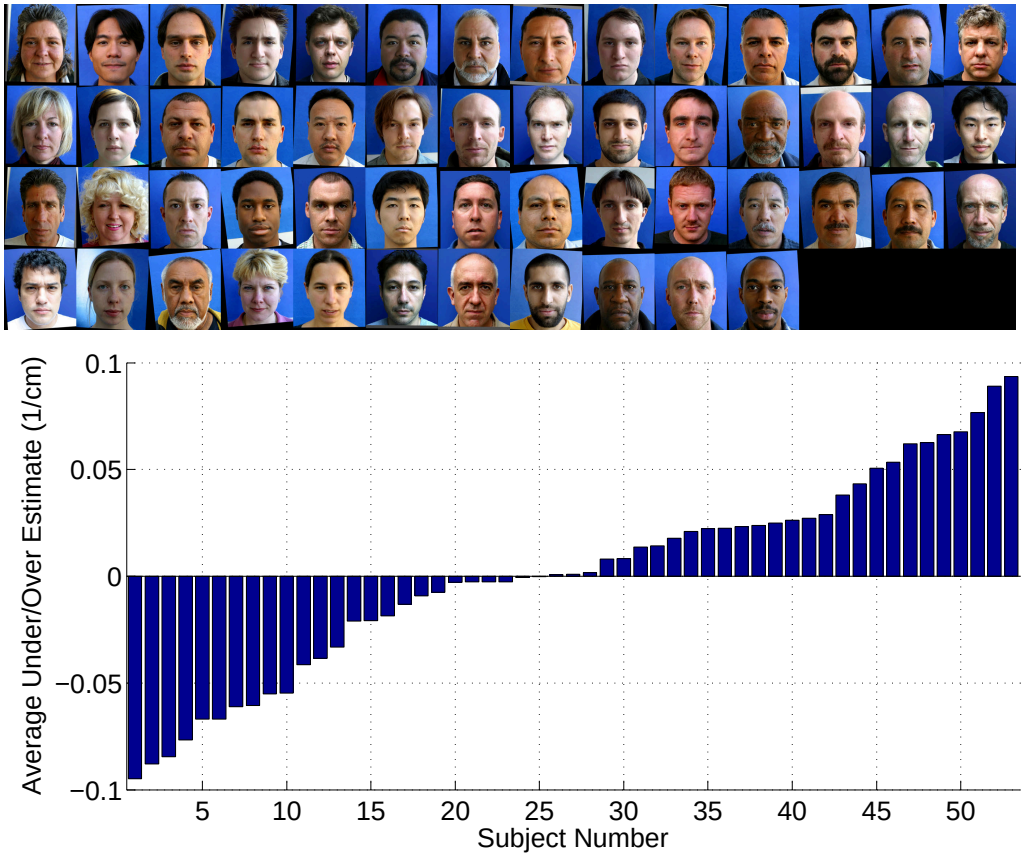


Figure 5.11: **Average per subject bias in the distance estimation task.** (Top) Pictures of all the subjects belonging to the dataset ordered by the amount of bias in their distance estimation - averaged over all the seven images of a subject - from top-left (most under-estimated distance) to bottom-right (most over-estimated distance). (Bottom) The value of the bias in the estimate of the distance of an image over all the distances for a certain subject.

Figure 5.12: **Visual example of how physiognomy biases distance estimation.** The ten most under-estimated (top) and over-estimated (bottom) subjects in terms of predicted distance to the camera.

We discussed in Section 5.5 normalization schemes discarding physiognomy and preserving the signal from perspective distortion, and, accordingly, we have found that the accuracy of the method increases when the normalization uses the subject's own average shape across the pictures at all distances, see Figure 5.9-(b).

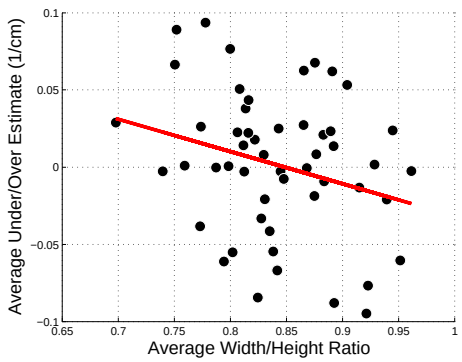


Figure 5.13: **Correlation between the physiognomy of a person and the bias in its distance estimation.** There is no apparent correlation between the width / height ratio of a subject's face and its overall bias in the distance estimation. Very squared (or round) faces are plotted on the right extreme of the x-axis, and elongated faces on the left. On the y-axis, faces are displayed from those whose distance is over-estimated on average (top) to those with under-estimated distance (bottom).

However this subject-specific normalization is only applicable in the re-ordering task, where we can legitimately assume the availability of information on the subject. This has no practical bearing in the regression task where the person being portrayed is unknown.

To answer the question of whether the results of our method could provide some information about a subject's physiognomy, we analyzed what specific attributes of a human face are most likely to bias its distance estimates. Therefore, we measured for all the faces in the dataset their average bias in the estimated distance over several runs with different training-test set combinations and show our findings over the whole dataset in Figure 5.11. The ten most under-estimated and over-estimated subjects are highlighted in Figure 5.12.

Besides a subjective feeling of roundness for the over-estimated faces (judged closer by the algorithm), no evident pattern was found so far. We also investigated the correlation between facial properties of the subjects, such as the width / height ratio and the bias in estimated distance, but again no evident pattern was found, as shown in Figure 5.13.

Further understanding the relationship and patterns between subject physiognomy and distance estimate bias could allow to cluster subjects into template categories of physiognomy based on their appearance, so that during the test of an unseen face, the normalization step can be done using only faces from subjects in the training set that belong to the same category.

Estimating the physiognomy of an unknown subject from a single picture is, thus, an open question answering which would have a great impact on the performance of our algorithm and should be one of the main lines of investigation in the future.

5.7 Conclusions

We proposed the first method for estimating automatically the distance from which a face was photographed. We assume that we have a single frontal photograph, where both the person and the camera are unknown. The method is based on two steps: first, estimating the position of a number of facial landmarks; second, estimating from their relative position the inverse distance by regression.

We find that the method is reasonably accurate. When using manually annotated landmarks as input, it outperforms relative depth judgments obtained from human observers. Furthermore, we find that performance does not suffer much when the method is fully automated with machine-based face landmark estimation. The fully automated method can estimate absolute distance, which human observers are unable to do. As expected, distance estimates beyond 3m, where perspective projection approaches parallel projection, are much noisier than distance estimates in the 0.5-2m range.

An interesting finding is that the main source of error is the variability of physiognomies. Some people appear to be systematically closer than others because their face is shaped differently. Once one normalizes for physiognomy, the accuracy of the method increases about 30%; this has no practical bearing when the person being portrayed is unknown, and therefore it is impossible to normalize for physiognomy.

Recovering the distance of a face has a number of applications: as an additional cue to depth in scene analysis, as an indicator of the possible emotional valence of the picture [1], as a tool to study portraiture in classical paintings, and as a tool for forensic analysis of images [3]. Our experiments are encouraging, and are sufficient as a proof of principle to demonstrate feasibility. However, they indicate that accuracy would be significantly better if a much larger training set was available. It is intuitive that such a dataset should include a representative range of facial expressions, as well as a range of viewpoints.

References

[1] R. Bryan, P. Perona, and R. Adolphs, "Perspective distortion from interpersonal distance is an implicit visual cue for social judgments of faces", *PloS one*, vol. 7, no. 9, e45301, 2012 (cit. on pp. 38, 40, 54).

[2] P. Perona, "A new perspective on portraiture", *Journal of Vision*, vol. 7, no. 9, pp. 992–992, 2007 (cit. on pp. 38, 40).

[3] H. Farid, "Image forgery detection", *Signal Processing Magazine, IEEE*, vol. 26, no. 2, pp. 16–25, 2009 (cit. on pp. 38, 54).

[4] C. Wheatstone, "Contributions to the physiology of vision. part the first. on some remarkable, and hitherto unobserved, phenomena of binocular vision", *Philosophical transactions of the Royal Society of London*, vol. 128, pp. 371–394, 1838 (cit. on p. 39).

[5] E. J. Gibson, J. J. Gibson, O. W. Smith, and H. Flock, "Motion parallax as a determinant of perceived depth.", *Journal of experimental psychology*, vol. 58, no. 1, p. 40, 1959 (cit. on p. 39).

[6] B. Rogers, M. Graham, *et al.*, "Motion parallax as an independent cue for depth perception", *Perception*, vol. 8, no. 2, pp. 125–134, 1979 (cit. on p. 39).

[7] D. Scharstein and R. Szeliski, "High-accuracy stereo depth maps using structured light", in *CVPR*, 2003 (cit. on p. 39).

[8] Microsoft, *Corp redmond wa. kinect for xbox 360.* (Cit. on p. 39).

[9] W. C. Gogel, "The effect of object familiarity on the perception of size and distance", *The Quarterly journal of experimental psychology*, vol. 21, no. 3, pp. 239–247, 1969 (cit. on p. 39).

[10] B. Triggs, "Camera pose and calibration from 4 or 5 known 3d points", in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, IEEE, vol. 1, 1999, pp. 278–284 (cit. on p. 39).

[11] E. Murphy-Chutorian and M. M. Trivedi, "Head pose estimation in computer vision: A survey", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 4, pp. 607–626, 2009 (cit. on p. 40).

[12] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Van Gool, "Random forests for real time 3d face analysis", *Int. J. Comput. Vision*, vol. 101, no. 3, pp. 437–458, Feb. 2013 (cit. on p. 40).

[13] C. H. Liu and A. Chaudhuri, "Face recognition with perspective transformation", *Vision Research*, vol. 43, no. 23, pp. 2393–2402, 2003 (cit. on p. 40).

[14] C. H. Liu and J. Ward, "Face recognition in pictures is affected by perspective transformation but not by the centre of projection", *Perception*, vol. 35, no. 12, p. 1637, 2006 (cit. on p. 40).

[15] A. Flores, E. Christiansen, D. Kriegman, and S. Belongie, "Camera distance from face images", in *International Symposium on Visual Computing (ISVC)*, Crete, Jul. 2013 (cit. on p. 40).

[16] V. Lepetit, F. Moreno-Noguer, and P. Fua, "Epnp: An accurate o (n) solution to the pnp problem", *International journal of computer vision*, vol. 81, no. 2, pp. 155–166, 2009 (cit. on p. 40).

[17] J. Saragih, S. Lucey, and J. F. Cohn, "Deformable model fitting by regularized landmark mean-shift", *IJCV*, vol. 2, no. 91, pp. 200–215, 2011 (cit. on p. 41).

[18] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localiz. in the wild", in *CVPR*, 2012 (cit. on p. 41).

[19] X. Cao, Y. Wei, F. Wen, and J. Sun, "Face alignment by explicit shape regression", in *CVPR*, 2012 (cit. on p. 41).

[20] X. Burgos-Artizzu, P. Perona, and P. Dollár, "Robust face landmark estimation under occlusion", in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1513–1520 (cit. on pp. 41, 45).

[21] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic., "300 faces in-the-wild challenge: The first facial landmark localization challenge", in *ICCV-Workshop*, 2013 (cit. on pp. 41, 45).

[22] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-pie", in *FG*, 2008 (cit. on pp. 42, 43, 45).

Chapter 6

BENCHMARKING AND ERROR DIAGNOSIS IN MULTI-INSTANCE POSE ESTIMATION

The contents of this chapter are adapted from the peer-reviewed publication:

M. R. Ronchi and P. Perona *"Benchmarking and Error Diagnosis in Multi-Instance Pose Estimation."* 16th International Conference on Computer Vision (2017, Venice, Italy).

DOI: 10.1109/ICCV.2017.48

URL: http://www.vision.caltech.edu/~mronchi/projects/PoseErrorDiagnosis

We propose a new method to analyze the impact of errors in algorithms for multi-instance pose estimation and a principled benchmark that can be used to compare them. We define and characterize three classes of errors: *i)* localization, *ii)* scoring, and *iii)* background, and study how they are influenced by instance attributes and their overall impact on an algorithm's performance. Our technique is applied to compare the two leading methods for human pose estimation on the COCO dataset, measure the sensitivity of pose estimation with respect to instance size, type and number of visible keypoints, clutter due to multiple instances, and the relative score of instances. The performance of algorithms, and the types of error they make, are highly dependent on all these variables, but mostly on the number of keypoints and the clutter. The analysis and software tools we propose offer a novel and insightful approach for understanding the behavior of pose estimation algorithms and an effective method for measuring their strengths and weaknesses.

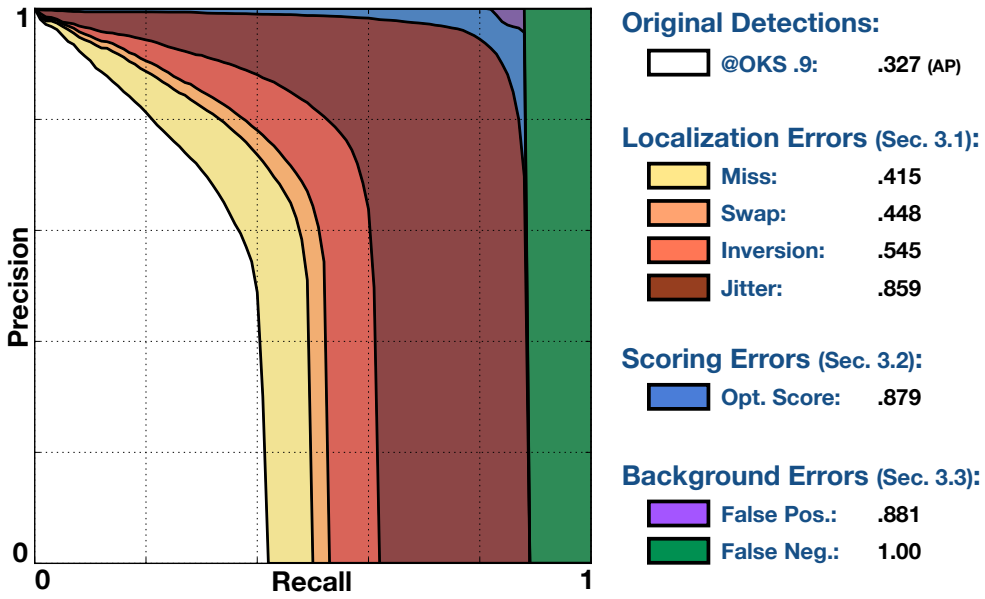


Figure 6.1: **Coarse to fine error analysis of multi-instance person keypoint estimation algorithms.** We study the errors occurring in multi-instance person keypoint estimation, and how they are affected by physical characteristics of the portrayed people. We build upon currently adopted evaluation metrics and provide the tools for a fine-grained description of performance, which allows to quantify the impact of different types of error at a single glance. The white Precision-Recall curve is obtained through the coarse error analysis of current evaluation algorithms. The colored Precision-Recall curves are obtained through our fine-grained error analysis and show the improvement in the performance of an algorithm after progressively correcting different types of mistakes.

6.1 Introduction

Estimating the pose of a person from a single monocular frame is a challenging task due to many confounding factors, such as perspective projection, the variability of lighting and clothing, self-occlusion, occlusion by objects, and the simultaneous presence of multiple interacting people. Nevertheless, the performance of human pose estimation algorithms has recently improved dramatically, thanks to the development of suitable deep architectures [1–10] and the availability of well-annotated image datasets, such as MPII Human Pose and COCO [11, 12]. There is broad consensus that performance is saturated on simpler single-person datasets [13, 14], and researchers' focus is shifting towards less constrained and more challenging datasets [11, 12, 15], where images may contain multiple instances of people, and a variable number of body parts (or keypoints) are visible.

However, evaluation is challenging: more complex datasets make it harder to benchmark algorithms due to the many sources of error that may affect performance,

and existing metrics, such as Average Precision (AP) or mean Percentage of Correct Parts (mPCP), hide the underlying causes of error and are not sufficient for truly understanding the behaviour of algorithms.

Our goal is to propose a principled method for analyzing the performance of algorithms for multi-instance human pose estimation. We make four contributions:

1. **Taxonomization** of the types of error that are typical of the multi-instance pose estimation framework;
2. **Sensitivity analysis** of these errors with respect to measures of image complexity;
3. Side-by-side **comparison of two leading human pose estimation algorithms** highlighting key differences in behaviour that are hidden in the average performance numbers;
4. Assessment of which types of datasets and **benchmarks** may be most productive in **guiding future research**.

Our analysis extends beyond humans, to any object category where the location of parts is estimated along with detections, and to situations where cluttered scenes may contain multiple object instances. This is common in fine-grained categorization [16], or animal behavior analysis [17, 18], where part alignment is often crucial.

6.2 Related Work

6.2.1 Error Diagnosis

Object Detection: Hoiem et al. [19] studied how a detailed error analysis is essential for the progress of recognition research, since standard benchmark metrics do not tell us *why* certain methods outperform others and *how* could they be improved. They determined that several modes of failure are due to different types of error, and highlighted the main confounding factors for object detection algorithms. While [19] pointed out the value of discriminating between different errors, it did not show how to do so in the context of pose estimation, which is one of our contributions.

Pose Estimation: In their early work on pose regression, Dollár et al. [20] observed that unlike human annotators, algorithms have a distribution of the normalized distances between a part detection and its ground truth that is typically bimodal,

highlighting the presence of multiple error modes. The *MPII Human Pose Dataset* [11] Single-Person benchmark enables the evaluation of the performance of algorithms along a multitude of dimensions, such as 45 pose priors, 15 viewpoints and 20 human activities. However, none of the currently adopted benchmarks for Multi-Person pose estimation [12, 15, 21] carry out an extensive error and performance analysis specific to this framework, and mostly rely on the metrics from the Single-Person case. No standards for performing or compactly summarizing detailed evaluations has yet been defined, and, as a result, only a coarse comparison of algorithms can be carried out.

6.2.2 Evaluation Framework

We conduct our study on COCO [12] for several reasons: *i)* it is the largest collection of multi-instance person keypoint annotations; *ii)* performance on it is far from saturated and conclusions on such a large and non-iconic dataset can generalize to easier datasets; *iii)* adopting their framework, with open source evaluation code, a multitude of datasets built on top of it, and annual competitions, will have the widest impact on the community. The framework involves simultaneous person detection and keypoint estimation, and the evaluation mimics the one used for object detection, based on Average Precision and Recall (AP, AR). Given an image, a distance measure is used to match algorithm detections, sorted by their confidence score, to ground-truth annotations. For bounding-boxes and segmentations, the distance of a detection and annotation pair is measured by their Intersection over Union. In the keypoint estimation task, a new metric called Object Keypoint Similarity (OKS) is defined. The OKS between a detection $\hat{\theta}^{(p)}$ and the annotation $\theta^{(p)}$ of a person p, Eq. 6.1, is the average over the labeled parts in the ground truth ($v_i = 1, 2$), of the *Keypoint Similarity* between corresponding keypoint pairs, Figure 6.2; unlabeled parts ($v_i = 0$) do not affect the OKS [22].

$$\begin{cases} ks(\hat{\theta}_i^{(p)}, \theta_i^{(p)}) & = e^{-\frac{||\hat{\theta}_i^{(p)} - \theta_i^{(p)}||_2^2}{2s^2k_i^2}} \\ OKS(\hat{\theta}^{(p)}, \theta^{(p)}) & = \frac{\sum_i ks(\hat{\theta}_i^{(p)}, \theta_i^{(p)})\delta(v_i > 0)}{\sum_i \delta(v_i > 0)} \end{cases} \tag{6.1}$$

The *ks* is computed by evaluating an un-normalized Gaussian function, centered on the ground-truth position of a keypoint, at the location of the detection to evaluate. The Gaussian's standard deviation k_i is specific to the keypoint type and is scaled by the area of the instance s, measured in pixels, so that the OKS is a perceptually meaningful and easy to interpret similarity measure. For each keypoint type, k_i

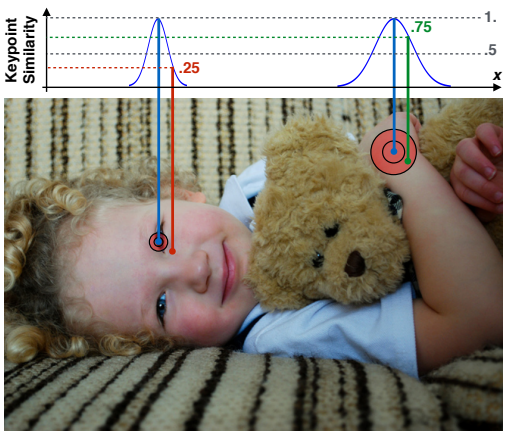


Figure 6.2: **Similarity measure between a keypoint detection and its ground-truth location.** The keypoint similarity (*ks*) between two detections, eye (red) and wrist (green), and their corresponding ground truth (blue). The red concentric circles represent *ks* values of .5 and .85 in the image plane and their size varies by keypoint type, see Section 6.2.2. As a result, detections at the same distance from the corresponding ground truth can have different *ks* values.

reflects the consistency of human observers clicking on keypoints of type i and is computed from a set of 5000 redundantly annotated images [22].

To evaluate an algorithm's performance, its detections within each image are ordered by confidence score and assigned to the ground-truth annotations that they have the highest OKS with. As matches are determined, the pool of available annotations for lower scored detections is reduced. Once all matches have been found, they are evaluated at a certain OKS threshold (ranging from .5 to .95 in [23]) and classified as True or False Positives (above or below threshold), and unmatched annotations are counted as False Negatives. Overall AP is computed as in the *PASCAL VOC Challenge* [24], by sorting the detections across all the images by confidence score and averaging precision over a predefined set of 101 recall values. AR is defined as the maximum recall given a fixed number of detections per image [25]. Finally, we will refer to cocoAP and cocoAR when AP and AR are additionally averaged over all OKS threshold values (.5:.05:.95), as done in the COCO framework [23].

6.2.3 Human Pose and Skeleton Color Coding

The COCO Keypoints format describes the 2D pose of a person portrayed in an image with a vector $\theta^{(p)} \in \mathcal{R}^{17\times3}$. There are four groups of keypoints annotated: *face*, *upper-body*, *torso*, and *lower-body*, for an overall total of 17 body parts, showed in Figure 6.3. Each individual body part is described using two 0-indexed coordinates (x, y) localizing that body part on the image plane, and a visibility flag v defined as follows: $v = 0$ if the part is not labeled (in which case $x = y = 0$), $v = 1$ if the part is labeled but occluded, and $v = 2$ if the part is labeled and visible. The (x, y, v) coordinates and flag values are used to compute the *ks* between corresponding keypoints in a detection and ground-truth annotation pair, as shown in Eq. 6.1.

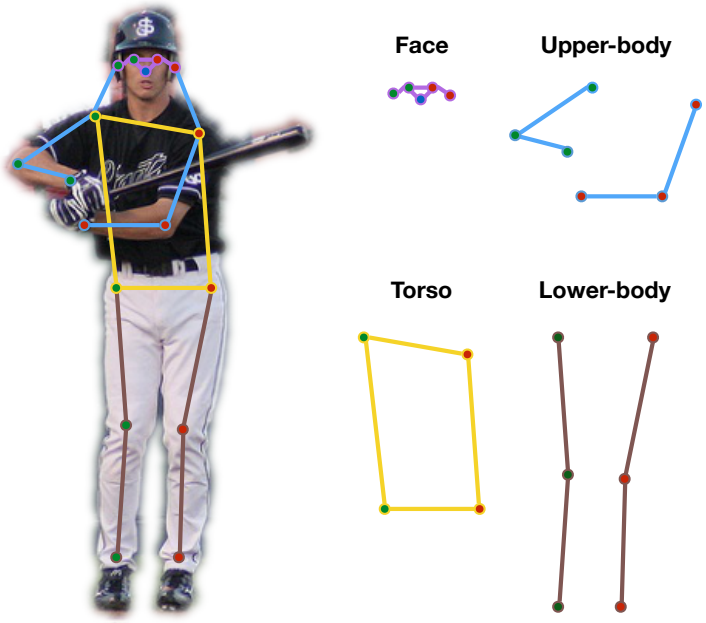


Figure 6.3: **COCO keypoints format for 2D human pose and skeleton color coding.**

We adopt the following color coding to visualize algorithm's keypoint detections:

- The location of the left and right parts of the body is indicated respectively with red and green dots; the location of the nose is plotted in blue.
- Face keypoints (*nose*, *eyes*, *ears*) are connected by purple lines.
- Upper-body keypoints (*shoulders*, *elbows*, *wrists*) are connected by blue lines.
- Torso keypoints (*shoulders*, *hips*) are connected by yellow lines.
- Lower-body keypoints (*hips*, *knees*, *ankles*) are connected by brown lines.

6.2.4 Algorithms

We conduct our analysis on the top-two ranked algorithms [2, 6] of the *2016 COCO Keypoints Challenge* [23], Table 6.1, and observe the impact on performance of the design differences between a top-down and a bottom-up approach.

Top-down (instance to parts) methods first detect humans contained in an image, then try to estimate their pose separately within each bounding box [4, 7, 10, 15]. The **Grmi** [6] algorithm is a two-step cascade. In the first stage, Faster-RCNN [26] based on a ResNet-Inception backbone [27] with inception layers [28] and residual connections [29] is used to produce a bounding box around each person instance.

	cocoAP	AP (50)	AP (75)	AP (M)	AP (L)	cocoAR	AR (50)	AR (75)	AR (M)	AR (L)
Cmu [31]	**0.608**	**0.834**	**0.664**	0.551	**0.681**	0.659	0.864	**0.713**	0.594	0.748
Grmi [6]	0.598	0.810	0.651	**0.567**	0.667	0.664	**0.865**	0.712	0.618	0.726
DL61	0.533	0.751	0.485	0.555	0.548	**0.708**	0.828	0.688	**0.740**	**0.782**
R4D	0.497	0.743	0.545	0.456	0.556	0.556	0.773	0.603	0.491	0.644
Umichvl	0.434	0.722	0.449	0.364	0.534	0.499	0.758	0.520	0.387	0.652

Table 6.1: **The 2016 COCO keypoints challenge leaderboard.** We show the leaderboard from [23] according to the average metrics used in the Keypoints Challenge, with the best result per each metric highlighted in bold. Even from such a simple analysis it is possible to observe how different algorithms perform better for different evaluation metrics.

The second stage serves as a refinement where a ResNet with 101 layers [29] is applied to the image crop extracted around each detected person instance in order to localize its keypoints. The authors adopt a combined classification and regression approach [26, 30]: for each spatial position, first a classification problem is solved to determine whether it is in the vicinity of each of the keypoints of the human body, followed by a regression problem to predict a local offset vector for a more precise estimate of the exact location. The results of both stages are aggregated to produce highly localized activation maps for each keypoint in the form of a voting process: each point in a detected bounding box casts a vote with its estimate for the position of every keypoint, and the vote is weighted by the probability that it lays near the corresponding keypoint.

Bottom-up (parts to instance) methods first separately detect all the parts of the human body from an image, then try to group them into individual instances [1, 5, 8, 9]. The **Cmu** [2] algorithm estimates the pose for all the people in an image by solving body part detection and part association jointly in one end-to-end trainable network, as opposed to previous approaches that train these two tasks separately [21, 32] (typically part detection is followed by graphical models for the association). Confidence maps with gaussian peaks in the predicted locations, are used to represent the position of individual body parts in an image. Part Affinity Fields (PAFs) are defined from the confidence maps, as a set of 2D vector fields that jointly encode the *location* and *orientation* of a particular limb at each position in the image. The authors designed a two-branch VGG [33] based architecture, inspired from CPMs [9], to iteratively refine confidence maps and PAFs with global spatial contexts. The final step consists of a maximum weight bipartite graph matching problem [34, 35] to associate body parts candidates and assemble them into full body poses for all the people in the image. A greedy association algorithm over a minimum spanning tree is used to group the predicted parts into consistent instance detections.

6.3 Multi-Instance Pose Estimation Errors

We propose a taxonomy of errors specific to the multi-instance pose estimation framework: *i)* **Localization**, due to the poor localization of the keypoint predictions belonging to a detected instance; *ii)* **Scoring**, due to a sub-optimal confidence score assignment; *iii)* **Background False Positives** (FP), detections without a ground-truth annotation match; *iv)* **False Negatives** (FN), missed detections. We assess the causes and impact on the behaviour and performance of [2, 6] for each error type.

6.3.1 Localization Errors

A localization error occurs when the location of the keypoints in a detection results in an OKS score with the corresponding ground-truth match that is lower than the evaluation threshold. These errors are typically due to the fact that body parts are difficult to detect because of self-occlusion or occlusion by other objects. We identify four types of localization errors, shown in Figure 6.4, using the keypoint similarity function ks, defined in Eq. 6.1 as a function of the keypoint i of a detection $\hat{\theta}_i^{(p)}$ and keypoint j of the ground-truth annotation $\theta_j^{(p)}$ for a person p.

Jitter: small error around the correct keypoint location:

$$.5 \leq ks(\hat{\theta}_i^{(p)}, \theta_i^{(p)}) < .85. \tag{6.2}$$

The limits can be chosen based on the application of interest; in the COCO framework, .5 is the smallest evaluation threshold, and .85 is the threshold above which also human annotators have a significant disagreement (around 30%) in estimating the correct position [22].

Inversion: confusion between semantically similar parts belonging to the same instance. The detection is in the proximity of the true keypoint location of the wrong body part:

$$\begin{aligned} & ks(\hat{\theta}_i^{(p)}, \theta_i^{(p)}) < .5, \text{ and} \\ \exists j \in \mathcal{J} \quad | \quad & ks(\hat{\theta}_i^{(p)}, \theta_j^{(p)}) \geq .5. \end{aligned} \tag{6.3}$$

In our study, we only consider inversions between the left and right parts of the body, however, the set of keypoints $\mathcal{J}$ can be arbitrarily defined to study any kind of inversion. In practice, an inversion error is the equivalent of a jitter error with the wrong body part.

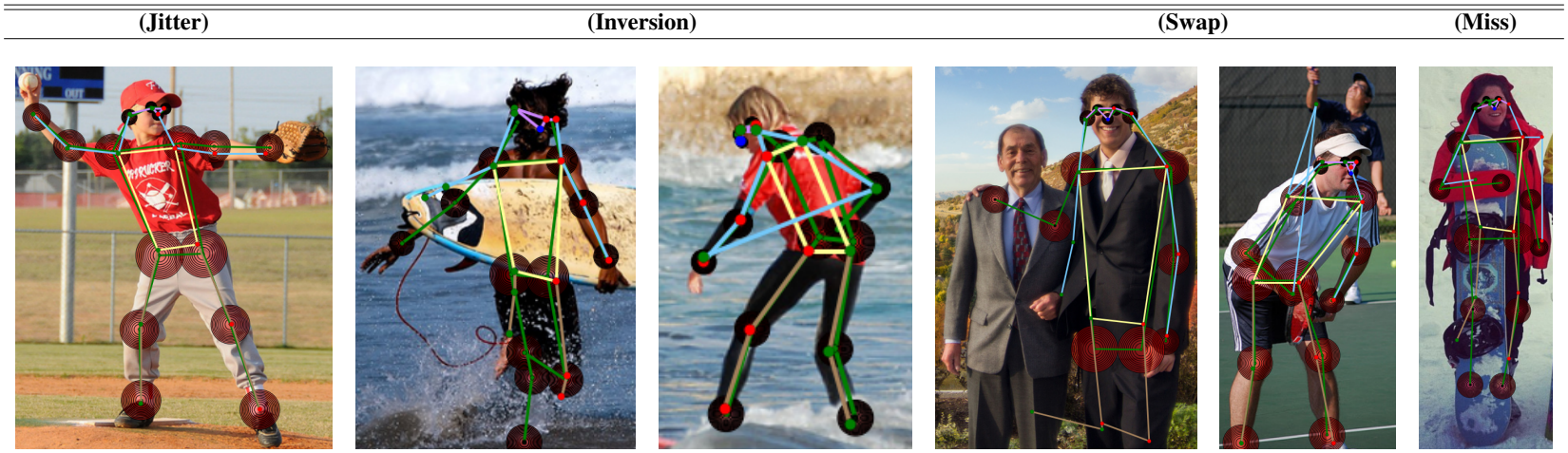


Figure 6.4: **Taxonomy of keypoint localization errors.** Keypoint localization errors, Section 6.3.1, are classified based on the position of a detection as, *Jitter*: in the proximity of the correct ground-truth location, but not within the human error margin - left hip in (a); *Inversion*: in the proximity of the ground-truth location of the wrong body part - inverted skeleton in (b), right wrist in (c); *Swap*: in the proximity of the ground-truth location of the body part of a wrong person - right wrist in (d), right elbow in (e); *Miss*: not in the proximity of any ground-truth location - both ankles in (f). While errors in (b,d) appear to be more excusable than those in (c,e) they have the same weight. Color-coding: (ground truth) - concentric red circles centered on each keypoint's location connected by a green skeleton; (prediction) - red/green dots for left/right body part predictions connected with colored skeleton, refer to Section 6.2.3 for an extended description.

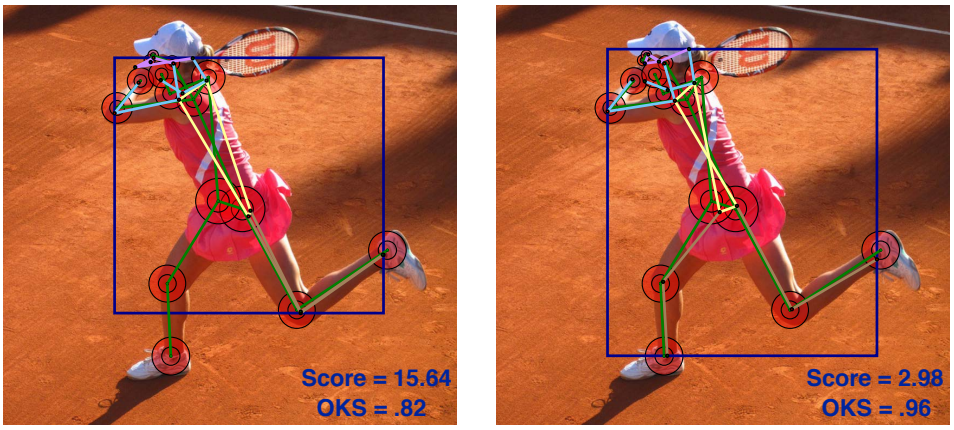


Figure 6.5: **Example of instance scoring error.** The detection with highest confidence score (Left) is associated to the closest ground-truth instance by the evaluation algorithm described in Section 6.2.2. However, its OKS is lower than the OKS of another detection (Right). This results in a loss in performance at high OKS thresholds, details in Section 6.3.2.

Swap: confusion between semantically similar parts of different instances. The detection is within the proximity of a body part belonging to a different person. In practice, a swap error is the equivalent of a jitter error with the correct body part of the wrong person:

$$\begin{aligned} & ks(\hat{\theta}_i^{(p)}, \theta_i^{(p)}) < .5, \text{ and} \\ \exists j \in \mathcal{J} \text{ and } \exists q \in \mathcal{P} \quad | \quad & ks(\hat{\theta}_i^{(p)}, \theta_j^{(q)}) \geq .5. \end{aligned} \tag{6.4}$$

Miss: large localization error, the detected keypoint is not within the proximity of any body part:

$$ks(\hat{\theta}_i^{(p)}, \theta_j^{(q)}) < .5 \quad \forall q \in \mathcal{P} \quad \text{and} \quad \forall j \in \mathcal{J}. \tag{6.5}$$

Every keypoint detection having a keypoint similarity with its ground-truth location that exceeds .85 is considered **Good**, as it is within the error margin of human annotators.

On average, different annotators are not able to provide annotations for a certain keypoint that agree consistently with each other at a value of $KS > .85$ [23], so we use it as the threshold value above which an algorithm should not be penalized.

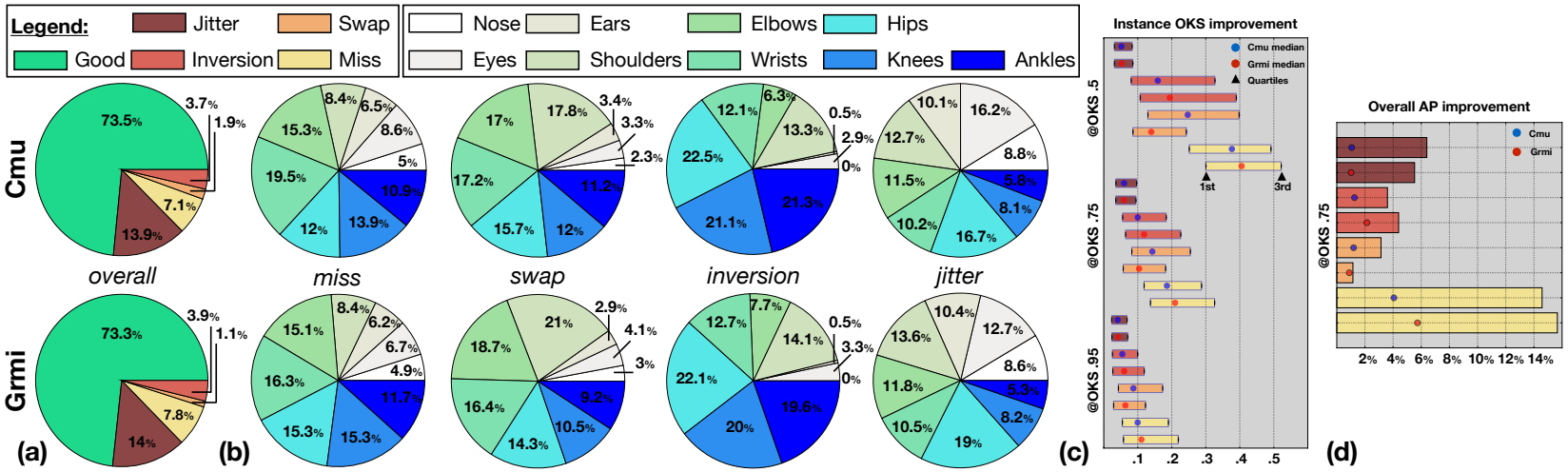


Figure 6.6: **Analysis of localization errors and their impact on performance.** Pie charts showing (a) the overall frequency of localization errors for all the keypoints predicted by [2, 6] and (b) the breakdown over all body parts for each type of localization error. (c) Each algorithm's OKS improvement obtained after separately correcting errors of each type (see Section 6.3.1 for details) evaluated over all the detected instances at OKS thresholds of .5, .75 and .95; the dots (blue for Cmu [2] and red for Grmi [6]) show the median, while the bars extend between the first and third quartile of the distribution. (d) The AP improvement obtained after correcting localization errors; evaluated at OKS thresholds of .75 (bars) and .5 (dots). A larger improvement shows what errors are more impactful on overall performance.

We can see in Figure 6.6-(a) that about 75% of both algorithms' detections are *good*, and while the percentage of *jitter* and *inversion* errors is approximately equal, [2] has twice as many *swaps*, and [6] has about 1% more *misses*.

Figure 6.6-(b) contains a breakdown of errors over keypoint type: faces are easily detectable (smallest percentage of *miss* errors); *swap* errors are focused on the upper-body, as interactions typically involve some amount of upper-body occlusion; the lower-body is prone to *inversions*, as people often self-occlude their legs, and there are less visual cues to distinguish left from right; finally, *jitter* errors are predominant on the hips. There are no major differences between the two algorithms in the above trends, indicating that none of the methods contain biases over keypoint type.

Correction of Localization Errors

After defining a novel taxonomy of localization errors, we measure the improvement in performance resulting from their correction.

To do so, for every keypoint predicted by an algorithm, we compute its Keypoint Similarity (KS) with *i)* the corresponding ground-truth body part, *ii)* different ground-truth body parts of the same person and *iii)* different ground-truth body parts of all the other people instances contained in the image. Once the above Keypoint Similarity values have been measured for all the individual keypoint predictions belonging to a detection, we use the definitions from Eq. 6.2 and Eq. 6.5 to identify which category of localization error they belong to, and individually correct them.

Given a detection $\hat{\theta}^{(p)}$ of a person in an image, and the corresponding ground-truth annotation $\theta^{(p)}$ with which it was matched by the evaluation procedure, we adjust the position of every body part keypoint j to obtain an optimal detection $\overset{*(p)}{\theta}$ using the definition of KS from Eq. 6.1:

- **Miss** errors (Eq. 6.5) such as *left-elbow* and *wrists* in Figure 6.7-(Top) are corrected by repositioning the keypoint prediction on the $KS = .5$ circle centered on the true location[1]:

$$\begin{cases} \overset{*(p)}{\theta}_j = \theta_j^{(p)} + v \\ v = (v_x, v_y) \quad | \quad ||v||_2^2 = -(2s^2k_j^2) \cdot \ln(.5) \end{cases} . \tag{6.6}$$

- **Swap** errors (Eq. 6.4) such as the *right-elbow* in Figure 6.7-(Top) are corrected by repositioning the keypoint prediction $\hat{\theta}_j^{(p)}$ at a distance from the correct

[1]The value of $KS = .5$ is the boundary between a miss and a jitter error.

ground-truth location $\theta_j^{(p)}$ so that the new value of KS is the same that the prediction had with the body part $\theta_j^{(q)}$ which was mistakenly detected since it belongs to a different person q:

$$\begin{cases} \overset{*(p)}{\theta_j} = \theta_j^{(p)} + v \\ v = (v_x, v_y) \quad | \quad ||v||_2^2 = -(2s^2k_j^2) \cdot \ln(KS) \\ KS = ks(\hat{\theta}_j^{(p)}, \theta_j^{(q)}) \end{cases} . \tag{6.7}$$

- **Inversion** errors (Eq. 6.3) such as the *right-knee* in Figure 6.7-(Top) are corrected by repositioning the keypoint prediction $\hat{\theta}_j^{(p)}$ at a distance from the correct ground-truth location $\theta_j^{(p)}$ so that the new value of KS is the same that the prediction had with the wrong body part $\theta_i^{(p)}$ it mistakenly detected from the same person:

$$\begin{cases} \overset{*(p)}{\theta_j} = \theta_j^{(p)} + v \\ v = (v_x, v_y) \quad | \quad ||v||_2^2 = -(2s^2k_j^2) \cdot \ln(KS) \\ KS = ks(\hat{\theta}_j^{(p)}, \theta_i^{(p)}) \end{cases} . \tag{6.8}$$

- **Jitter** errors (Eq. 6.2) such as the *left-ankle* in Figure 6.7-(Top) are corrected by repositioning the keypoint prediction on the $KS = .5$ circle centered on the true location[2]:

$$\begin{cases} \overset{*(p)}{\theta_j} = \theta_j^{(p)} + v \\ v = (v_x, v_y) \quad | \quad ||v||_2^2 = -(2s^2k_j^2) \cdot \ln(.85) \end{cases} . \tag{6.9}$$

- **Good** keypoints such as both *hips* and *shoulders* in Figure 6.7-(Top) have $ks(\hat{\theta}_j^{(p)}, \theta_j^{(p)}) > .85$ and their position does not need to be corrected[2]:

$$\overset{*(p)}{\theta_j} = \hat{\theta}_j^{(p)}. \tag{6.10}$$

In practice, *miss* and *jitter* errors are corrected by bringing a prediction to a fixed distance from its true position, while the new location of keypoints with *swap* and *inversion* errors depends on how good the prediction of the wrong body part was: after correction, a good/bad prediction of the wrong body part becomes a good/bad prediction (high/low KS) of the true body part.

[2]The value of $KS = .85$ is the boundary between a jitter error and a good prediction.

Figure 6.7: **Example of the progressive correction of keypoint localization errors.** The change of a detection's keypoint positions and the resulting OKS improvement as localization errors are progressively corrected. We plot the ground-truth skeleton in green and the detection using the color coding described in Section 6.2.3. The red concentric circles indicate the .5 and .85 KS threshold as discussed in Figure 6.2. When visualizing the individual error types, we show the concentric circles around the ground-truth location only for the keypoints that are being corrected.

After the above corrections have been applied to all the predicted keypoints, we obtain a new detection $\overset{*(p)}{\theta}$ which has a higher OKS with the corresponding ground truth $\theta^{(p)}$. The OKS increase depends on the number of errors, order in which they were corrected[3], and on the total number of visible keypoints present in a ground-truth instance.

Figure 6.7 provides two visual examples of how the keypoints belonging to a detection can be progressivily improved: the original algorithm's detection in Figure 6.7-(Top) had an OKS with its corresponding ground truth of .736, which would have resulted in a FP for all the OKS evaluation thresholds $\geq .75$. After correction, the OKS is now .945, corresponding to a TP the four thresholds at .75, .8, .85 and .9.

[3]The order (i.e. *miss* -> *swap* -> *inversion* -> *jitter*) in which localization errors are progressively corrected matters: different orders result in different corrected detections and improved OKS values.

Figure 6.6-(c) shows the quartiles of the distribution of the OKS improvement obtainable after correcting errors of each type at three OKS evaluation thresholds (measured for all the detections of [2, 6]). We see that *miss* errors are the ones that impact the most the overall OKS of a detection: the median improvement of fixing *miss* errors at OKS of .5 is $\sim$ .4 corresponding to an 80% improvement of its OKS. On the contrary, *jitter* errors, although occurring most frequently (Figure 6.6-(a)) have a small impact on the OKS. *Inversions* and *swaps*, even though less impactful than *miss* errors overall, have the largest difference between the first and third quartiles showing that their impact on OKS is very variable.

Changing the evaluation threshold changes the impact of errors (for instance by lowering it to .5, more detections are TP so there is less improvement from their correction), but the same relative trends are verified, indicating that the above observations reflect the behavior of the methods and are not determined by the strictness of evaluation. However, while the impact of *jitter* errors is mostly constant at the three evaluation thresholds (.5, .75, .95), it is greatly reduced for *miss*, *swap* and *inversion* errors when going to higher OKS thresholds. This indicates that for a detection to be a True Positive at the higher OKS thresholds, it cannot present multiple errors other than *jitter*.

Finally, in Figure 6.6-(d) we look at the improvement of performance in terms of AUC of the PR curves computed over the entire test set. The fine-grained Precision-Recall curves, as shown in Figure 6.1, are computed by fixing an OKS threshold and evaluating the performance after all of the algorithms' detections have been corrected. We learn that *misses* are the most costly error in terms of overall AP over the entire dataset ($\sim$ 15%), followed by *jitter* ($\sim$ 6%), *inversions* ($\sim$ 4%) and *swaps* ($\sim$ 2%).

An interesting difference between the two algorithms can be observed by looking at the improvements for *swap* and *inversion* errors: Grmi [6] suffers more of *inversions* and Cmu [2] of *swaps*. This seems plausible with the fact that a top-down method can only mistakenly assign keypoints of people whose detected bounding boxes overlap (as opposed to the whole image), and that a bottom-up method might more easily disambiguate left and right since it can take advantage of visual cues outside of the bounding box. Similarly, the higher *miss* rate of Grmi [6] can be due to the fact that it is restricted to finding keypoints inside the bounding box, so if the person detector does not provide a good bounding box, errors will always result in a miss.

6.3.2 Scoring Errors

Assigning scores to instances is a typical task in object detection, but a novel challenge for keypoint estimation. A scoring error occurs when two detections, $\hat{\theta}_1^{(p)}$ and $\hat{\theta}_2^{(p)}$, are in the proximity of a ground-truth annotation $\theta^{(p)}$, and the one with the highest confidence has the lowest OKS:

$$\begin{cases} Score(\hat{\theta}_1^{(p)}) & > Score(\hat{\theta}_2^{(p)}) \\ OKS(\hat{\theta}_1^{(p)}, \theta^{(p)}) & < OKS(\hat{\theta}_2^{(p)}, \theta^{(p)}) \end{cases}. \tag{6.11}$$

This can happen in cluttered scenes when many people and their detections are overlapping, or in the case of an isolated person for which multiple detections are fired, Figure 6.5. Confidence scores affect the evaluation procedure, Section 6.2.2, *locally* by determining the order in which detections get matched to the annotations within every image, and *globally*, when detections are sorted across the whole dataset to compute AP and AR. As a result, it is important for the detection scores to be: *i) 'OKS monotonic increasing'*, so that a higher score always results in a higher OKS; and *ii) calibrated*, so that scores reflect as much as possible the probability of being a true positive (TP).

We call a score possessing such properties *optimal*, as it will achieve the highest performance possible for the provided detections. It follows that if we assign the maximum OKS value obtainable with any ground-truth annotation from the same image as the score for a given detection, we have an optimal score: monotonicity and perfect calibration are both guaranteed by construction, as higher OKS detections would have higher scores, and the OKS is the exact predictor of the quality of a detection, by its own definition.

Correction of Scoring Errors

The optimal scores can be computed at evaluation time, by an oracle assigning to each detection a confidence corresponding to the maximum OKS score achievable with any ground-truth instance. To obtain them, we solve an assignment problem [34] over the OKS matrix between all detections and ground-truth annotations in an image, using the Hungarian algorithm [35], which maximizes the sum of the OKS value obtained over all matches.

Using optimal scores in place of the original ones from [2, 6] yields about 5% AP improvement, averaged over all OKS thresholds, and up to 10% at OKS .95, Figure 6.8-(a), pointing to the importance of assigning low scores to unmatched detections.

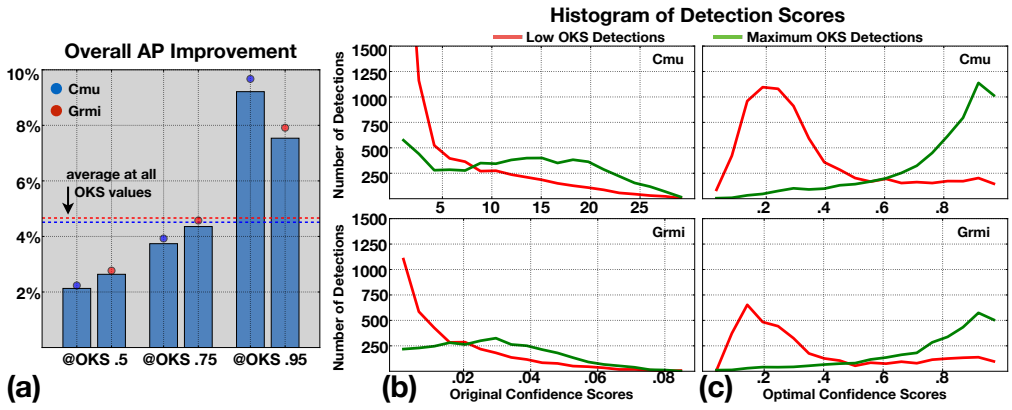


Figure 6.8: **Analysis of scoring errors and their impact on performance.** (a) The AP improvement obtained when using the optimal detection scores, as defined in Section 6.3.2. The histogram of detections' (b) original and (c) optimal confidence scores. We histogram separately the scores of detections achieving the maximum OKS with a given ground-truth instance (green) and the other detections achieving OKS of at least .1 (red). High overlap of the histograms, as in (b), is caused by the presence of many detections with high OKS and low score or vice versa; a large separation, as in (c), is indication of a better score.

	Cmu [2]	**Grmi** [6]
Imgs. w. detections	11940	14634
Imgs. w. optimal detection order	7456 (62.4%)	9934 (67.8%)
Number of Scoring Errors	407	82
Increase of Matches (using opt. score)	64	156
Matches with OKS Improvement (using opt. score)	590	430

Table 6.2: **Performance improvement due to the optimal rescoring of detections.**

A careful examination shows that the reason of the improvement is two-fold, Table 6.2: *i)* there is an increase in the number of matches between detections and ground-truth instances (reduction of *FP* and *FN*), and *ii)* the existing matches obtain a higher OKS value. Both methods have a significant amount of overlap, Figure 6.8-(b), between the histogram of *original* scores for the detections with the highest OKS with a given ground truth (green line) and all other detections with a lower OKS (red line). This indicates the presence of many detections with high OKS and low score, or vice versa. Figure 6.8-(c) shows the effect of rescoring: *optimal score* distributions are bi-modal and present a large separation, so confidence score is a better OKS predictor. Although the AP improvement after rescoring is equivalent, [6] provides scores that are in the same order as the optimal ones for a higher percentage of images and makes less errors, indicating that it is using a better scoring function.

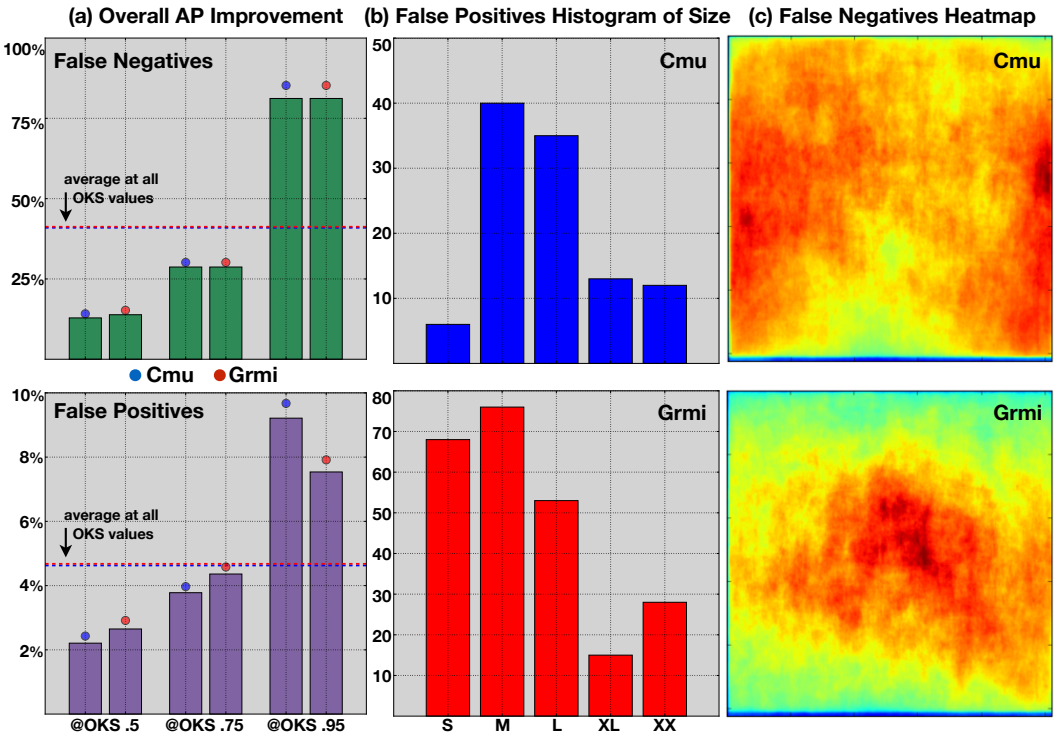


Figure 6.9: **Analysis of background errors (FP, FN) and their impact on performance.** (a) The overall AP improvement obtained after *FN* (top) and *FP* (bottom) errors are removed from evaluation; horizontal lines show the average value for each method. (b) The histogram of the area size of *FP* having a high confidence score. (c) The heatmaps obtained by adding the resized ground-truth COCO segmentation masks of all the *FN*.

6.3.3 Background False Positives and False Negatives

False Positives (*FP*) and False Negatives (*FN*), respectively, consist of an algorithm's detections and the ground-truth annotations that remain unmatched after evaluation is performed. *FP* typically occur when objects resemble human features or when body parts of nearby people are merged into a wrong detection. Most of the *FP* errors could be resolved by performing better Non-Max-Suppression [36] and scoring, since their impact is greatly reduced when using optimal scores, i.e. Figure 6.1. Small size and low number of visible keypoints are instead the main cause of *FN*. In Figure 6.9-(a), we show the impact of background errors on the AP at three OKS evaluation thresholds: FN affect performance significantly more than *FP*, on average about 40% versus only 5%. For both methods, the average number of people in images containing *FP* and *FN* is about 5 and 7, compared to the dataset's average of 3, suggesting that cluttered scenes are more prone to having background errors. Interestingly, the location of *FN* errors for the two methods differs greatly, Figure 6.9-

Figure 6.10: **Sample images from the proposed benchmarks of the COCO dataset.** We suggest to separate the ground-truth instances in the COCO dataset into twelve benchmarks, based on the number of visible keypoints and overlap between annotations; Figure 6.12-(b) shows the number of images in each benchmark.

(c): [2] predominantly misses annotations around the image border, while [6] misses those at the center of an image. Another significant difference is in the quantity of *FP* detections having a high confidence score (in the top-20th percentile of overall scores), Figure 6.9-(b): [6] has more than twice the number, mostly all with small pixel area size ($< 32^2$).

6.4 Sensitivity to Occlusion, Crowding, and Size

One of the goals of this study is to understand how the layout of people portrayed in images, such as the number of visible keypoints (*occlusion*), and the amount of overlap between instances (*crowding*) and size affects the errors and performance of algorithms. This Section is focused on the properties of the data, so we perform the analysis only on the detections from [2].

The COCO dataset contains mostly visible instances having little overlap: Figure 6.12 shows that only 1.7% of the annotations have more than two overlaps with an IoU $\geq .1$, and 86.6% have 5 or more visible keypoints. Consequently, we divide the dataset into twelve benchmarks, Figure 6.10, and study the performance and occurrence of errors in each separate one.

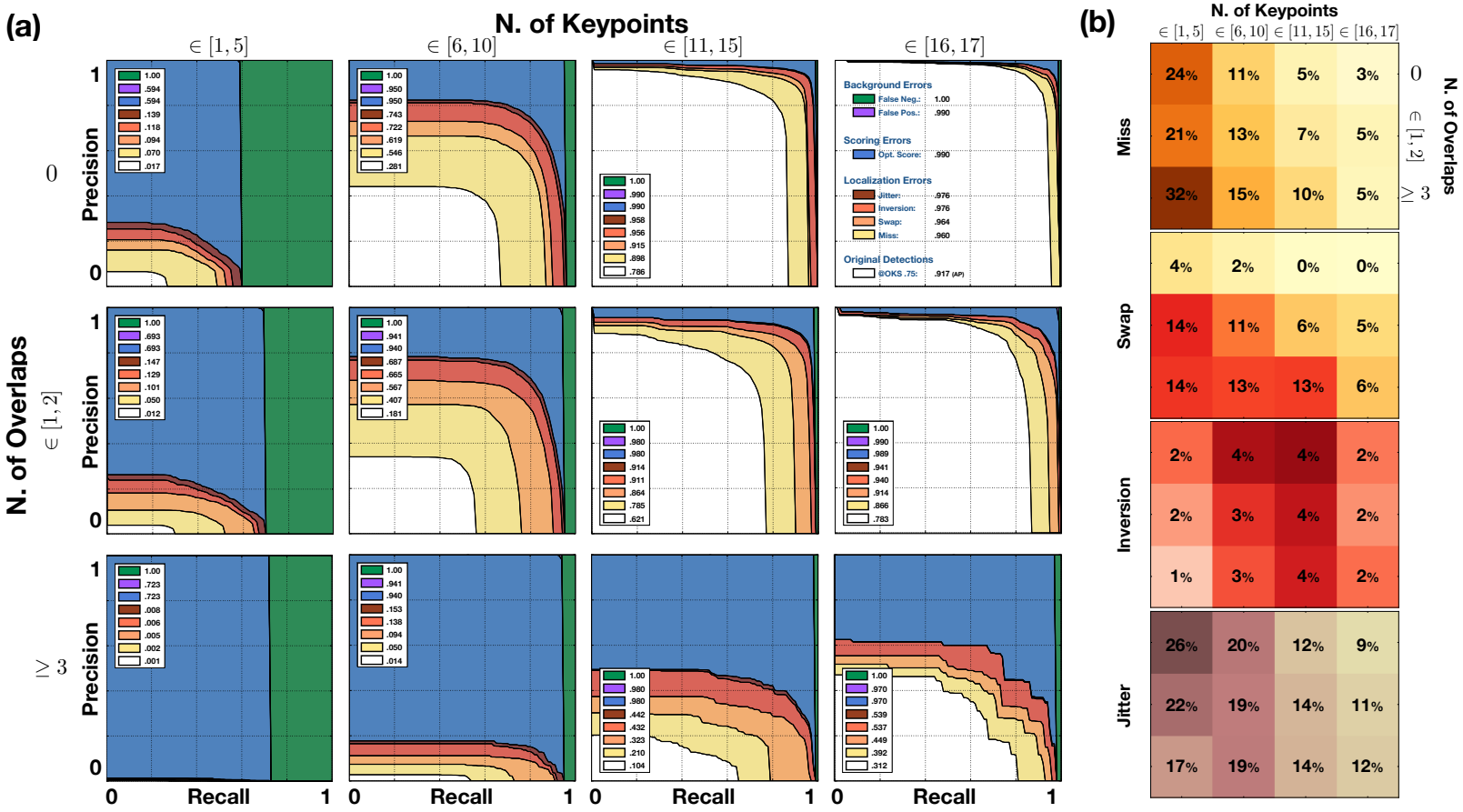


Figure 6.11: **Performance and error sensitivity to occlusion and crowding.** (a) The PR curves showing the performance of [2] obtained by progressively correcting errors of each type at the OKS evaluation threshold of .75 on the twelve *Occlusion and Crowding Benchmarks* described in Section 6.4; every legend contains the overall AP values. (b) The frequency of localization errors occurring in each benchmark set.

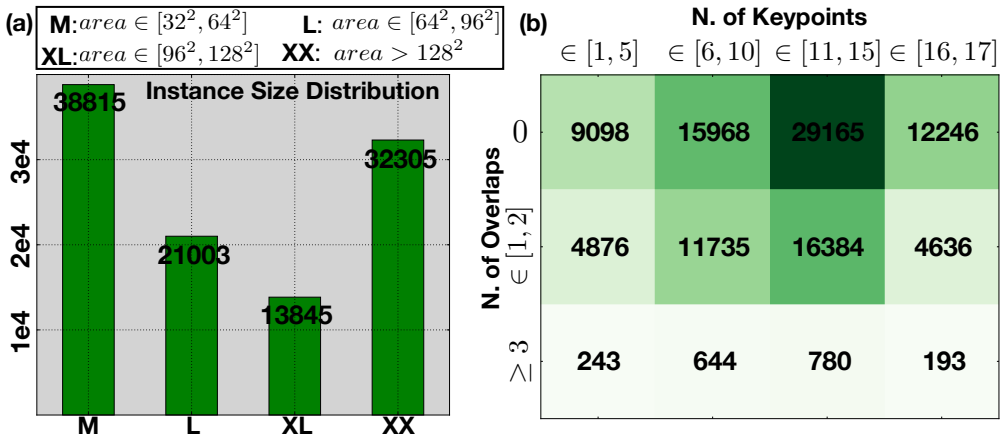


Figure 6.12: **Amount of training data in each of the suggested benchmarks for the COCO dataset.** The number of instances in each benchmark of the COCO training set based on (a) the size of instances, or (b) the number of overlapping ground-truth annotations with IoU $\geq .1$ and visible keypoints, Figure 6.10.

The PR curves obtained at the evaluation threshold of .75 OKS, after sequentially correcting errors of each type are shown in Figure 6.11-(a). It appears that the performance of methods listed in Table 6.1 is a result of the unbalanced data distribution, and that current algorithms still vastly underperform humans in detecting people and computing their pose, specifically when less than 10 keypoints are visible and overlap is present. Localization errors degrade the performance across all benchmarks, but their impact alone does not explain the shortcomings of current methods. Over 30% of the annotations are missed when the number of visible keypoints is less than 5 (regardless of overlap), and background *FP* and *scoring* errors account for more than 40% of the loss in precision in the benchmarks with high overlap. In Figure 6.11-(b), we illustrate the frequency of each localization error. *Miss* and *jitter* errors are predominant when there are few keypoints visible, respectively with high and low overlap. *Inversions* are mostly uncorrelated with the amount of overlap, and occur almost always in mostly visible instances. Conversly, *swap* errors depend strongly on the amount of overlap, regardless of the number of visible keypoints. Compared to the overall rates in Figure 6.6-(a-cmu), we can see that *inversion* and *jitter* errors are less sensitive to instance overlap and number of keypoints.

A similar analysis can be done by separating COCO into four size groups: *medium*, *large*, *extra-large* and *extra-extra large*, Figure 6.12-(a). The performance at all OKS evaluation thresholds improves with size, but degrades when instances occupy such a large part of the image that spatial context is lost, Figure 6.13-(a). AP is affected by size significantly less than by the amount of overlap and number of

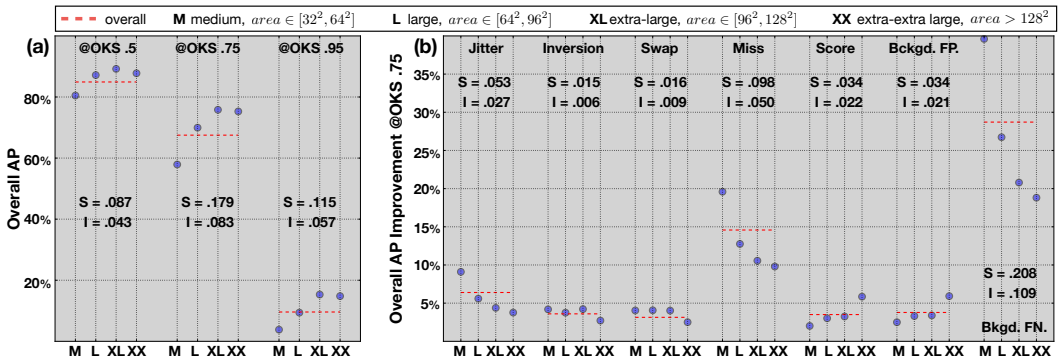


Figure 6.13: **Performance and error sensitivity to size.** (a) The overall AP obtained by evaluating [2] at three OKS evaluation thresholds on the four size benchmarks described in Section 6.4. (b) The AP improvement at the OKS threshold of .75 obtained after separately correcting each error type on the benchmarks. The dashed red line indicates evaluation over all the instance sizes, Sensitivity (S) and Impact (I) are respectively computed as the difference between the maximum and minimum, and the maximum and average, values.

visible keypoints. In Figure 6.13-(b), we show the AP improvement obtainable by separately correcting each error type in all benchmarks. Errors impact performance less (they occur less often) on larger instances, except for *scoring* and *FP*. Finally, while *FN*, *miss*, and *jitter* errors are concentrated on medium instances, all other errors are mostly insensitive to size.

6.5 Discussion and Recommendations

Multi-instance pose estimation is a challenging visual task where diverse errors have complex causes. Our analysis defines three types of error - *localization*, *scoring*, *background* - and aims to discover and measure their causes, rather than averaging them into a single performance metric. Furthermore, we explore how well a given dataset may be used to probe methods performance through its statistics of instances visibility, crowding and size.

The biggest problem for pose estimation is localization errors, present in about 25% of the predicted keypoints in state-of-the-art methods, Figure 6.6-(a). We identify four distinct causes of localization errors, *Miss*, *Swap*, *Inversion*, and *Jitter*, and study their occurrence in different parts of the body, Figure 6.6-(b). The correction of such errors, in particular *Miss*, can bring large improvements in the instance OKS and AP, especially at higher evaluation thresholds, Figure 6.6-(c-d).

Another important source of error is noise in the detections' confidence scores. To minimize errors, the scores should be *i)* 'OKS monotonic increasing' and *ii)* cali-

brated over the whole dataset, Section 6.3.2. The *optimal score* of a given detection corresponds to the maximum OKS value obtainable with any annotation. Replacing a method's scores with the optimal scores yields an average AP improvement of 5%, Figure 6.8-(a), due to the fact that ground-truth instances match detections that obtain higher OKS, and the overall number of matches is increased, Table 6.2. A key property of good scoring functions is to separate as much as possible the distribution of confidence scores for detections obtaining high OKS versus low OKS, Figure 6.8-(c).

Characteristics of the portrayed people, such as the amount of overlap with other instances and the number of visible keypoints, substantially affects performance. A comparison between Figure 6.11-(a) and Table 6.1, shows that average performance strongly depends on the properties of the images, and that state-of-the-art methods still vastly underperform humans when multiple people overlap and significant occlusion is present. Since COCO is not rich in such challenging pictures, it remains to be seen whether poor performance is due to the low number of training instances, Figure 6.12-(b), and a new collection and annotation effort will be needed to investigate this question. The size of instances also affects the quality of the detections, Figure 6.13-(a), but is less relevant than occlusion or crowding. This conclusion may be biased by the fact that small instances are not annotated in COCO and excluded from our analysis.

In this study, we also observe that, despite their design differences, [2, 6] display similar error patterns. Nonetheless, [2] is more sensitive to *swap* errors, as keypoint predictions from the entire image can be erroneously grouped into the same instance, while [6] is more prone to *misses*, as it only predicts keypoint locations within the detected bounding box. [6] has more than twice the number of high confidence *FP* errors, compared to [2]. Finally, we observe that *FN* are predominant around the image border for [2], where grouping keypoints into consistent instances can be harder, and concentrated in the center for [6], where there is typically clutter and bounding boxes accuracy is reduced.

Based on the above considerations, we provide some practical guidelines that could lead to substantial improvements of pose estimation algorithm's performance.

Improving Localization: 3D reasoning along with the estimation of 2D body parts [37] can improve localization by both incorporating constraints on the anatomical validity of the body part predictions, and learning priors on where to

expect visually occluded parts. Two promising directions for improvement are possible: *i)* collecting 3D annotations [38] for the humans in COCO and learning to directly regress 3D pose end-to-end [39]; *ii)* modeling the manifold of human poses [40–42] and learning how to jointly predict the 3D pose of a person along with its 2D skeleton [43].

Improving Scoring: Graphical models [44] can be used to learn a scoring function based on the relative position of body part locations, improving upon [2, 6] which only use the confidence of the predicted keypoints. Another promising approach is to use the validation set to learn a regressor for estimating optimal scores (Section 6.3.2) from the confidence maps of the predicted keypoints and from the sub-optimal detection scores generated by the algorithm. Comparing scores of detections in the same image relatively to each other will allow optimizing their order.

References

[1] A. Bulat and G. Tzimiropoulos,
"Human pose estimation via convolutional part heatmap regression",
in *European Conference on Computer Vision*, Springer, 2016, pp. 717–732
(cit. on pp. 58, 63).

[2] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields",
ArXiv preprint arXiv:1611.08050, 2016
(cit. on pp. 58, 62, 63, 64, 67, 68, 71, 72, 73, 75, 76, 78, 79, 80).

[3] X. Chen and A. L. Yuille, "Articulated pose estimation by a graphical model with image dependent pairwise relations",
in *Advances in Neural Information Processing Systems*, 2014,
pp. 1736–1744 (cit. on p. 58).

[4] G. Gkioxari, B. Hariharan, R. Girshick, and J. Malik,
"Using k-poselets for detecting people and localizing their keypoints",
in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3582–3589 (cit. on pp. 58, 62).

[5] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation", *ArXiv preprint arXiv:1603.06937*, 2016
(cit. on pp. 58, 63).

[6] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy, "Towards accurate multi-person pose estimation in the wild", *ArXiv preprint arXiv:1701.01779*, 2017
(cit. on pp. 58, 62, 63, 64, 67, 68, 71, 72, 73, 75, 79, 80).

[7] L. Pishchulin, A. Jain, M. Andriluka, T. Thormählen, and B. Schiele,
"Articulated people detection and pose estimation: Reshaping the future",
in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, IEEE, 2012, pp. 3178–3185 (cit. on pp. 58, 62).

[8] V. Ramakrishna, D. Munoz, M. Hebert, A. J. Bagnell, and Y. Sheikh,
"Pose machines: Articulated pose estimation via inference machines",
in *ECCV*, 2014 (cit. on pp. 58, 63).

[9] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh,
"Convolutional pose machines", in *CVPR*, 2016 (cit. on pp. 58, 63).

[10] Y. Yang and D. Ramanan,
"Articulated pose estimation with flexible mixtures-of-parts", in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, IEEE, 2011, pp. 1385–1392 (cit. on pp. 58, 62).

[11] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2d human pose estimation: New benchmark and state of the art analysis", in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2014 (cit. on pp. 58, 60).

[12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context", in *Computer Vision–ECCV 2014*, Springer, 2014, pp. 740–755 (cit. on pp. 58, 60).

[13] S. Johnson and M. Everingham, "Clustered pose and nonlinear appearance models for human pose estimation", in *Proceedings of the British Machine Vision Conference*, doi:10.5244/C.24.12, 2010 (cit. on p. 58).

[14] ——, "Learning effective human pose estimation from inaccurate annotation", in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2011 (cit. on p. 58).

[15] M. Eichner and V. Ferrari, "We are family: Joint pose estimation of multiple persons", in *European Conference on Computer Vision*, Springer, 2010, pp. 228–242 (cit. on pp. 58, 60, 62).

[16] S. Branson, G. Van Horn, S. Belongie, and P. Perona, "Bird species categorization using pose normalized deep convolutional nets", *ArXiv preprint arXiv:1406.2952*, 2014 (cit. on p. 59).

[17] X. Burgos-Artizzu, P. Dollár, D. Lin, D. Anderson, and P. Perona, "Social behavior recognition in continuous videos", in *CVPR*, 2012 (cit. on p. 59).

[18] E. Eyjolfsdottir, S. Branson, X. P. Burgos-Artizzu, E. D. Hoopfer, J. Schor, D. J. Anderson, and P. Perona, "Detecting social actions of fruit flies", in *European Conference on Computer Vision*, Springer, 2014, pp. 772–787 (cit. on p. 59).

[19] D. Hoiem, Y. Chodpathumwan, and Q. Dai, "Diagnosing error in object detectors", in *European conference on computer vision*, Springer, 2012, pp. 340–353 (cit. on p. 59).

[20] P. Dollár, P. Welinder, and P. Perona, "Cascaded pose regression", in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, IEEE, 2010, pp. 1078–1085 (cit. on p. 59).

[21] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele, "Deepcut: Joint subset partition and labeling for multi person pose estimation", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4929–4937 (cit. on pp. 60, 63).

[22] *COCO Keypoints Evaluation.* `http://mscoco.org/dataset/#keypoints-eval`, October, 2016. (cit. on pp. 60, 61, 64).

[23] *COCO Keypoints Challenge, ECCV 2016.* `http://image-net.org/challenges/ilsvrc+coco2016`, October, 2016. (cit. on pp. 61, 62, 63, 66).

[24] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge", *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010 (cit. on p. 61).

[25] J. Hosang, R. Benenson, P. Dollár, and B. Schiele, "What makes for effective detection proposals?", *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 4, pp. 814–830, 2016 (cit. on p. 61).

[26] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks", in *Advances in neural information processing systems*, 2015, pp. 91–99 (cit. on pp. 62, 63).

[27] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning", *ArXiv preprint arXiv:1602.07261*, 2016 (cit. on p. 62).

[28] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9 (cit. on p. 62).

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778 (cit. on pp. 62, 63).

[30] C. Szegedy, S. Reed, D. Erhan, and D. Anguelov, "Scalable, high-quality object detection", *ArXiv preprint arXiv:1412.1441*, 2014 (cit. on p. 63).

[31] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields", in *CVPR*, 2017 (cit. on p. 63).

[32] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele, "Deepercut: A deeper, stronger, and faster multi-person pose estimation model", in *European Conference on Computer Vision*, Springer, 2016, pp. 34–50 (cit. on p. 63).

[33] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", *ArXiv preprint arXiv:1409.1556*, 2014 (cit. on p. 63).

[34] D. B. West *et al.*, *Introduction to graph theory*.
Prentice hall Upper Saddle River, 2001, vol. 2 (cit. on pp. 63, 72).

[35] H. W. Kuhn, "The hungarian method for the assignment problem",
Naval research logistics quarterly, vol. 2, no. 1-2, pp. 83–97, 1955
(cit. on pp. 63, 72).

[36] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Improving object
detection with one line of code", *ArXiv preprint arXiv:1704.04503*, 2017
(cit. on p. 74).

[37] C. J. Taylor, "Reconstruction of articulated objects from point
correspondences in a single uncalibrated image", in *Computer Vision and
Pattern Recognition, 2000. Proceedings. IEEE Conference on*, IEEE,
vol. 1, 2000, pp. 677–684 (cit. on p. 79).

[38] L. Bourdev and J. Malik,
"Poselets: Body part detectors trained using 3d human pose annotations",
in *Computer Vision, 2009 IEEE 12th International Conference on*, IEEE,
2009, pp. 1365–1372 (cit. on p. 80).

[39] G. Pavlakos, X. Zhou, K. G. Derpanis, and K. Daniilidis, "Coarse-to-fine
volumetric prediction for single-image 3d human pose",
ArXiv preprint arXiv:1611.07828, 2016 (cit. on p. 80).

[40] I. Akhter and M. J. Black,
"Pose-conditioned joint angle limits for 3d human pose reconstruction",
in *Proceedings of the IEEE Conference on Computer Vision and Pattern
Recognition*, 2015, pp. 1446–1455 (cit. on p. 80).

[41] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black,
"Keep it smpl: Automatic estimation of 3d human pose and shape from a
single image", in *ECCV*, 2016 (cit. on p. 80).

[42] M. R. Ronchi, J. S. Kim, and Y. Yue, "A rotation invariant latent factor
model for moveme discovery from static poses",
ArXiv preprint arXiv:1609.07495, 2016 (cit. on p. 80).

[43] D. Tome, C. Russell, and L. Agapito, "Lifting from the deep: Convolutional
3d pose estimation from a single image", *CVPR*, 2017 (cit. on p. 80).

[44] D. Koller and N. Friedman,
Probabilistic graphical models: Principles and techniques. MIT press, 2009
(cit. on p. 80).

Part II

Recovering 3D Human Pose and Motion from Images and Videos using Weak Supervision

Chapter 7

A ROTATION INVARIANT LATENT FACTOR MODEL FOR MOVEME DISCOVERY FROM STATIC POSES

The contents of this chapter are adapted from the peer-reviewed publication:

M. R. Ronchi, J. S. Kim and Y. Yue *"A Rotation Invariant Latent Factor Model for Moveme Discovery from Static Poses."* IEEE 16th International Conference on Data Mining (2016, Barcelona, Spain).

DOI: 10.1109/ICDM.2016.0156

URL: http://www.vision.caltech.edu/~mronchi/projects/RotationInvariantMovemes

We tackle the problem of learning a rotation invariant latent factor model when the training data is comprised of lower-dimensional projections of the original feature space. The main goal is the discovery of a set of 3D bases poses that can characterize the manifold of primitive human motions, or movemes, from a training set of 2D projected poses obtained from still images taken at various camera angles. The proposed technique for basis discovery is data-driven rather than hand-designed. The learned representation is rotation invariant, and can reconstruct any training instance from multiple viewing angles. We apply our method to modeling human poses in sports (via the Leeds Sports Dataset), and demonstrate the effectiveness of the learned bases in a range of applications such as activity classification, inference of dynamics from a single frame, and synthetic representation of movements.

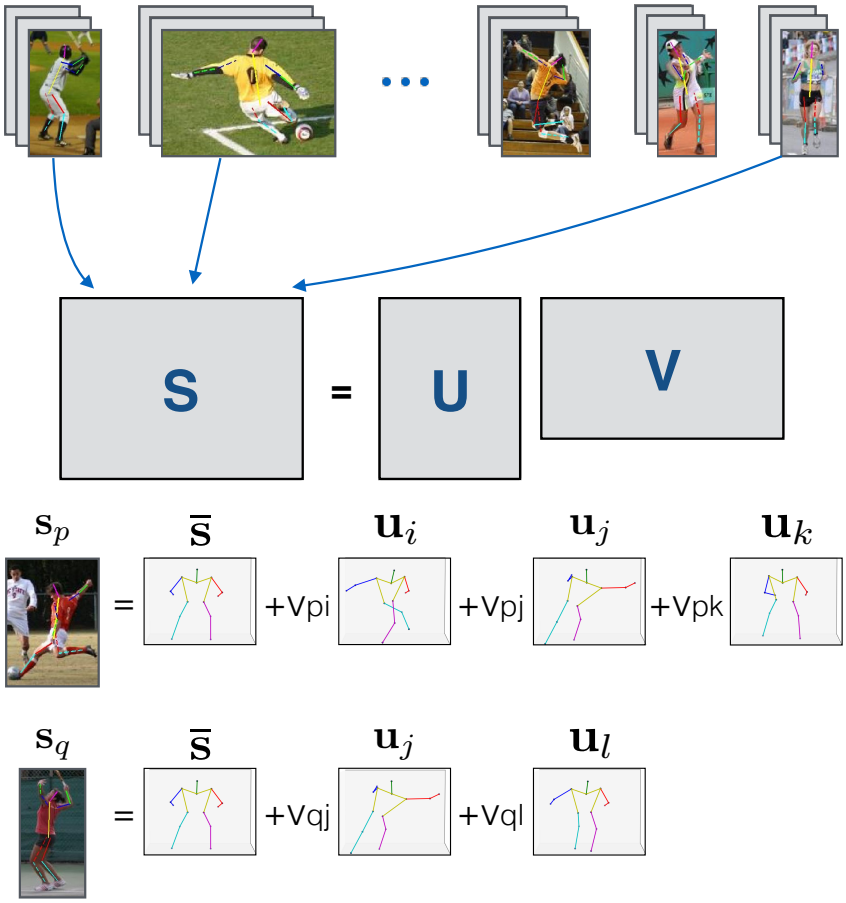


Figure 7.1: **Rotation invariant moveme discovery.** Given a matrix of static joint locations annotated from images taken at any angle of view, we learn a factorization into a basis pose matrix **U** and a coefficient matrix **V**. The learned poses in **U** *i)* are rotation-invariant, as they capture the change in pose underlying a specific movement regardless of the angle of view at which it is portrayed; *ii)* span the space of all human poses, as a sparse linear combination of the bases accurately reconstructs the pose of a human involved in an action at any angle of view, also for poses not contained in the training set.

7.1 Introduction

What are the typical ranges of motion for human arms? What types of leg movements tend to correlate with specific shoulder positions? How can we expect the arms to move given the current body pose? Our goal is to address these questions by recovering a set of "bases poses" that summarize the variability of movements in a given collection of static poses captured from images at various viewing angles.

One of the main difficulties of studying human movement is that it is a priori unrestricted, except for physically imposed joint angle limits which have been studied in medical text books, typically for a limited number of configurations [1, 2].

Furthermore, human movement may be distinguished into *movemes*, *actions*, and *activities* [3, 4] depending on structure, complexity, and duration. *Movemes* refer to the simplest meaningful pattern of motion: a short, target-oriented trajectory, that cannot be further decomposed, e.g. "reach", "grasp", "step", "kick". A complex gesture should be composed out of simple movemes: we define an action as a predefined and ordered sequence of movemes, such as "drink from a glass", or "open a door". An activity is a (possibly stochastic) combination of actions taking place over a stretch of time with a typical and yet variable structure, e.g. "dine", "read". Extensive studies have been carried out on human action and activity recognition [5, 6], however little attention has been paid to movemes since human behaviour is difficult to analyze at such a fine scale of dynamics.[1]

In this line of research, our primary goal is to learn a basis space to smoothly capture movemes from a collection of two dimensional images, although our learned representation can also aid in higher level reasoning.

Static poses extracted from two-dimensional images are the most abundant source of pose information. Thus, finding a basis representation using such training data can prove extremely valuable, given the number of image datasets (as opposed to video or mo-cap data) that are currently being collected with a focus on common activities [8–10]. However, such images are typically taken from a wide range of viewing angles, and can yield only two-dimensional projections of the underlying three-dimensional pose. Any method that does not directly address these issues will learn a naive representation that fails to provide a set of global three-dimensional bases poses that can capture pose changes due to the true human motion while disregarding those due to a change of the angle of view.

We propose a rotation invariant latent factor model that can recover a set of three-dimensional bases poses from a training set of two-dimensional projections. Our approach is distinguished from previous latent factor modeling approaches by directly incorporating geometric operations in an integrated way, and yields interpretable three dimensional bases poses that can be easily visualized as well as manipulated to express a natural range of human poses (as depicted in Figure 7.1). We applied our approach in a case study on modeling poses that arise in sports activities, since they have very characteristic and recognizable motions and typically share trajectories

[1]The extent in time and complexity of human motion is not directly observable in still images, but requires videos of humans involved in activities which cannot be recorded extensively without legal or ethical issues, as opposed to fly or mouse behaviour which is very well documented [7].

of parts of the body (e.g., tennis serve and volleyball strike), which allows to more easily interpret and evaluate qualitatively the learned movemes.

Our study is not purely academic, as we have four applications in mind. We carry out a quantitative and qualitative analysis for the first two of them, and leave the study of the latter to future work:

Activity recognition: A compact representation, such as the proposed one, can be used in addition to the feature representation of state-of-the-art methods for activity recognition, favoring both the performance [11], and the interpretability of results.

Action dynamics inference: Modifying the weights of the learned bases poses is analogous to moving along a line in the high-dimensional space of human poses (either 2D or 3D). This allows to predict the future dynamic of an action [12], or morph a pose into another from a single frame, by observing the dynamics of the movemes which better describe the captured pose.

Computer graphics animation: many animation systems are still based on *key-framing* and *in-betweening* [13]: master animators draw the key frames of a sequence to be animated and assistant animators complete the intermediate frames by inferring the movements occurring between the keys. Knowing the movemes underlying human actions would provide an automated method for interpolating between key frames, resulting in a faster and simplified animation pipeline.

3D pose estimation: a sparse overcomplete dictionary of human poses has been used effectively for the reconstruction of 3D human pose given its 2D joint locations from a single frame image [14–16]. Our technique would allow to identify the most suited pose bases for a given collection of images without any experimenter bias, or the need of curating the angle of view of the images in the training set.

In summary, the main contributions of this line of investigation are:

1. An **unsupervised** method for learning a **rotation-invariant** set of **bases poses**. We propose a solution to the intrinsically ill-posed problem of going from static poses to movements, without being affected by the angle of view.

2. A demonstration of how the learned bases poses can be used in various applications, including **manifold traversal**, **discriminative classification**, and **synthesis** of movements.

7.2 Related Work

Human Pose Analysis

There are two main directions of research for human pose analysis. The first one is estimation: given a picture containing a person, the goal is to predict the location of a predefined set of joints of its body, either in the 2D image [17, 18] or in the 3D space [14–16]. Methods for 3D pose reconstruction build upon the results of 2D pose estimators by using mechanisms based on physical constraints and domain knowledge to infer the true underlying human pose observed in an image, and are more of interest in this study since they implicitly learn an overcomplete basis for modeling human movement. However, such methods typically predefine the dictionary of actions, use additional data in the training phase (such as mo-cap), and do not treat explicitly the problem of varying angles of view. In contrast, our goal is to learn a low-rank manifold of 3D poses consistent across multiple viewing angles, given only two-dimensional data.

The second line of investigation uses pose as a form of contextual information that can be combined with objects' category and location in an image, to obtain higher performance for activity recognition through joint learning procedures [19–21]. Our approach can as well be used as a feature representation for improved activity recognition.

From the perspective of pose analysis, the goal of this work is to learn a semantically meaningful representation of human pose that can also be used to model human motion. This representation should be independent of the application domain, and flexible, allowing it to be incorporated with other representations. Other people investigated this problem: it is known that dynamic information can be recovered from static images of humans engaged in activities [22], and similar representations for action recognition have been learned using video data [23, 24].

We are the first to propose a method for learning a representation of human motions that *i)* learns the best basis autonomously without any experimenter bias, *ii)* directly treats the problem of rotation-invariance, and *iii)* can be learned only from static 2D poses as opposed to motion capture datasets [25], which is a fundamental contribution since it is the most abundant form of data available.

Latent Factor Models and Representation Learning

We build upon a long line of research in latent factor models, first popularized for collaborative filtering problems in content recommendation [26]. Applications include modeling variations of faces [27], document and text analysis [28], and behavior patterns in sports [29], amongst many others. Latent factor models are variants of matrix and tensor factorization, which can easily incorporate missing values or other types of constraints. In this regard, our work introduces an approach for learning a latent factor model in a high-dimensional space, when the observed training data are lower-dimensional projections. Our method is complementary to and can be integrated with other latent factor modeling approaches.

Our approach can be viewed as a form of representation learning, which includes methods such as deep neural networks and dictionary learning [30, 31]. One of the benefits of representation learning is the ability to smoothly traverse the representation space [32], which in our setting translates to learning movemes as transitions between poses.

7.3 Models

We develop our approach by building from the classical singular value decomposition. We characterize the challenge of learning only from lower-dimensional projections of the underlying feature space, and present a rotation-invariant latent factor model for dealing with such training data.

7.3.1 Basic Notation and Framework

In this work, we focus on learning from two-dimensional projections of three-dimensional human poses, however, it is straightforward to generalize to other settings. We are given a training set $S = \{(\mathbf{x}_j, \mathbf{y}_j)\}_{j=1}^{n}$ of n two-dimensional poses, where x and y correspond to the image coordinates of the pose joints from the observed viewing angle, see Figure 7.2. Let $\mathbf{S} \in \mathfrak{R}^{2d \times n}$ denote the dataset matrix, where $2d$ is the dimensionality of the projected space (twice the number of joints d for two-dimensional projections). Our goal is to learn a bases poses matrix $\mathbf{U} \in \mathfrak{R}^{2d \times k}$ composed of k latent factors, and a coefficient matrix $\mathbf{V} \in \mathfrak{R}^{k \times n}$, so that every training example can be represented as a linear combination:

$$\mathbf{s}_j = \mathbf{U} \cdot \mathbf{v_j} + \bar{\mathbf{s}}, \tag{7.1}$$

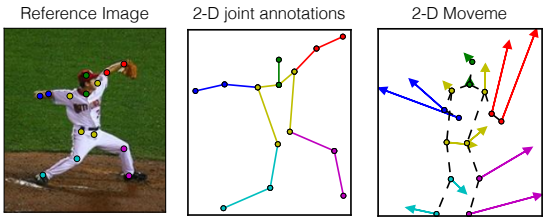


Figure 7.2: **Moveme representation in the 2D image plane.** The human body landmark annotations from an image in LSP [33], and their displacement from the mean pose, which we use to encode movemes.

where $\bar{\mathbf{s}}$ denotes the "mean" pose. Of course, Eq. 7.1 does not deal with rotation invariance and treats the x and y coordinates as having the same semantics across training examples. We present in Section 7.3.3 a rotation-invariant latent factor model to address this issue and recover a three-dimensional $\mathbf{U} \in \Re^{3d \times k}$.

7.3.2 Baselines

To the best of our knowledge, no existing approach tackles the problem of learning a rotation-invariant bases for modeling human movement. Previous work is focused on either learning bases poses only from frontal viewing angles or by extensive manual crafting of a predefined set of poses [14, 16]. As such, we develop our approach by building upon classical baselines such as the SVD, which we briefly describe here.

Singular Value Decomposition

The example in Eq. 7.1 is the most basic form of a latent factor model. When the training objective is to minimize the squared reconstruction error of the training data, then the solution can be recovered via SVD, also used for eigenfaces [27]. The bases matrix $\mathbf{U}$ and the coefficient matrix $\mathbf{V}$, respectively, correspond (up to a scaling) to the left and right singular vectors of the mean-centered data matrix $\mathbf{S}_c = (\mathbf{S} - \bar{\mathbf{s}})$. However, naively applying the SVD to our setting will result in the bases matrix $\mathbf{U}$ conflating viewing angle rotations with true pose deformations.

Clustered Singular Value Decomposition

If the viewing angle of the training data is available, or a quantized approximation of it, then the basic latent factor model in Eq. 7.1 can be instantiated separately for different viewing angles, via:

$$\mathbf{s}_j = \mathbf{U}(a_j) \cdot \mathbf{v_j} + \bar{\mathbf{s}}(a_j), \tag{7.2}$$

where a_j denotes the viewing angle cluster that example j belongs to. In other words, given p clusters, we learn p separate latent factor models, one per cluster. Intuitively, we expect this method to suffer less conflation between changes in pose due to a viewing angle rotation and true pose deformation, and the more clusters, the less susceptible. The main drawbacks are that: *i)* the learned bases representation is not global, and will not be consistent across the clusters since they are learned independently, and *ii)* the amount of training data per model is reduced, which can yield a worse representation.

7.3.3 Rotation-Invariant Latent Factor Model

Our goal is to develop a latent factor model that can learn a global representation of bases poses across different angles. For simplicity, we restrict ourselves to settings where there are only differences in the pan angle, and assume no variation in the tilt angle (i.e., all horizontal views). To that end, we propose both a 2D and a 3D model which can be used depending on the quality and quantity of additional information available at training time. For some applications, it may suffice to use the 2D model, however the 3D model is generally better able to intrinsically capture rotation-invariance.

We first motivate some of the desirable properties:

- **Unsupervised** – the bases discovery should not be limited to or dependent on images of specific classes of actions.
- **Rotation Invariant** – the learned bases should be composed of movements from a given canonical view (e.g., frontal) and be able to reconstruct poses oriented at any angle. The exact same pose may look different when observed from different camera angles; as such, it is important to disambiguate pose from viewing angle.
- **Sparse** – to encourage interpretability, the learned bases should be sparsely activated for any training instance.
- **Complementary** – our method should be easy to integrate with other modeling approaches, and thus should implement an orthogonal extension of the basic latent factor modeling framework.

General Framework

Our general framework aims to learn a latent factor matrix $\mathbf{U}$, containing the bases poses instantiated globally across all the training data; a coefficient matrix $\mathbf{V}$, whose columns correspond to the weights given to the bases poses to reconstruct all training instance; and a vector θ, containing the angle of view of each training pose.

We can thus model every training example as:

$$\mathbf{s}_j = f(\bar{\mathbf{s}} + \mathbf{U} \cdot \mathbf{v}_j, \theta_j), \tag{7.3}$$

where $f(\cdot, \cdot)$ is a projection operator of the higher-dimensional model into the two-dimensional space. We train our model via:

$$\mathbf{U}, \mathbf{V}, \theta = \arg\min_{\mathbf{U},\mathbf{V},\theta} \mathcal{L}(\mathbf{U}, \mathbf{V}, \theta), \tag{7.4}$$

$$\mathcal{L}(\mathbf{U}, \mathbf{V}, \theta) = \mathcal{E}(\mathbf{U}, \mathbf{V}, \theta) + \Omega(\mathbf{U}, \mathbf{V}, \theta), \tag{7.5}$$

$$\mathcal{E}(\mathbf{U}, \mathbf{V}, \theta) = \sum_j \left(\mathbf{s}_j - f(\bar{\mathbf{s}} + \mathbf{U} \cdot \mathbf{v}_j, \theta_j)\right)^2, \tag{7.6}$$

where $\mathcal{E}$ is the squared reconstruction error over the training instances, and Ω is a model-specific regularizer. The projection operator f and the regularizer Ω are specified separately for the 2D and 3D approach. This optimization problem is non-convex, and requires a reasonable initialization in order to converge to a good local optimum.

2D approach

The 2D approach, uses the same approach as the clustered SVD baseline and, given a set of p angle clusters, instantiates the projection operator as:

$$f(\bar{\mathbf{s}} + \mathbf{U} \cdot \mathbf{v}_j, \theta_j) = \bar{\mathbf{s}}(a_j) + \mathbf{U}(a_j) \cdot \mathbf{v}_j, \tag{7.7}$$

a_j denotes the cluster that θ_j belongs to, and a separate rank-k $\mathbf{U}$ is learned for each viewing angle cluster. At this point, Eq. 7.7 looks identical to Eq. 7.2. However, we encourage global consistency between the per-cluster models via the regularization terms:

$$\Omega(\mathbf{U}, \mathbf{V}, \theta) = R_{reg}(\mathbf{U}, \mathbf{V}, \theta) + R_{spat}(\mathbf{U}, \mathbf{V}, \theta). \tag{7.8}$$

The first term in Eq. 7.8 is a standard regularizer used to prevent overfitting:

$$R_{reg}(\mathbf{U}, \mathbf{V}, \theta) = \sum_{a=1}^{p} \left[\lambda_U \|\mathbf{U}(a)\|_F^2 + \lambda_V \|\mathbf{V}(a)\|_1 \right]. \tag{7.9}$$

We wish to have sparse activations so we regularize $\mathbf{V}$ using L1 norm. Depending on the application, Section 7.4, we sometime enforce that $\mathbf{V}$ be non-negative for added interpretability.

The second term in Eq. 7.8 is the spatial regularizer that encourages (or in some cases enforces) consistency across the per-cluster models:

$$R_{spat}(\mathbf{U}, \mathbf{V}, \theta) = \lambda_{spat} \sum_{a,a'} \kappa_{a,a'} \|\mathbf{U}^{(x)}(a) - \mathbf{U}^{(x)}(a')\|_F^2 \tag{7.10}$$

$$+ \sum_{a,a'} \mathbf{1}\left(\mathbf{U}^{(y)}(a), \mathbf{U}^{(y)}(a') \right), \tag{7.11}$$

$\mathbf{U}^{(x)}$ and $\mathbf{U}^{(y)}$ represent the x and y coordinate portions of the bases poses: e.g. $\mathbf{U}^{(x)} = [\mathbf{U}_{i,-}]$, $i \in X$, where X is the set of indices corresponding to x coordinates in the pose representation. Since we are only modeling variations in the pan angle, the x coordinates can vary across different viewing angles, while the y coordinates should remain constant. As such, the first term in R_{spat}, Eq. 7.10, corresponds to encouraging the $\mathbf{U}^{(x)}(a)$ and $\mathbf{U}^{(x)}(a')$ of different clusters to be similar to each other (with $\kappa_{a,a'}$ controlling the degree of similarity), and the second term, Eq. 7.11, is a $\{0, \infty\}$ indicator function that takes value 0 if the two arguments are identical, and value ∞ if they are not (i.e., it is a hard constraint).

In summary, the spatial regularization term is the main difference between the 2D latent factor model and the clustered SVD baseline. Global consistency of the per-cluster models is obtained by encouraging similar values in the x coordinates, and enforcing identical y coordinates. In a sense, one can view spatial regularization as a form of multi-task regularization, which enables sharing statistical strength across the clusters. The main limitation of the 2D model is that the spatial regularization does not incorporate more sophisticated geometric constraints, so the notion of consistency achieved may not align with the true underlying three-dimensional data.

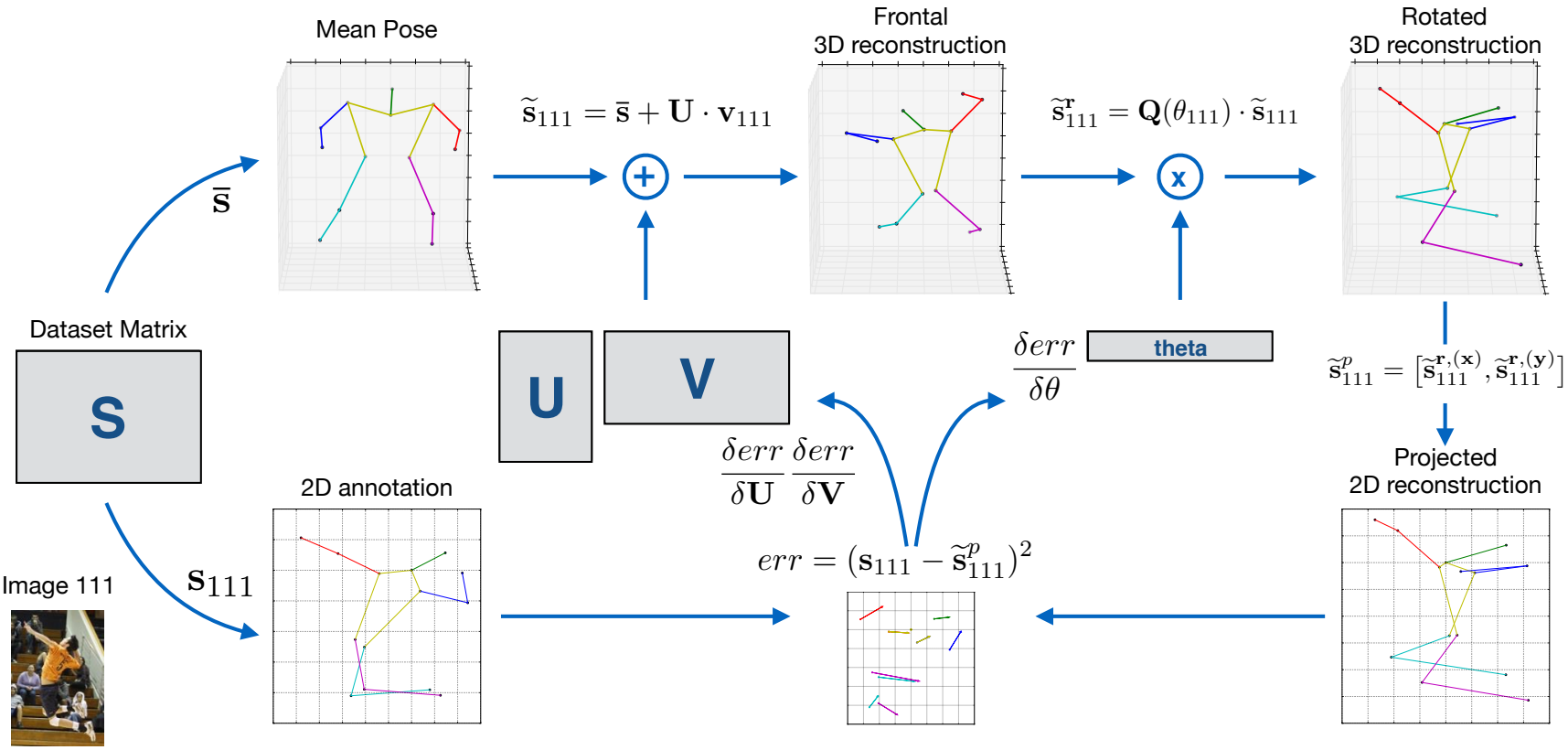


Figure 7.3: **Training pipeline of the proposed method LFA-3D.** Bases poses $\mathbf{U}$, coefficient matrix $\mathbf{V}$ and angles of view θ are initialized and updated through alternate stochastic gradient descent. Each iteration consists of the following steps: (1) a sparse linear combination of the current bases poses with coefficients from $\mathbf{V}$ is added to the dataset mean pose to obtain a frontal 3D reconstruction of the true pose; (2) the 3D reconstruction is rotated by the current estimate of the angle of view θ_j for that pose; (3) the 3D pose is projected to the 2D space where it is compared to the ground truth; (4) the gradient update step is computed to minimize the root mean square error *wrt.* to quantities $\mathbf{U}$, $\mathbf{V}$ and θ.

3D approach

The 3D model directly learns a three-dimensional representation of the underlying pose space, through a single and global $\mathbf{U} \in \mathfrak{R}^{3d \times k}$ that is inherently three-dimensional, and captures k bases poses.

The projection operator is now defined as:

$$f(\bar{\mathbf{s}} + \mathbf{U} \cdot \mathbf{v}_j, \theta_j) = \left[\mathbf{Q}(\theta_j) \left(\bar{\mathbf{s}} + \mathbf{U} \cdot \mathbf{v}_j \right) \right]^{(x,y)}, \tag{7.12}$$

where $\mathbf{Q}(\cdot)$ is the 3D rotation matrix around the vertical axis:

$$\mathbf{Q}(\theta_j) = \begin{bmatrix} \cos(\theta_j) & 0 & \sin(\theta_j) \\ 0 & 1 & 0 \\ -\sin(\theta_j) & 0 & \cos(\theta_j) \end{bmatrix}, \tag{7.13}$$

and the superscript $^{(x,y)}$ denotes the projection from the 3D space of $\mathbf{U}$ to the 2D space of the dataset annotations, obtained by indexing only the x and y coordinates (the underlying model provides x, y, and z coordinates). The projection operator in Eq. 7.12 allows to compute the two-dimensional projection of any underlying three-dimensional pose at any viewing angle θ_j using standard geometric rules. Spatial regularization is no longer needed, because the rotation operator $\mathbf{Q}$ relates all the viewing angles to a common model, thus the regularizer assumes the standard form:

$$\Omega(\mathbf{U}, \mathbf{V}, \theta) = \lambda_U \|\mathbf{U}\|_F^2 + \lambda_V \|\mathbf{V}\|_1. \tag{7.14}$$

In summary, the 3D latent factor model improves upon the 2D version by learning a global representation that is intrinsically three-dimensional and integrates domain knowledge of how the viewing angle affects pose via geometric projection rules. This results in a more robust method, that does not learn a separate model per viewing angle or rely on the spatial regularization to obtain consistency. The main drawback is that a more complex initialization will be required. Figure 7.3 provides an overview of the steps for the 3D approach.

7.3.4 Training Details

Initialization

Our approaches require an initial guess of the viewing angle for each training instance, and the bases poses $\mathbf{U}$. For angle initialization, we show in our experiments

(Section 7.4.5) that we only need a fairly coarse prediction of the viewing angle (e.g., into quadrants). The 2D latent factor model bases poses $\mathbf{U}$ are initialized uniformly between -1 and 1, while for the 3D model we use an off-the-shelf pose estimator [16] and initialize $\mathbf{U}$ as the left singular vectors of the mean centered 3D pose data, obtained through SVD.

Optimization

For both models, we optimize Eq. 7.4 using alternating stochastic gradient descent, divided in two phases:

- Representation Update: we employ standard stochastic gradient descent to update $\mathbf{U}$ and $\mathbf{V}$ while keeping θ fixed. For the 3D model, this involves computing how the training data (which are two-dimensional projections) induce a gradient on $\mathbf{U}$ and $\mathbf{V}$ through the rotation $\mathbf{Q}$. Because we employ an L1 regularization penalty, we use the standard soft-thresholding technique [34].
- Angle Update: Once the optimal $\mathbf{U}$ and $\mathbf{V}$ are fixed, we employ standard stochastic gradient descent to update θ.

Convergence and Learning Rates

Three training epochs of 10000 iterations are usually sufficient for convergence to a good local minimum. Typical values of the learning rate are 1×10^{-4} for $\mathbf{U}$ and $\mathbf{V}$ and 1×10^{-6} for θ. We use a smaller step size in the update of θ, since the curvature of the objective function in Eq. 7.4 is higher w.r.t. θ than w.r.t. $\mathbf{U}$ and $\mathbf{V}$.

7.4 Experiments and Empirical Results

We analyze the flexibility and usefulness of the proposed model in a variety of application domains and experiments. In particular, we evaluate *i)* the performance of the learned representation for supervised learning tasks such as activity classification; *ii)* whether the learned representation captures enough semantics for meaningful manifold traversal and visualization; and *iii)* the robustness to initialization and the generalization error. Collectively, results suggest that our approach is effective at capturing rotation invariant semantics of the underlying data.

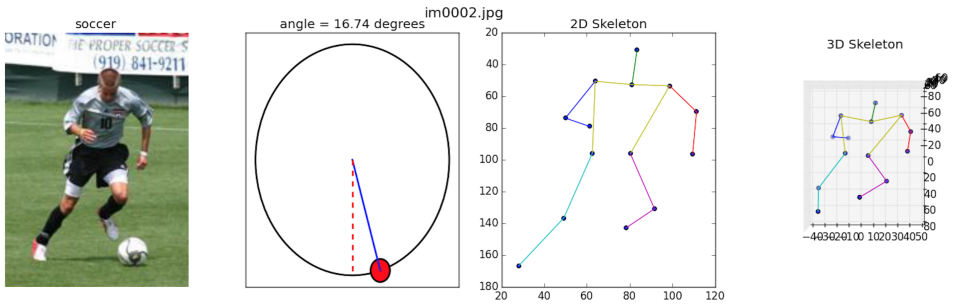


Figure 7.4: **Example annotation for the orientation angle collected on the LSP dataset.** For every image in the LSP dataset, three separate workers were asked to provide the visualized person's angle of orientation, described as the pan angle at which the person is facing with respect to the camera taking the picture.

7.4.1 Dataset and Additional Annotations

We use the Leeds Sports Dataset (LSP) [33] for our experiments. LSP is composed of 2000 images containing a single person performing one of eight sports (Athletics, Badminton, Baseball, Gymnastics, Parkour, Soccer, Tennis, Volleyball) annotated with the x,y location and a visibility flag for 14 joints of the human body. Example images and annotations are shown in Figures 7.1, 7.2, 7.4, and 7.9. Sports activities are particularly well suited for this study, as they present characteristic motions that share trajectories of parts of the body, that allow investigating basis pose sharing across sports. As part of preprocessing, we normalize all the poses in the dataset by modifying each bone to have the average bone length computed over all the training instances [15]. We discard "Gymnastics" and "Parkour" from our analysis because they have few examples and the class poses do not vary exclusively along the pan angle (but appear in very unconventional views, i.e. upside-down and horizontal), violating the assumption in Section 7.3.3. Generalizing the framework, to incorporate a wider variability of the viewing angles, is an interesting future direction.

We developed an Amazon Mechanical Turk GUI to collect high-quality orientation angle annotations for each pose in LSP, as shown in Figure 7.4. Although these annotations are not necessary for training our model, we use them to demonstrate the robustness to poor angle initialization, and that it can in fact recover the ground-truth value at evaluation time, see Section 7.4.5. Three annotators evaluated each image and were instructed to provide the direction at which the torso was facing, and we combined their answers using a simple average. The standard deviation in the reported angle of view averaged over the whole dataset is 12 degrees, and more than half of the images have a deviation of less than 10 degrees, showing a very high annotator agreement for the task.

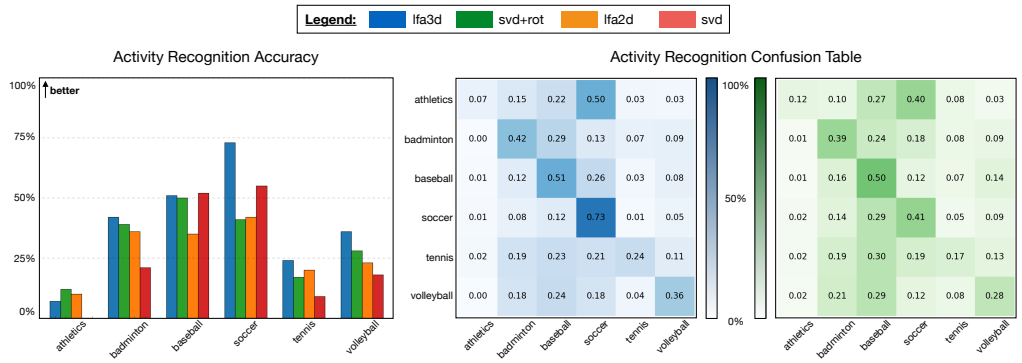


Figure 7.5: **Performance analysis on the activity recognition task.** (Left) The activity classification accuracy across the sports in LSP for the following methods: "svd" – baseline, "svd+rot" – clustered version of the baseline, "lfa2d" – 2D latent factor model with spatial regularization, "lfa3d" – full 3D latent factor model. (Right) The confusion tables for the best two performing methods, "lfa3d" and "svd+rot". Full details in Section 7.4.2.

7.4.2 Activity Recognition

The matrix $\mathbf{V}$ describes each pose in the dataset as a linear combination of the learned latent factors, Section 7.3.1. Thus, $\mathbf{v}_j$ can be interpreted as a semantically more meaningful feature representation for j-th data point. For instance, if a lower body basis pose (e.g. Figure 7.7 top row) has a high weight, the reconstructed pose is very likely to represent a movement from an activity related to running, or kicking.

A natural way to test the effectiveness of the learned representation is to use it for supervised learning tasks. To that end, we used the coefficients in $\mathbf{V}$ as input features for classifying the sport categories in LSP.

Figure 7.5 shows the results obtained from five-fold cross validation. The proposed 3D latent factor model ("lfa3d") outperforms all other methods by an average accuracy of about 11%. The 2D model ("lfa2d") performs slightly worse than the clustered SVD baseline ("svd+rot"), but both show more than a 5% average improvement over the "svd" baseline. The two most challenging activities are "athletics", which does not posses characterizing movements; and "tennis", whose movemes are shared and thus confused with multiple other sports, "badminton" and "baseball" above all. We also report the full classification confusion tables in Figure 7.5. Note that only the weights of the latent factors reconstructing a pose are being used to discriminate between the activities, without the aid of visual cues from the image. It is thus surprising that "lfa3d" achieves an average 39% accuracy, when a random guess would merely give 16.7%. Finally, the obtained feature representation is complementary to other representations, such as the hidden

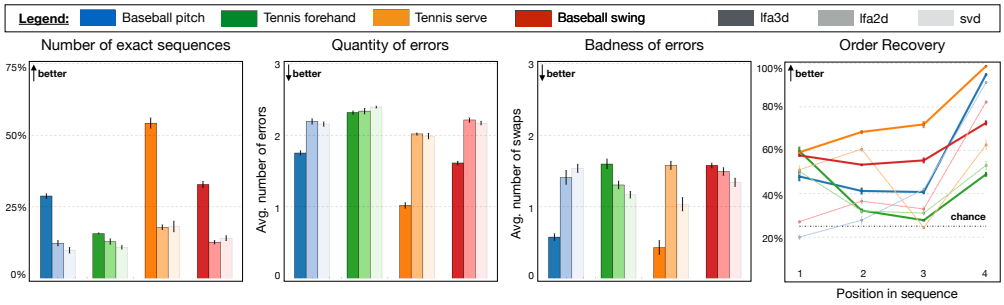


Figure 7.6: **Performance analysis on the action dynamics inference task.** We compare the methods "svd", "lfa2d", and "lfa3d" in the task of reordering shuffled sequences of images sampled from four different sport actions. The color scheme represents actions, and the methods are plotted with a different transparency value. The performance is described in terms of: (1) number of sequences exactly reordered; (2) average number of errors contained in a sequence; (3) average number of swaps needed to obtain the correct sequence; and (4) accuracy per position in the sequence – shown only for the best two methods ("lfa3d" - dark marker, "lfa2d" - light marker). Example sequences are shown in Table 7.1, and the full experiment details are explained in Section 7.4.3.

layer activations of a convolutional neural network [35], and we wish to investigate in future work the performance obtained by their combination.

7.4.3 Action Dynamics Inference & Manifold Traversal

Every pose in the training set belongs to a movement of the body corresponding to a complex trajectory in the manifold of human motion. If the latent factor model captures the semantics of the data, then poses that occur in chronological order within a given action should lie in a monotonic sequence within the learned space.

A quantitative measure of the quality of the representation can be obtained by observing how well the order of poses belonging to a same action is preserved. One straightforward way to find the sequence in which a set of poses lies in the manifold, is to look at the coefficient of their projection along the "total least squares" line fit [36] of the corresponding columns in the matrix $\mathbf{V}$. In other words, we are computing a linear traversal through the representation space. Furthermore, this ordering should hold regardless of the angle of view of the input instances.

In this experiment, we shuffled 1000 sequences of four images for four sport actions ("baseball pitch", "tennis forehand", "tennis serve", "baseball swing"), and verified how precisely could the underlying chronological sequence be recovered. The analysis is repeated five times to obtain standard deviations, and performance is

Method	Reordered Sequence	Errors	Swaps
lfa3d		0	0
lfa2d		2	1
svd		2	3

Table 7.1: **Example of the chronological re-ordering of a sequence of images using the LFA-3D method.** The coefficients in **V** are used to chronologically order four images sampled from a "tennis serve" sequence. We report the number of errors (images out of position) and swaps necessary to obtain the correct order.

measured in terms of three metrics: (1) what percentage of the 1000 sequences is exactly reordered; (2) how many poses are wrongly positioned; and (3) how bad are the reordering mistakes, computed as the number of swaps necessary to sort a predicted sequence in the correct order.

Figure 7.6 shows the results for the latent factor models "lfa2d", "lfa3d", and for the "svd" baseline. It is not possible to study the performance of the clustered baseline "svd+rot" since it does not learn a global matrix **U**, thus the coefficients in **V** are not comparable across different viewing angles. The "lfa3d" model has significantly better outcomes compared to "lfa2d" and "svd", which perform similarly. Specifically, "lfa3d" correctly reorders more than twice the sequences overall (1314 against 555 of "lfa2d") averages 1.6 errors, and is the only algorithm to require an average number of swaps smaller than 1. Figure 7.6-(4) shows the per-position accuracy.

In Table 7.1, we show the results of the reordering task for an example tennis "serve sequence", and only the "lfa3d" method recovers the order correctly. Note how the images are all taken from different viewing angles.

7.4.4 Moveme Visualization

The "lfa3d" method can be used to recover and synthesize realistic human motions from static joint locations in images. The underlying idea, is that, as long as the time scale is small, linear approximations of human motion can be successfully learned from observations of poses of people performing various actions, as opposed to

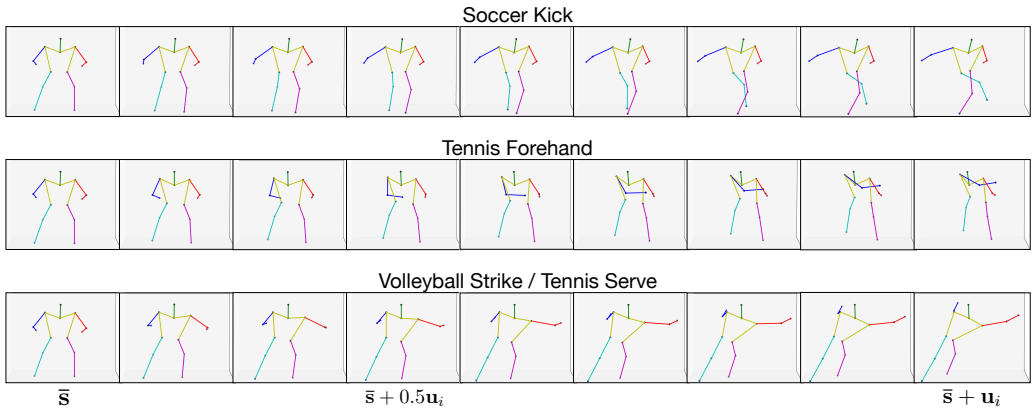


Figure 7.7: **Time evolution visualization for three learned movemes.** Three latent factors, encoding movemes, from the learned bases poses matrix **U**; two are easily interpretable ("soccer kick", "tennis forehand") and one is not as well defined ("volleyball strike / tennis serve"). The sequences are obtained by adding an increasing fraction of the basis to the mean pose of the dataset and differentiate very clearly, as early as 30% of the final movement, as visible in the second column. Full details in Section 7.4.4.

deriving mathematical principles which define control laws (e.g. inverse kinematics).

The most significant movemes contained in the training set are captured by the bases poses matrix **U** and encoded in the form of a displacement from the mean pose. Each column of **U** corresponds to a latent factor that describes some of the movement variability present in the data.

Figure 7.7 reports the motion described by three latent factors: the rows show the pose obtained by adding an increasing portion of the learned moveme (from 30% - second column, to 100% - last column) to the mean pose of the data (first column). Two are easily interpretable, "soccer kick" and "tennis forehand", while one is not as well defined, "volleyball strike / tennis serve". The movemes differentiate very quickly, with as little as 30% of the moveme added to the mean pose.

We verify empirically that two parameters mainly affect the correspondence between an action and a latent factor (moveme purity): the number of latent factors, and the constraints put on the coefficients of **V**. We obtain the best visualizations by approximately matching the number of latent factors with the number of recognizable actions contained in the dataset (10 for this experiment), and constraining the coefficients of V to be between 0 and 1.

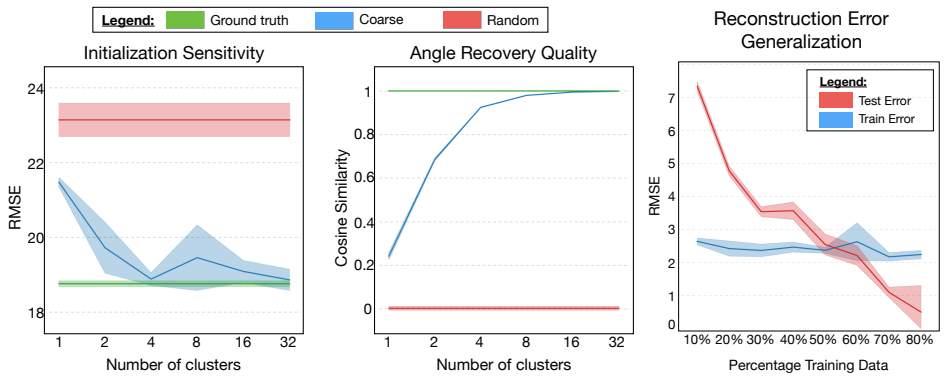


Figure 7.8: **Performance analysis on the viewpoint-angle estimation task and generalization analysis.** (a) Sensitivity with respect to the initial value of the angle of view of the training poses of (Left) the Root Mean Squared Error and (Right) the Cosine Similarity between the learned and ground-truth angles. A coarse initialization, within the correct quadrant of the true value, yields performances similar to ground truth. (b) The reconstruction error for poses not contained in the training set with respect to the percentage of data used in the training set. Full details in Section 7.4.5, and Section 7.4.6.

7.4.5 Angle Recovery

The "lfa3d" method learns a rotation invariant representation by treating the angle of view of each pose as a variable which is optimized through gradient descent (Section 7.3.3 and Figure 7.3), and requires an initial guess for each training instance. We investigate how sensitive is the model to initialization, and how close is the recovered angle of view to the ground truth. Figure 7.8-(a) shows the Root Mean Squared Error (RMSE) and cosine similarity with the ground truth, for three initialization methods: (1) "random", between 0 and 2π; (2) "coarse", coarsening into discrete buckets (e.g., 4 clusters indicates that we only know the viewing angle quadrant during initialization); and (3) "ground truth".

As the number of clusters increases, we see that performance remains constant for "random" and "ground truth", while both evaluation metrics improve significantly for "coarse" initialization. For instance, using just four clusters, "coarse" initialization obtains almost minimal RMSE and perfect cosine similarity. These results suggest that using very simple heuristics to predict the viewing angle quadrant of a pose is sufficient to obtain optimal performance.

7.4.6 Generalization Behaviour

A desirable property of the obtained model is to be able to reconstruct with low error poses that are not contained in the training set, so the representation is not tied uniquely to the specific image collection it was learned from. To verify the generalization quality of the learned bases poses, we trained the "lfa3d" model on a subset of the dataset and measured the RMSE on the remaining part, for an increasingly larger portion of the data. We repeated the experiment five times to obtain standard deviations.

As reported in Figure 7.8-(b), the RMSE over the training set is approximately constant, while the test set RMSE decreases significantly when going from 10% to 80% of the data used in training. This indicates that the learned latent factors can successfully reconstruct poses of unseen data.

7.4.7 Manifold Visualization

Figure 7.9 visualizes an embedding of the manifold of human motion learned with the "lfa3d" method. Each pose in LSP is mapped in the human motion space through the coefficients of the corresponding column of $\mathbf{V}$ and then projected in two-dimensions using t-SNE [37].

Poses describing similar movements are mapped to nearby positions and form consistent clusters, whose relative distance depends on which latent factors are used to reconstruct the contained poses. Upper body movements are mapped closely in the lower right corner, while lower body movements appear at the opposite end of the embedding. The mapping in the manifold is not affected by the direction each pose is facing, as nearby elements may have very different angle of view, confirming that the learned representation is rotation invariant.

In Figure 7.10, we show the heatmaps obtained from the activations of two latent factors from Figure 7.7, overlaid on top of the t-SNE mapping of Figure 7.9. To compute the heatmaps, we extract the coefficients for the "soccer kick" and "volleyball strike" latent factors from each column of $\mathbf{V}$ corresponding to a location in the embedding, and plot their value after normalization[2].

[2]To better depict the high-level trends, we enhance the contrast using a power of 1.5 and employ Gaussian smoothing.

Figure 7.9: **T-SNE visualization of the manifold of human poses for all the input images in the LSP dataset.** T-SNE embedding of the poses contained in the LSP dataset. Images, instead of poses, are shown for interpretability purpose. The type of body movement, and the influence of the learned bases poses determine the location in the manifold: "tennis serve" and "volleyball block" appear close in the manifold, while "running" is at the opposite end of the embedding. The angle of view does not affect the location in the manifold, as nearby poses may have very different angle of view. Full details in Section 7.4.7.

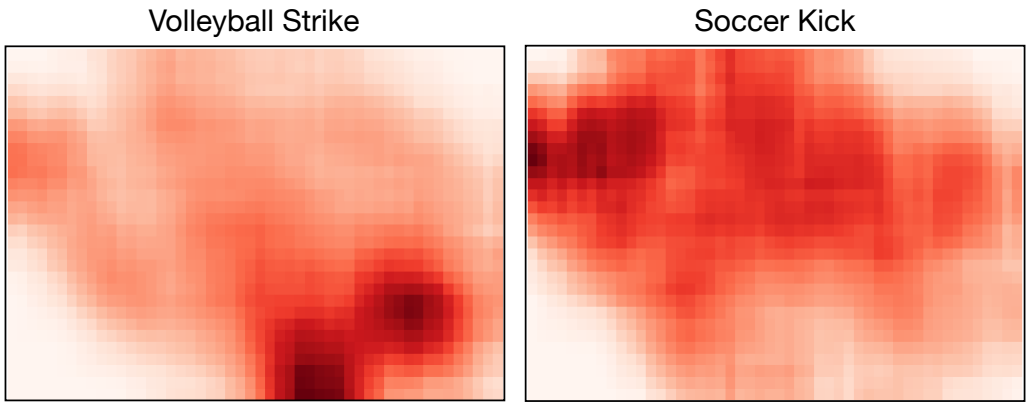


Figure 7.10: **Activation heatmap over all the LSP dataset images for two learned movemes.** Activation strength of the learned "volley strike" and "soccer kick" bases poses from Figure 7.7 (third and first row) in the t-SNE embedding. The heat-maps are consistent with Figure 7.9 in which movements of the upper and lower body are respectively mapped to the low-right and high-left corner.

Clearly, the epicentrum of the "volleyball strike" basis pose is located where volleyball-like poses appear in the t-SNE plot (lower-right corner). Noticeable upward arm movements are not as present in many other sports, hence the low intensity of the activation in the rest of the map. Conversely, the "soccer kick" basis pose is mostly dominant in the top-left area and the heatmap is diffused, consistent with the observation that most poses contain some movement of the legs.

7.5 Conclusion and Future Directions

In this work, we proposed a model for learning the primitive movements underlying human actions (movemes) from a set of static 2D poses obtained from images taken at various angles of view. The bases poses are rotation-invariant and learned through a modified latent matrix factorization that intrinsically accounts for geometric properties inherent to viewing angle variability. The approach can be trained efficiently, requires modest effort to identify a reasonable initialization, and yields very good generalization on unseen data.

We investigated the practical use of the learned representation for applications such as activity recognition and inference of action dynamics, observing significantly better performance compared to conventional baselines that do not account for variability of viewing angles. We used the bases poses for synthetic generation of movements, and explored how specific poses are mapped to different parts of the high-dimensional manifold of human motion.

One desirable property of our algorithm is that it is complementary to existing latent factor, pose estimation and feature extraction approaches, and may be used in combination with them to yield a better overall rotation-invariant representation.

An interesting future direction of investigation would be to use the proposed model in a semi-supervised setting where there is some availability of true three-dimensional data along with a large collection of two-dimensional joint locations.

Other possible extensions of our work are: learning to morph actions and synthesize *unseen* actions from the set of extracted movemes; inferring the location of occluded or missing joints based on the position of the visible ones; applying these techniques to large-scale datasets [38] in conjunction with fine grained annotations of the performed actions [9, 10] to gain new insights on the structure, complexity, and duration of human behaviour.

References

[1] E. S. Grood, S. F. Stowers, and F. R. Noyes, "Limits of movement in the human knee. effect of sectioning the posterior cruciate ligament and posterolateral structures.",
J Bone Joint Surg Am, vol. 70, no. 1, pp. 88–97, 1988 (cit. on p. 87).

[2] H. Hatze, "A three-dimensional multivariate model of passive human joint torques and articular boundaries",
Clinical Biomechanics, vol. 12, no. 2, pp. 128–135, 1997 (cit. on p. 87).

[3] D. J. Anderson and P. Perona, "Toward a science of computational ethology", *Neuron*, vol. 84, no. 1, pp. 18–31, 2014 (cit. on p. 88).

[4] C. Bregler,
"Learning and recognizing human dynamics in video sequences",
in *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, IEEE, 1997, pp. 568–574
(cit. on p. 88).

[5] R. Poppe, "A survey on vision-based human action recognition",
Image and vision computing, vol. 28, no. 6, pp. 976–990, 2010
(cit. on p. 88).

[6] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey", *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, no. 11, pp. 1473–1488, 2008
(cit. on p. 88).

[7] S. C. Hoyer, A. Eckart, A. Herrel, T. Zars, S. A. Fischer, S. L. Hardie, and M. Heisenberg, "Octopamine in male aggression of drosophila",
Current Biology, vol. 18, no. 3, pp. 159–167, 2008 (cit. on p. 88).

[8] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele,
"2d human pose estimation: New benchmark and state of the art analysis",
in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*,
Jun. 2014 (cit. on p. 88).

[9] M. R. Ronchi and P. Perona,
"Describing common human visual actions in images",
in *Proceedings of the British Machine Vision Conference (BMVC)*,
M. W. J. Xianghua Xie and G. K. L. Tam, Eds., BMVA Press, Sep. 2015,
pp. 52.1–52.12. DOI: 10.5244/C.29.52 (cit. on pp. 88, 108).

[10] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. Bernstein, and L. Fei-Fei,
"Visual genome: Connecting language and vision using crowdsourced dense image annotations", 2016.
[Online]. Available: http://arxiv.org/abs/1602.07332
(cit. on pp. 88, 108).

[11] A. Yao, J. Gall, G. Fanelli, and L. J. Van Gool,
"Does human action recognition benefit from pose estimation?.", in *BMVC*, vol. 3, 2011, p. 6 (cit. on p. 89).

[12] D. Fouhey and C. Zitnick, "Predicting object dynamics in scenes",
in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2019–2026 (cit. on p. 89).

[13] R. Parent, *Computer animation: Algorithms and techniques*. Newnes, 2012 (cit. on p. 89).

[14] V. Ramakrishna, T. Kanade, and Y. Sheikh,
"Reconstructing 3d human pose from 2d image landmarks",
in *Computer Vision–ECCV 2012*, Springer, 2012, pp. 573–586
(cit. on pp. 89, 90, 92).

[15] X. Fan, K. Zheng, Y. Zhou, and S. Wang,
"Pose locality constrained representation for 3d human pose reconstruction",
in *Computer Vision–ECCV 2014*, Springer, 2014, pp. 174–188
(cit. on pp. 89, 90, 99).

[16] I. Akhter and M. J. Black,
"Pose-conditioned joint angle limits for 3d human pose reconstruction",
in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1446–1455 (cit. on pp. 89, 90, 92, 98).

[17] X. Burgos-Artizzu, P. Perona, and P. Dollár,
"Robust face landmark estimation under occlusion",
in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1513–1520 (cit. on p. 90).

[18] X. Chen and A. L. Yuille, "Articulated pose estimation by a graphical model with image dependent pairwise relations",
in *Advances in Neural Information Processing Systems*, 2014,
pp. 1736–1744 (cit. on p. 90).

[19] S. Maji, L. Bourdev, and J. Malik, "Action recognition from a distributed representation of pose and appearance", in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, IEEE, 2011,
pp. 3177–3184 (cit. on p. 90).

[20] A. Eweiwi, M. S. Cheema, C. Bauckhage, and J. Gall,
"Efficient pose-based action recognition", in *Computer Vision–ACCV 2014*, Springer, 2014, pp. 428–443 (cit. on p. 90).

[21] B. Yao and L. Fei-Fei, "Modeling mutual context of object and human pose in human-object interaction activities", in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, IEEE, 2010, pp. 17–24
(cit. on p. 90).

[22] Z. Kourtzi and N. Kanwisher, "Activation in human mt/mst by static images with implied motion",
Journal of cognitive neuroscience, vol. 12, no. 1, pp. 48–55, 2000 (cit. on p. 90).

[23] T. Kim, G. Shakhnarovich, and R. Urtasun,
"Sparse coding for learning interpretable spatio-temporal primitives",
in *Advances in neural information processing systems*, 2010, pp. 1117–1125 (cit. on p. 90).

[24] M. Raptis and L. Sigal,
"Poselet key-framing: A model for human activity recognition",
in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2650–2657 (cit. on p. 90).

[25] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments", *PAMI*, 2014 (cit. on p. 90).

[26] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems", *Computer*, no. 8, pp. 30–37, 2009 (cit. on p. 91).

[27] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces",
in *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, IEEE, 1991, pp. 586–591 (cit. on pp. 91, 92).

[28] S. T. Dumais, "Latent semantic analysis", *Annual review of information science and technology*, vol. 38, no. 1, pp. 188–230, 2004 (cit. on p. 91).

[29] Y. Yue, P. Lucey, P. Carr, A. Bialkowski, and I. Matthews,
"Learning fine-grained spatial models for dynamic sports play prediction",
in *IEEE International Conference on Data Mining (ICDM)*, Dec. 2013 (cit. on p. 91).

[30] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 8, pp. 1798–1828, 2013 (cit. on p. 91).

[31] J. Mairal, F. Bach, J. Ponce, and G. Sapiro,
"Online dictionary learning for sparse coding", in *Proceedings of the 26th annual international conference on machine learning*, ACM, 2009, pp. 689–696 (cit. on p. 91).

[32] J. R. Gardner, M. J. Kusner, Y. Li, P. Upchurch, K. Q. Weinberger, and J. E. Hopcroft, "Deep manifold traversal: Changing labels with convolutional features", *ArXiv preprint arXiv:1511.06421*, 2015 (cit. on p. 91).

[33] S. Johnson and M. Everingham, "Clustered pose and nonlinear appearance models for human pose estimation",
in *Proceedings of the British Machine Vision Conference*,
doi:10.5244/C.24.12, 2010 (cit. on pp. 92, 99).

[34] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems",
SIAM journal on imaging sciences, vol. 2, no. 1, pp. 183–202, 2009
(cit. on p. 98).

[35] A. Krizhevsky, I. Sutskever, and G. E. Hinton,
"Imagenet classification with deep convolutional neural networks",
in *Advances in neural information processing systems*, 2012, pp. 1097–1105
(cit. on p. 101).

[36] P. P. de Groen, "An introduction to total least squares",
ArXiv preprint math/9805076, 1998 (cit. on p. 101).

[37] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne",
Journal of Machine Learning Research, vol. 9, no. 2579-2605, p. 85, 2008
(cit. on p. 105).

[38] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context",
in *Computer Vision–ECCV 2014*, Springer, 2014, pp. 740–755
(cit. on p. 108).

Chapter 8

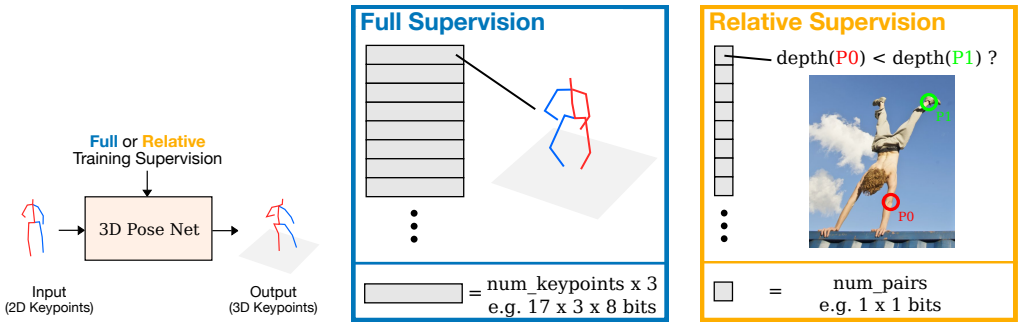
IT'S ALL RELATIVE: MONOCULAR 3D HUMAN POSE ESTIMATION FROM WEAKLY SUPERVISED DATA

The contents of this chapter are adapted from the peer-reviewed publication:

M. R. Ronchi, O. Mac Aodha, R. Eng and P. Perona *"It's all Relative: Monocular 3D Human Pose Estimation from Weakly Supervised Data."* 29th British Machine Vision Conference (2018, Newcastle, England)

URL: `http://www.vision.caltech.edu/~mronchi/projects/RelativePose`

We address the problem of 3D human pose estimation from 2D input images using only weakly supervised training data. Despite showing considerable success for 2D pose estimation, the application of supervised machine learning to 3D pose estimation in real world images is currently hampered by the lack of varied training images with corresponding 3D poses. Most existing 3D pose estimation algorithms train on data that has either been collected in carefully controlled studio settings or has been generated synthetically. Instead, we take a different approach, and propose a 3D human pose estimation algorithm that only requires relative estimates of depth at training time. Such training signal, although noisy, can be easily collected from crowd annotators, and is of sufficient quality for enabling successful training and evaluation of 3D pose algorithms. Our results are competitive with fully supervised regression based approaches on the Human3.6M dataset, despite using significantly weaker training data. Our proposed algorithm opens the door to using existing widespread 2D datasets for 3D pose estimation by allowing fine-tuning with noisy relative constraints, resulting in more accurate 3D poses.



(a) Lifting Methods (b) Full 3D Labels (c) Relative Labels

Figure 8.1: **Monocular 3D pose estimation from weak supervision.** (a) Lifting based methods for 3D human pose estimation take a set of 2D keypoints as input and predict their 3D position. (b) This is typically achieved using ground-truth 3D poses during training. (c) We show that weak supervision in the form of relative depth labels for as little as one pair of keypoints per image is effective for successfully training 3D pose estimation algorithms.

8.1 Introduction

Reasoning about the pose of humans in images and videos is a fundamental problem in computer vision and robotics. To ensure that future autonomous systems are safe to interact with, they need to be able to understand not only the positions, but also the poses of the people around them. Recent success in 2D pose estimation has been driven by larger, more varied, labeled datasets. While laborious, it is possible for human annotators to click on the 2D locations of different body parts to generate such training data. Unfortunately, in the case of 3D pose estimation, it is much more challenging to acquire large amounts of training data containing people in real world settings with their corresponding 3D poses. This lack of large scale training data makes it difficult to both train deep models for 3D pose estimation and to evaluate the performance of existing methods in situations where there are large variations in scene types and poses. While researchers have resorted to various alternative methods for collecting 3D pose training data - such as motion capture, synthetic datasets, video, and multi-camera setups - they are expensive, time consuming, and not easily scalable.

In this work, we argue that instead of using additional hardware to acquire full 3D ground-truth training data in controlled settings, Figure 8.1-(b), we can make use of human annotated relative depth information from images in the wild, Figure 8.1-(c), and train our method with large amounts of very inexpensive annotations (as little as one extra bit per image) on a large number of collections of images showing great variability in scene types and portrayed poses.

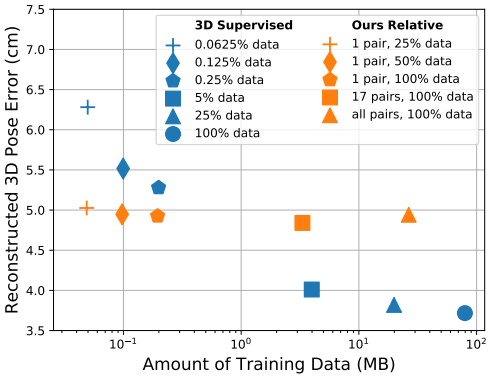


Figure 8.2: **Trade-off between training data and 3D pose reconstruction error.** As the amount of training data (measured in megabytes of annotations) is reduced, the performance of a fully supervised algorithm, Figure 8.1-(b), breaks down; on the other hand, our model, Figure 8.1-(c), predicts accurate 3D poses even with very small amounts of relative training data.

As a result, our method has a much more efficient trade-off between the amount of training annotations required for training and the pose reconstruction error, as shown in Figure 8.2.

The main contributions of this investigation are:

1. A loss for 3D pose estimation of articulated objects that can be trained on sparse and easy to collect relative depth annotations with performance comparable to the state-of-the-art;

2. an empirical evaluation of the ability of crowd annotators to provide relative depth supervision in the context of human poses;

3. a dataset of relative joint depth annotations that can be used for both training and evaluation purposes.

8.2 Related Work

2D Pose Estimation: Current state-of-the-art methods for 2D human keypoint estimation are based on deep networks [1–5] trained on large quantities of supervised data [6–9]. In addition to more sophisticated network architectures, a large driver in the improved accuracy of these approaches is the increase in the size and complexity [10] of datasets that contain images with corresponding keypoint annotations indicating the locations, in pixels, of specific body parts. While suitable for 2D pose estimation, by and large, most existing 2D datasets do not contain any supervision signal for 3D pose estimation. In this work, we show that existing 2D pose datasets can indeed be used for 3D pose estimation by augmenting them with relative depth annotations collected from crowd annotators.

3D Pose Estimation: There exist two main categories of methods for 3D pose estimation: (1) end-to-end models and (2) lifting based approaches. The first set of models take a 2D image or the output of a person detector as input, and then produce an estimate of the individual's 3D pose. This is achieved by learning to regress the 3D keypoint locations during training, either as a set of 3D coordinates [11] or as volumetric heat maps [12]. These methods assume the availability of a training set of 2D images paired with corresponding 3D annotations. To further constrain the problem, it is possible to enforce a strong prior on the predictions in the form of a parameterized model of human body shape [13–16]. While this ensures realistic looking outputs [17], it can be limiting if the prior is not flexible enough to cover the full range of valid poses.

As an alternative, lifting based approaches simply take a set of predicted 2D keypoints as input and *lift* them into 3D. The rise in popularity of these methods is driven by two factors: (1) the 2D location of keypoints is a strong cue indicating their 3D configuration and (2) the limited number of 'in the wild' datasets featuring *paired* 2D images with 3D poses. A variety of lifting approaches have been proposed that either frame the problem as one of regression [18, 19], data driven retrieval [20–22], dictionary based reconstruction [23], or use generative adversarial learning [24].

Instead of requiring full 3D pose information for each individual in an input image, we propose a method that only needs a small amount of sparse data indicating the relative depth of different body parts. This results in high quality predictions with as little as one relative depth constraint per pose at training time.

3D Pose Training Data: A major factor holding back progress in 3D pose estimation is the lack of *in the wild* datasets featuring images with ground-truth 3D poses. Most existing 3D pose datasets feature single individuals captured in controlled studio settings [25, 26] and are challenging to acquire due to the need for specialized equipment such as motion capture cameras and markers. Setups with multiple cameras make it easier to capture small numbers of interacting people [27, 28], but require multiple synchronized cameras in confined spaces to produce accurate depth. Depth cameras can be used to generate 3D training data [29], but are usually limited to the indoors. Alternative approaches use additional equipment such as inertial sensors [30, 31] or passive markers [32]. The main limitation of these setups is that it is very difficult to also capture 2D images that cover all the variation in appearance that one would encounter in real world, non-studio, settings.

One technique to extend studio captured motion capture data is to use computer graphics techniques to generate potentially unlimited amounts of synthetic training data [33, 34]. Synthetic training data has been successful for low-level vision tasks such as depth and optical flow estimation [35]. However, rendering realistic environments featuring people interacting with each other and their surroundings is a challenging problem. Furthermore, even if it is possible to successfully generate plausible scenes, these methods are still limited by the variation in pose and the range of subject interactions that are present in the input motion capture data. Different graphical user interfaces have been explored to allow crowd annotators to annotate 3D pose information in existing image datasets. Examples include, manually configuring 3D skeletons [36] or providing coarse body part orientation information [8, 37, 38]. These can be very laborious tasks, taking a large amount of time per image.

Relative Depth Supervision: Using ordinal relations between a sparse set of point pairs as a weak form of supervision has been previously explored in the context of dense depth estimation in monocular images [39, 40]. While this is not comparable to metric ground-truth depth, it enables the easy collection of data that can be used for both training and evaluating monocular depth estimation. This type of data has also been used for 3D pose estimation [41, 42]. However, unlike previous work that require complete annotations of all joint pairs to infer the 3D pose, we use only sparse relative annotations. Our annotations, Figure 8.1-(c), specify the relative distance between keypoints and the camera. Also [43] crowdsources sparse pairwise pose constraints. However, in their case the relative relationships are in the coordinate frame of the person in the image, and used as a replacement for 2D keypoints. We show that depth annotations relative to the camera are easy to collect and can be combined with existing 2D keypoint annotations for improving 3D pose estimation.

In parallel to our work, [44] also explored training 3D pose estimation models using relative (i.e. ordinal) depth constraints. They use multiple relative annotations per image to fine-tune 3D supervised trained models. In contrast, we use as little as one relative pair per image and focus on the setting where no camera intrinsics or 3D supervised ground-truth are available for pre-training or for calibrating the output of the model. Finally, we also conduct a detailed user study evaluating how accurate crowd annotators are at providing relative depth annotations for human poses.

8.3 Method

Our goal is to predict the 3D pose of an individual depicted in an input image. We represent pose in 2D as a set of coordinates $\mathbf{p} \in \mathbb{R}^{2\times J}$, where each element $\mathbf{p}_j = [u_j, v_j]$ is a row vector that encodes the location, in pixels, of one of J different joints. For each $\mathbf{p}$, we aim to infer its position in 3D $\mathbf{P} \in \mathbb{R}^{3\times J}$, where each entry specifies the location of the joint j in 3D, $\mathbf{P}_j = [x_j, y_j, z_j]$. In this work, we take inspiration from lifting based approaches, Figure 8.1-(a), and attempt to learn the parameters of a function $f : \mathbb{R}^{2\times J} \rightarrow \mathbb{R}^{3\times J}$, that maps 2D input keypoints to their estimated 3D position, where $f(\mathbf{p}) = \hat{\mathbf{P}}$. We parametrize f as a neural network, where the input joint positions $\mathbf{p}$ can come from the output of a 2D human pose estimation algorithm *e.g.* [2–4].

8.3.1 Supervised 3D Pose Estimation

Given a set of N input 2D keypoints and their corresponding ground-truth 3D pose, one could use a supervised loss to train f:

$$\mathcal{L}_{sup}(\hat{\mathbf{P}}, \mathbf{P}) = ||\hat{\mathbf{P}} - \mathbf{P}||_2. \tag{8.1}$$

This is the approach taken in [19], where a neural network, f, is trained to project the input coordinates $\mathbf{p}$ into 3D. While they only need to infer the missing z_j values for each 2D keypoint, their model predicts each $[x_j, y_j, z_j]$ coordinate, making the approach more robust to errors in the input 2D locations.

8.3.2 3D Pose Estimation with Relative Constraints

As noted earlier, acquiring large quantities of varied ground-truth 3D pose data is challenging. Instead, we opt to use much weaker supervision in the form of depth ordering labels that describe the relative distance to the camera for a pair of keypoints, see Figure 8.1-(c).

We assume we have access to a set of crowdsourced relative annotations for an image i, $\mathcal{A}^i = \{(j_1, k_1, r_1), (j_2, k_2, r_2), ..., (j_A, k_A, r_A)\}$, where each annotation (j, k, r) is a tuple specifying the joints j and k and their estimated relative depth $r \in \{-1, 1, 0\}$. The number of specified pairwise constraints, A, can be different for every image, and varies between one and $\binom{J}{2}$, when the ordinal supervision is provided for every pair of keypoints. The value $r = -1$ indicates that $z_j < z_k + \epsilon$ (joint j is closer to the camera compared to k), while $r = 1$ specifies that $z_k < z_j + \epsilon$. If the distance

between two keypoints is below a certain tolerance ϵ, then $r = 0$. In practice, this corresponds to the case in which human annotators cannot disambiguate the relative position of two keypoints. Unless otherwise noted, we explore the setting where $r \in \{-1, 1\}$.

Similar to [40], we use a pairwise ranking loss to encourage our model to predict the correct depth ordering of a 3D keypoint pair:

$$\mathcal{L}_{rel}(\hat{\mathbf{P}}, \mathcal{A}) = \sum_{(j,k,r)\in\mathcal{A}} \begin{cases} \log(1 + \exp(-r\hat{d}_{jk})), & r = -1, +1 \\ ||\hat{d}_{jk}||_2 & r = 0 \end{cases}, \tag{8.2}$$

where $\hat{d}_{jk} = \lambda(\hat{z}_j - \hat{z}_k)$, $\hat{z}_j$ being the predicted depth from our network for keypoint j, and λ controlling the strength of the loss. In practice, we found that it is important to normalize the range of depth values $\hat{d}$ to ensure numerical stability [45]. This is achieved by scaling by the mean absolute depth difference across each minibatch during training. We also constrain our 3D predictions so they are centered at a root joint that is encouraged to remain at the origin:

$$\mathcal{L}_{root}(\hat{\mathbf{P}}) = ||\hat{\mathbf{P}}_{root}||_2. \tag{8.3}$$

This controls the range of the output space, as the network does not have to model all possible poses at all possible distances from the camera.

The above ranking loss only encourages the relative distances to the camera to be respected for each keypoint pair, in essence constraining the z values. To force the correct location in both x and y in image space, we use a reprojection loss:

$$\mathcal{L}_{proj}(\hat{\mathbf{P}}, \mathbf{p}, \mathbf{v}, s) = \sum_j ||v_j(\Pi\hat{\mathbf{P}}_j^\top - \mathbf{p}_j)||_2, \tag{8.4}$$

where $v_j \in \{0, 1\}$ is a visibility flag and Π is a projection matrix. If no camera intrinsic information is available $\Pi = \left[\begin{smallmatrix} s & 0 & 0 \\ 0 & s & 0 \end{smallmatrix}\right]$, *i.e.* scaled orthographic projection, and in addition to predicting the 3D pose, our network also learns to predict the scaling parameter s for each input pose. If the ground-truth focal lengths are available during training, we can use perspective projection, and the reprojection loss becomes:

$$\mathcal{L}_{proj}(\tilde{\mathbf{P}}, \mathbf{p}, \mathbf{v}, s) = \sum_j ||v_j(\Pi\tilde{\mathbf{P}}_j^\top - \mathbf{p}_j)||_2, \tag{8.5}$$

where $\Pi = \left[\begin{smallmatrix} f_x & 0 & 0 \\ 0 & f_y & 0 \end{smallmatrix}\right]$, and $\tilde{\mathbf{P}}_j = \left[x_j/(z_j + s), y_j/(z_j + s), 1\right]$. Now the network's scaling parameter s has a different interpretation and is used to predict the distance from the camera to the center of the person in 3D.

Even with the above terms, 3D pose estimation from 2D inputs is heavily underconstrained as many different 3D pose configurations can respect both the relative depth constrains and the reprojection loss. To further constrain the problem, we include one additional geometric loss that enforces weak prior knowledge related to the ratio between the lengths of the different limbs. We assume that we are given an input skeleton $\mathcal{B} = \{(b_1^1, b_1^2, l_1), (b_2^1, b_2^2, l_2), ..., (b_B^1, b_B^2, l_B)\}$, consisting of B 'bones' (*i.e.* limbs), where each entry (b^1, b^2, l) specifies the indices of the keypoint pair that are the endpoints for that particular limb, and its length l. The limb length loss then measures the difference in length between the predicted limb and the predefined reference length,

$$\mathcal{L}_{skel}(\hat{\mathbf{P}}) = \sum_{(b^1,b^2,l)\in\mathcal{B}} ||\text{len}(\hat{\mathbf{P}}, b^1, b^2) - l||_2, \tag{8.6}$$

where $\text{len}(\hat{\mathbf{P}}, j, k) = ||\hat{\mathbf{P}}_j - \hat{\mathbf{P}}_k||_2$. In practice, we do not minimize the difference between the absolute bone lengths, but instead normalize the predicted and reference bones by fixing one of the limbs to be unit length, in effect constraining their ratios as in [46, 47]. The skeleton loss also implicitly enforces symmetry between the sets of left and right limbs.

Our final loss $\mathcal{L}$ is the combination of the above four terms with additional weighting hyperparameters to control the influence of each component

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_{root}(\hat{\mathbf{P}}^i) + \alpha\mathcal{L}_{rel}(\hat{\mathbf{P}}^i, \mathcal{A}^i) + \beta\mathcal{L}_{proj}(\hat{\mathbf{P}}^i, \mathbf{p}^i, \mathbf{v}^i, s^i) + \gamma\mathcal{L}_{skel}(\hat{\mathbf{P}}^i). \tag{8.7}$$

8.3.3 Implementation Details

We use the same fully connected network architecture presented in [19]. For all our experiments, we adopted the one stage version of the model (Figure 8.3) since it showed only a minor loss in performance compared the two-stage version, but provided a significant speed-up.

To predict the scale parameter s used in our reprojection loss, we add an additional fully connected layer to the output of the penultimate set of layers and apply a sigmoid non-linearity to its output. The output of the non-linearity is scaled using a hyperparameter r to allow the network to predict an arbitrarily wide range. For the default method 'Ours Relative', detailed in Table 8.1, we set $r = 1$. In the relative depth loss, we set $\lambda = 2.5$. We set the weighting hyperparameters α and γ in the

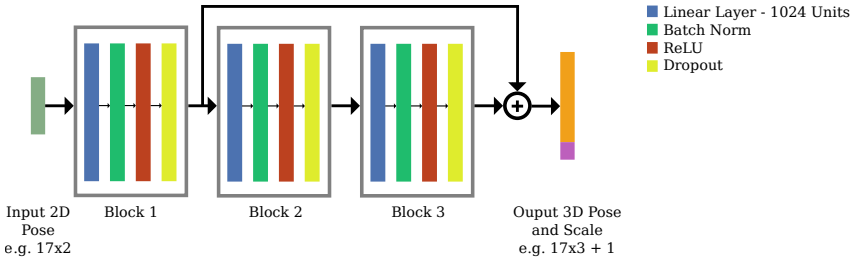


Figure 8.3: **Neural network architecture adopted for 3D pose regression.** Our architecture is inspired by [19], but includes an additional scaling factor prediction at the output layer of the network which we use in the introduced reprojection loss, Eq. 8.4 and Eq. 8.5.

main loss to 1.0 and set β to 0.1. Both fully supervised and relative models are trained on Human3.6M for a fixed number of 25 epochs, as they do not tend to benefit from additional training time. When using LSP, we instead train our relative models from scratch for 100 epochs.

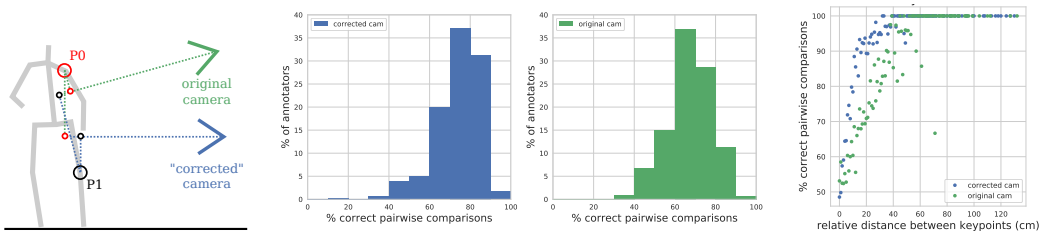
The input 2D keypoints for our relative models are always centered by setting the root location to (0,0). We did not perform this centering for the supervised baseline as we found that it hurt performance, but we did center the 3D output coordinates in a similar fashion.

As in [19], we clip the gradients to 1.0 during training. Training time on Human3.6M is less than five minutes for one epoch for our relative model.

8.4 Human Relative Depth Annotation Performance

Our model for 3D pose estimation makes use of relative depth annotations at training time. In order to use this type of supervision, it is important to understand how accurately can humans provide these labels. This is in contrast to the study carried out in [48], which investigates the ability of humans to observe and physically reenact a target 3D pose. To measure the quality of relative joint annotations collected via a crowd sourcing platform, we performed an evaluation using participants recruited on Mechanical Turk with 1,000 images randomly selected from the Human3.6M dataset [26], as it features ground-truth depth.

For each annotation task, the crowd annotators were presented with an image along with two randomly selected keypoints and were instructed to imagine themselves looking through the camera and report which of the two keypoints appeared closer to them. We forced annotators to choose from one of the two possibilities and did not provide a 'same distance' option for ambiguous situations, as those cases can be inferred by inspecting the disagreement between annotators.



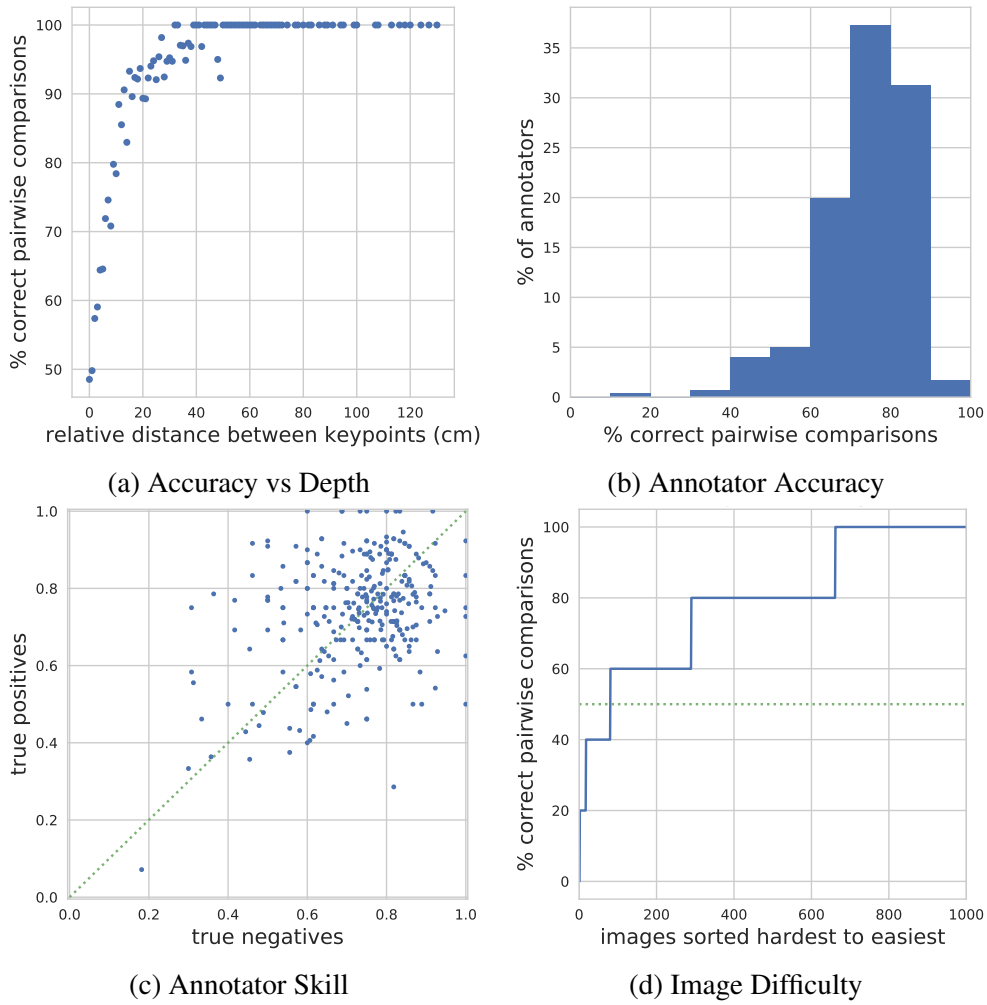
(a) Camera Correction (b) Annotator Accuracy (c) Accuracy vs Depth

Figure 8.4: **Bias in the annotations due to camera perspective.** (a) The green camera represents the input view and the blue is the upright orientated view as perceived by our annotators. If the camera is orientated upwards when evaluating annotators' performance, the provided labels are a better match with the true relative depths measured with the ground truth. (b-c) With/without correcting for the orientation of the camera (blue/green lines), annotators perform better/worse.

For each of the 1,000 images, we collected five random pairs of keypoints, ensuring that five different annotators labeled the same keypoints and image combination. In total, this resulted in 25,000 individual annotations collected from 301 annotators, with an average of 83 annotations each. We merged each of the five votes per keypoint pair using the crowd annotation system of [49], resulting in a single predicted label per pair. We found this to perform slightly better than a majority vote, with the added benefit of providing a probabilistic label.

We observed a bias in the annotations due to the fact that crowd annotators tend to not factor in the forward lean of the camera in Human3.6M when making their predictions. In Figure 8.4-(a), we see an illustration of this effect. Here, from the perspective of the original camera view in green the keypoint 'P0' is closer than 'P1'. In practice, even though annotators see an image of the scene taken from the perspective of the green camera, they seemingly correct for the orientation of the camera and 'imagine' the distance of the scene from the perspective of the blue camera. While this change in camera position is subtle, it affects the relative ordering of the points as 'P1' is now closer to the camera. We hypothesize that this is a result of the annotator imagining themselves in the same pose as the individual in the image, and then estimating the distance to the camera in a Manhattan world sense. Without correcting for this effect, 67% of the provided pairwise annotations are correct, but when this is taken into account then accuracy increases to 71%.

We correct for the bias by forcing the camera to be upright when computing the scene depth. The results before and after applying this correction and annotator accuracies can be viewed in Figure 8.4-(b-c). Note that fixing this bias requires



(a) Accuracy vs Depth

(b) Annotator Accuracy

(c) Annotator Skill

(d) Image Difficulty

Figure 8.5: **Human performance analysis on the relative keypoint depth annotation task.** We measured human performance on the task of relative keypoints depth annotation on 1,000 random images from the Human3.6M dataset [26]: (a) accuracy versus distance between a pair of keypoints; (b) histogram of all the annotator's accuracy; (c) comparison of the annotator's skill at identifying true positives versus true negatives; (d) sorting of the images based on their overall difficulty measured as accuracy obtained by the annotators.

ground-truth depth data, and is performed here to evaluate annotators' performance. In real world scenarios, our model would learn this annotation bias. This effect is likely to be exacerbated in Human3.6M as there are only four different camera viewpoints in the entire dataset, and they are all facing downwards. We expect this to be less of an issue for datasets that feature a larger variation in camera viewpoints relative to the subject of interest as the dominant ground plane will have less of a biasing effect.

In Figure 8.5, we quantify the accuracy of the crowd annotations, shown after correcting for the annotation bias. In Figure 8.5-(a), we see that for keypoint pairs

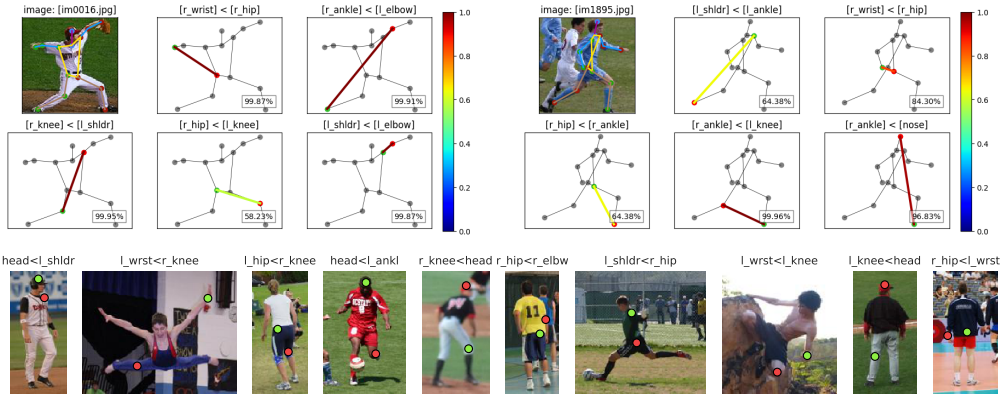


Figure 8.6: **Example of the relative keypoint depth annotations collected for the LSP dataset.** (Top) The ground-truth 2D skeleton and five randomly selected pairs of keypoints with the consolidated relative depth supervision and confidence based on annotator's agreement. (Bottom) The names and relative order of the pair of keypoints are written on top of every image, with the green keypoint annotated to be closer to the camera compared to the red one after consolidating all the annotators' answers. Images are sorted from most to least confident based on the annotator's agreement. The two right-most examples illustrate challenging cases where the keypoints are at a similar distance to the camera.

that are separated by more than 20 cm, our merged predictions are correct over 90% of the time, where random guessing is 50%. While only a small number of annotators annotated over 90% of the pairs correctly, Figure 8.5-(b), the vast majority tend to perform better than random guessing. In Figure 8.5-(c), we observe that the rate of true positives versus true negatives for every annotator is fairly symmetric, indicating that annotators are equally good at providing the correct answer independently of a keypoint being in front or behind another one. Some image and keypoint combinations are more challenging than others, and in Figure 8.5-(d), we sort the images from hardest to easiest based on the percentage of keypoint pairs that are correctly annotated. For over two thirds of the images, four out of the five pairs are correctly annotated. Importantly, the cases where annotators have trouble predicting the correct keypoint order, by and large, tend to be ambiguous pairs where the actual ground-truth distances are small. These results indicate that human annotators can indeed provide high quality weak 3D pose information in the form of relative depth annotations that can be used as supervision for our algorithm.

Using the same protocol described above, we collected annotations for all 2,000 images in the Leeds Sports Pose (LSP) dataset [6]. LSP features a much larger variation in camera viewpoint and pose compared to Human3.6M. We selected five random keypoint pairs per image, with five repeats, for a total of 50,000 annotations.

Annotations were performed by 348 annotators who provided an average of 144 labels each. Example annotations, after merging the five responses using [49], can be seen in Figure 8.6. Unlike Human3.6M, there is no ground-truth depth data available for LSP, so to evaluate the quality of the crowd annotations, two of the authors independently annotated the same subset of 500 keypoint pairs. Agreement between the two was 84%, where the majority of disagreements occurred in ambiguous cases. For the set of pairs where the two annotators agreed, the merged crowd annotations were the same 90.2% of the time. These results are consistent with the performance on Human3.6M, despite the larger variation in poses and viewpoints.

8.5 3D Pose Estimation Results

We use a similar model to [19] for all our experiments and report the 3D pose estimation performance of our model on the Human3.6M [26] and LSP [6] datasets in the following sections. A description of our network architecture and implementation details are available in Section 8.3.3.

8.5.1 Human3.6M Dataset

As noted in [21], many different evaluation protocols have been defined for Human3.6M, making it very challenging to comprehensively compare to all existing methods. We opt to use protocol #2 from the state-of-the-art [19], as it is the model most similar to ours, and has been used by several recent approaches. Here, training is performed on subjects [1, 5, 6, 7, 8] and the test set consists of all frames and cameras for subjects [9, 11]. Some methods sub-sample the test data from the original 50fps to 10fps, but we observed little difference to the test scores and thus opted to preserve the original frame rate of the dataset. As in [19], Procrustes alignment is performed at test time between each prediction and its corresponding test example. With the exception of the results in Table 8.2, where we average across actions, everywhere else we report results by averaging across all frames. Finally, unless specified, we use 17 2D keypoints as input and predict their corresponding 3D locations, with the relative annotations derived from the ground-truth depth.

We explored different configurations of our model at training time, by varying the type of reprojection loss adopted, the proportions of the input skeleton, the amount of training data, and the distance tolerance.

Model					**3D Pose Error (mm)**	
	Reprojection Type	**Skeleton**	**Amount Train**	**Distance Tolerance**	**Scale**	**Procrustes**
1) 3D Supervised	-	-	-	-	49.62	36.54
2) **Ours Relative**	Scaled Orthographic	H36 Avg	1 Pair	No	64.58	48.97
3) Known focal length	Perspective	H36 Avg	1 Pair	No	65.22	48.54
4) Generic skeleton	Scaled Orthographic	Avg from [41]	1 Pair	No	73.26	55.27
5) No skeleton loss	Scaled Orthographic	H36 Avg	1 Pair	No	118.34	89.81
6) Amount train	Scaled Orthographic	H36 Avg	All Pairs (136)	No	63.96	48.90
7) Distance tolerance	Scaled Orthographic	H36 Avg	1 Pair	Yes (100mm)	63.45	47.54

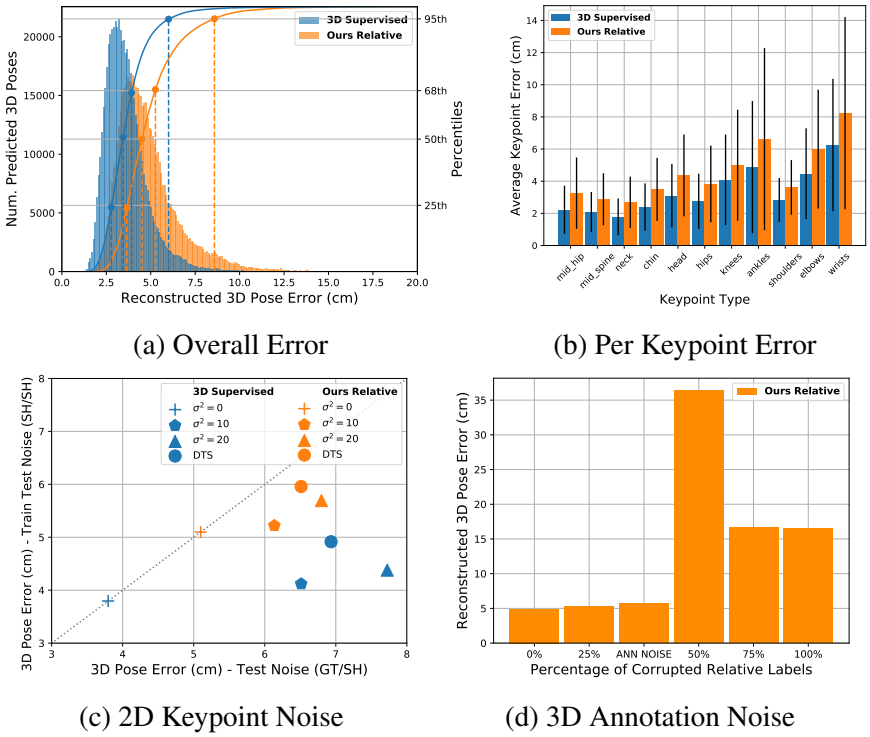
Table 8.1: **Ablation study of our model on the Human3.6M dataset.** The last two columns show the performance at test time when an optimal single parameter re-scaling (*Scale*) or full rigid alignment (*Procrustes*) is performed for each predicted pose based on the ground truth. '3D Supervised' is our re-implementation of [19]. 'Ours Relative' shows the default settings adopted for our model, while rows 3-7 contain different variants.

We describe below our most interesting findings from the results in Table 8.1:

- *Reprojection*: The availability of ground-truth focal lengths at training time, Eqn. 8.5, yields comparable performance to using the scaled orthographic projection of Eqn. 8.4.
- *Skeleton*: Using less accurate limb length ratios in Eqn. 8.6, obtained from [41] instead of Human3.6M's training set average, slightly hurts the accuracy, while entirely removing the skeleton loss significantly reduces performance.
- *Amount Train*: Increasing the amount of relative depth pairs does not significantly alter the results. However, this effect is likely due to the high redundancy present in Human3.6M. This is consistent with the findings of Figure 8.1-(d), where we report the error against the percentage of training images used.
- *Distance Tolerance*: Setting the depth tolerance to 100 mm helps performance. This is because using $r = 0$ in Eqn. 8.2 forces the network to constrain a pair of predicted keypoints to be at the same depth, as opposed to just determining their relative order.

Some of these configurations are more realistic than others when using crowd provided annotations on an 'in the wild' image dataset. We denote 'Ours Relative' as the model with the most realistic assumptions and use it for the remaining experiments.

Figure 8.7 summarizes the overall performance and robustness to noise of our model.



(a) Overall Error

(b) Per Keypoint Error

(c) 2D Keypoint Noise

(d) 3D Annotation Noise

Figure 8.7: **Performance analysis on the 3D pose estimation task.** We compare the performance of our algorithm and one using full 3D supervision. (a) Histogram of the errors on all the images in the Human3.6M dataset test set (lines and dots respectively show the cumulative sum and percentiles). (b) A breakdown of errors over individual keypoints. (c) Comparison of the robustness to perturbations of the 2D input keypoint locations. (d) Analysis of our method's robustness to errors in the relative depth labels during training.

In Figure 8.7-(a), we show a histogram of the pose errors on the test set both for our method and [19]. The mode and median of the two curves are 10mm from each other. However, our method suffers from more catastrophic errors, as can be seen in the longer tail. This is due to the fact that even when respecting all the relative depth labels we do not fully constrain their absolute depth. This can also be seen in the breakdown of the error over keypoint type in Figure 8.7-(b). As one might expect, body extremities such as ankles and wrists show a larger error (and deviation).

Figure 8.7-(c) shows the degradation in performance for the cases in which the 2D input keypoints are obtained by adding a Gaussian noise $\mathcal{N}(0, \sigma^2)$ with increasingly high variance (up to 20) to the 2D ground-truth keypoints or by using the outputs of a keypoint detector [3] fined-tuned on the Human3.6M training set, taken from [19]. The performance of [19] is better when the *train* and *test* data have the same amount of noise degradation (lower error along the y-axis), while our method performs best when noise is only added at *test* time (lower error along the x-axis).

	Direct.	Discuss	Eating	Greet	Phone	Photo	Pose	Purch.	Sitting	SitingD	Smoke	Wait	WalkD	Walk	WalkT	Avg
Sanzari et al. [50]	48.8	56.3	96.0	84.8	96.5	105.6	66.3	107.4	116.9	129.6	97.8	65.9	130.5	92.9	102.2	93.2
Rogez et al. [51]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	71.6
Kanazawa et al. [17]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	58.1
Pavlakos et al. [12]	47.5	50.5	48.3	49.3	50.7	55.2	46.1	48.0	61.1	78.1	51.1	48.3	52.9	41.5	46.4	51.9
Tekin et al. [52]	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	50.1
Fang et al. [53]	38.2	41.7	43.7	44.9	48.5	55.3	40.2	38.2	54.5	64.4	47.2	44.3	47.3	36.7	41.7	45.7
Hossain et al. [54] (T)	36.9	37.9	42.8	40.3	46.8	46.7	37.7	36.5	48.9	52.6	45.6	39.6	43.5	35.2	38.5	42.0
Pavlakos et al. [44] (E)	34.7	39.8	41.8	38.6	42.5	47.5	38.0	36.6	50.7	56.8	42.6	39.6	43.9	32.1	36.5	41.8
Martinez et al. [19] 17j GT/GT	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	37.1
Martinez et al. [19] 17j SH/SH	39.5	43.2	46.4	47.0	51.0	56.0	41.4	40.6	56.5	69.4	49.2	45.0	49.5	38.0	43.1	47.7
3D Supervised 17j GT/GT	31.1	37.0	34.3	36.3	37.2	42.5	36.6	33.8	39.9	49.3	37.0	37.7	38.8	33.1	37.5	37.5
Ours Relative 17j GT/GT	43.6	45.3	45.8	50.9	46.6	55.3	43.3	47.3	56.6	74.3	47.1	48.5	52.1	48.5	49.8	50.3
Ours Relative 17j GT/GT (N)	46.4	47.3	49.7	53.3	52.3	59.2	43.8	50.9	80.5	107.9	54.3	51.6	53.1	52.4	53.3	57.1
3D Supervised 16j SH/SH	40.2	44.4	46.6	48.3	53.3	56.7	41.7	42.4	55.9	68.1	50.8	46.2	51.8	40.4	45.5	48.8
Ours Relative 16j SH/SH	51.1	50.8	56.4	60.5	60.6	64.7	48.9	52.1	67.5	89.0	59.1	55.8	61.3	59.2	59.0	59.7
Ours Relative 16j SH/SH (N)	54.0	54.5	70.4	64.0	78.3	71.6	50.4	58.8	113.9	131.0	76.7	60.1	64.8	61.8	63.7	71.6

Table 8.2: **3D pose reconstruction errors on the Human3.6M dataset.** We use evaluation protocol #2, defined in Section 8.5.1. GT and SH are the ground-truth and fine-tuned (from [19]) detected 2D input keypoints respectively, with 'j' indicating the number of joints used for testing. (T) represents the use of temporal information, (E) extra training data, and (N) the use of simulated annotation noise in the relative depth pairs. Errors are reported in mm.

We hypothesize that this behavior is due to the presence of the reprojection and skeleton losses at training time, which encourages our method to find plausible 3D poses that respect the input poses, making it more robust to slight changes in the distribution of the input keypoints.

In Figure 8.7-(d), we demonstrate that our model is also robust to noise in the relative depth labels during *training*. Performance is mostly unchanged when up to 25% of the labels are randomly flipped. The third bar corresponds to the amount of noise obtained from simulated crowd annotators, regressed from Figure 8.5-(a). This is of interest, as it shows performance with noise comparable to what we would expect to collect in the wild. The worst performance is obtained when the labels are randomly flipped, and improves for cases in which the amount of noise is larger than 50%, as the model is able to exploit structure that is still present in the data, but produces poses that are flipped back to front.

Finally, in Table 8.2 we compare our model to existing fully 3D supervised approaches. Even with significantly less training data, and without any architecture exploration, we still perform competitively.

Overall, our results show that, when available, 3D ground-truth is a very powerful training signal, but using our proposed training methodology relative depth data can instead be used at the expense of little accuracy at test time. Furthermore, our model is robust to using noisy predicted 2D keypoints at test time, again with a minor decrease in performance.

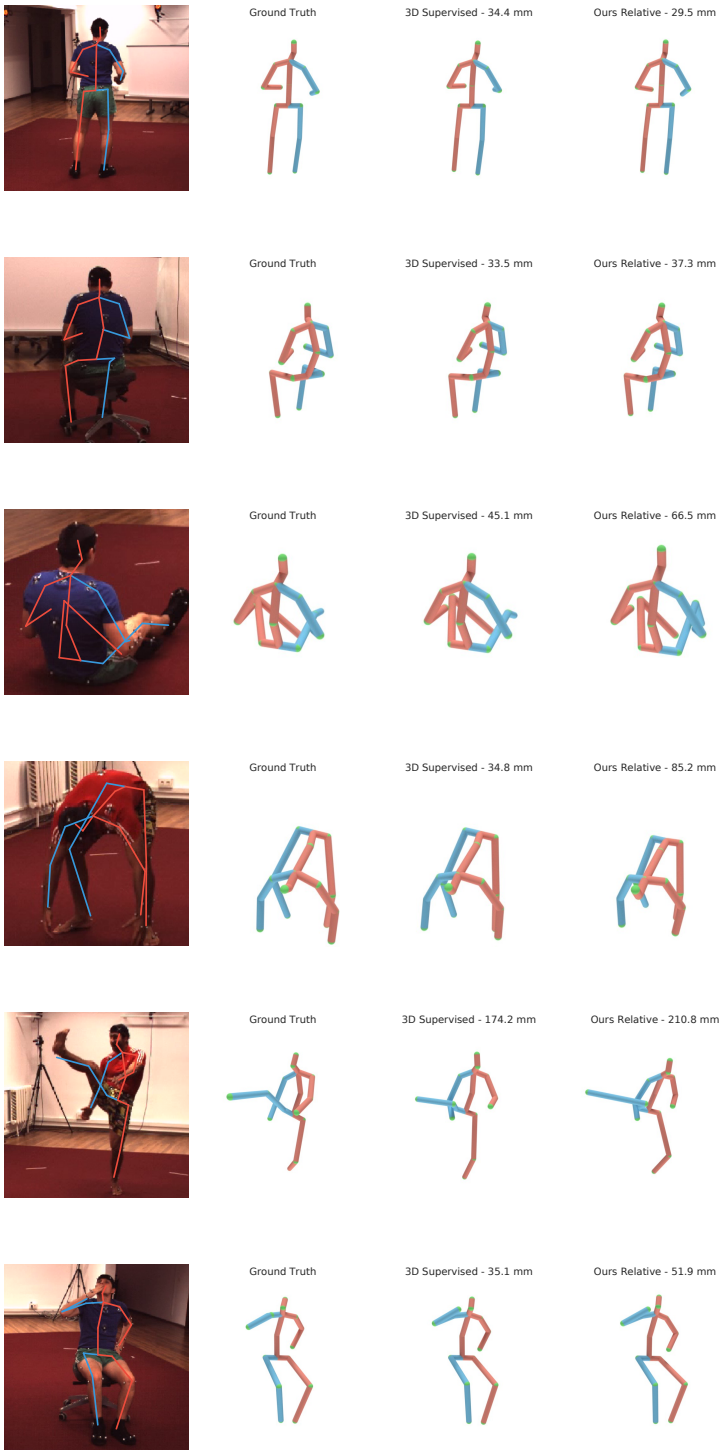


Figure 8.8: **Inference time 3D pose predictions on the Human3.6M dataset.** Despite using much weaker training data, our relative model (Ours Relative 17j GT/GT) produces sensible results for most input poses. Both the supervised and our approach are depicted after rigid alignment, with the obtained error (in mm) displayed on top.

Example 3D predicted poses on input images from the Human3.6M dataset can be seen in Figure 8.8. Our algorithm is very competitive compared to the fully supervised baseline, even for images that show more complicated human poses.

8.5.2 Leeds Sports Pose Dataset

We perform experiments on the LSP dataset which, unlike Human3.6M, does not contain ground-truth 3D pose annotations. As a result, we evaluate the models by measuring the percentage of relative labels incorrectly predicted compared to the merged crowd annotations collected via Mechanical Turk.

The 3D supervised model [19] and our relative model, with one comparison per input pose, trained on Human3.6M achieve test errors of 34.4% and 34.3% respectively. We are able to further reduce the error of our relative model to 24.1% by fine-tuning it using the ordinal annotations collected for the LSP training set. Furthermore, training our relative model from scratch using exclusively the LSP training set also outperforms the supervised baseline obtaining an error of 27.1%. This shows that weak training data is sufficient for competitive performance on this benchmark, and we can successfully make use of noisy annotations to improve the prediction of 3D poses if 3D ground truth is available.

Example outputs of the above methods on the LSP dataset are displayed in Figure 8.9. These results show that our method significantly improves the quality of the predictions compared to the fully supervised baseline. This highlights the importance of fine-tuning on hard images containing uncommon poses and rare viewpoints that are not found in the Human3.6M dataset.

8.6 Conclusion

We presented a weakly supervised approach for 3D human pose estimation. We showed that sparse constraints that indicate the relative depth of pairs of keypoints can be used as a training signal, resulting in competitive results at a fraction of the amount of training data. Unlike most approaches that require ground-truth 3D poses, our method can be applied to legacy image collections, as only the input 2D keypoints and relative depth annotations are required. This opens the door to using existing datasets for 3D pose estimation in the wild.

Large scale annotation is time consuming and expensive, even when only collecting weak supervision. In future, we plan to investigate efficient, active learning based,

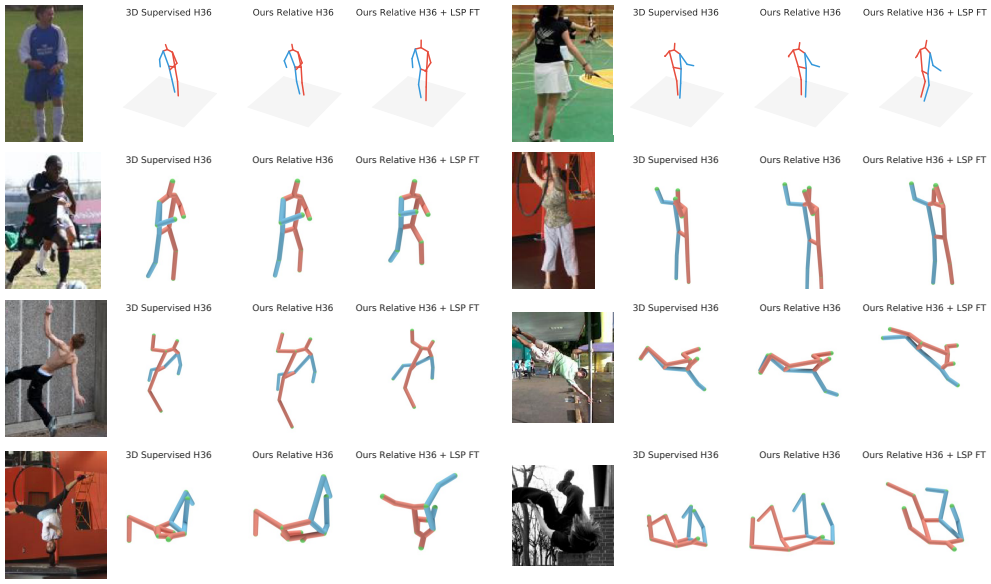


Figure 8.9: **Inference time 3D pose predictions on the LSP dataset.** Fine-tuning (FT) on LSP significantly improves the quality of our predictions especially for images containing uncommon poses and viewpoints that are not found in Human3.6M, such as those visualized in the last row.

approaches for collecting annotations [55]. Current state-of-the-art 2D pose estimation algorithms perform best on single humans in isolation and their performance deteriorates when there are large numbers of occluded keypoints and closely interacting people [10]. Including weak 3D information for multiple interacting individuals may help resolve some of these ambiguities.

References

[1] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler, "Joint training of a convolutional network and a graphical model for human pose estimation", in *NIPS*, 2014 (cit. on p. 115).

[2] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines", in *CVPR*, 2016 (cit. on pp. 115, 118).

[3] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation", *ArXiv preprint arXiv:1603.06937*, 2016 (cit. on pp. 115, 118, 127).

[4] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields", in *CVPR*, 2017 (cit. on pp. 115, 118).

[5] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn", in *ICCV*, 2017 (cit. on p. 115).

[6] S. Johnson and M. Everingham, "Clustered pose and nonlinear appearance models for human pose estimation", in *Proceedings of the British Machine Vision Conference*, doi:10.5244/C.24.12, 2010 (cit. on pp. 115, 124, 125).

[7] B. Sapp and B. Taskar, "Modec: Multimodal decomposable models for human pose estimation", in *CVPR*, 2013 (cit. on p. 115).

[8] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2d human pose estimation: New benchmark and state of the art analysis", in *CVPR*, 2014 (cit. on pp. 115, 117).

[9] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context", in *Computer Vision–ECCV 2014*, Springer, 2014, pp. 740–755 (cit. on p. 115).

[10] M. R. Ronchi and P. Perona, "Benchmarking and error diagnosis in multi-instance pose estimation", in *ICCV*, 2017 (cit. on pp. 115, 131).

[11] S. Li and A. B. Chan, "3d human pose estimation from monocular images with deep convolutional neural network", in *ACCV*, 2014 (cit. on p. 116).

[12] G. Pavlakos, X. Zhou, K. G. Derpanis, and K. Daniilidis, "Coarse-to-fine volumetric prediction for single-image 3d human pose", in *CVPR*, 2017 (cit. on pp. 116, 128).

[13] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, "Scape: Shape completion and animation of people", in *ACM Transactions on Graphics (TOG)*, 2005 (cit. on p. 116).

[14] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model", *ACM transactions on graphics (TOG)*, 2015 (cit. on p. 116).

[15] B. Tekin, I. Katircioglu, M. Salzmann, V. Lepetit, and P. Fua, "Structured prediction of 3d human pose with deep neural networks", in *BMVC*, 2016 (cit. on p. 116).

[16] D. Tome, C. Russell, and L. Agapito, "Lifting from the deep: Convolutional 3d pose estimation from a single image", *CVPR*, 2017 (cit. on p. 116).

[17] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose", in *CVPR*, 2018 (cit. on pp. 116, 128).

[18] F. Moreno-Noguer, "3d human pose estimation from a single image via distance matrix regression", in *CVPR*, 2017 (cit. on p. 116).

[19] J. Martinez, R. Hossain, J. Romero, and J. J. Little, "A simple yet effective baseline for 3d human pose estimation", in *ICCV*, 2017 (cit. on pp. 116, 118, 120, 121, 125, 126, 127, 128, 130).

[20] H. Yasin, U. Iqbal, B. Kruger, A. Weber, and J. Gall, "A dual-source approach for 3d pose estimation from a single image", in *CVPR*, 2016 (cit. on p. 116).

[21] C.-H. Chen and D. Ramanan, "3d human pose estimation= 2d pose estimation+ matching", in *CVPR*, 2017 (cit. on pp. 116, 125).

[22] G. Rogez, P. Weinzaepfel, and C. Schmid, "Lcr-net++: Multi-person 2d and 3d pose detection in natural images", *ArXiv:1803.00455*, 2018 (cit. on p. 116).

[23] I. Akhter and M. J. Black, "Pose-conditioned joint angle limits for 3d human pose reconstruction", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1446–1455 (cit. on p. 116).

[24] W. Yang, W. Ouyang, X. Wang, J. Ren, H. Li, and X. Wang, "3d human pose estimation in the wild by adversarial learning", in *CVPR*, 2018 (cit. on p. 116).

[25] L. Sigal, A. O. Balan, and M. J. Black, "Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion", *IJCV*, 2010 (cit. on p. 116).

[26] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments", *PAMI*, 2014 (cit. on pp. 116, 121, 123, 125).

[27] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh, "Panoptic studio: A massively multiview system for social motion capture", in *ICCV*, 2015 (cit. on p. 116).

[28] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko, W. Xu, and C. Theobalt, "Monocular 3d human pose estimation in the wild using improved cnn supervision", in *3DV*, 2017 (cit. on p. 116).

[29] T. Shu, M. S. Ryoo, and S.-C. Zhu, "Learning social affordance for human-robot interaction", *IJCAI*, 2016 (cit. on p. 116).

[30] G. Pons-Moll, A. Baak, J. Gall, L. Leal-Taixe, M. Mueller, H.-P. Seidel, and B. Rosenhahn, "Outdoor human motion capture using inverse kinematics and von mises-fisher sampling", in *ICCV*, 2011 (cit. on p. 116).

[31] M. Trumble, A. Gilbert, C. Malleson, A. Hilton, and J. Collomosse, "Total capture: 3d human pose estimation fusing video and inertial sensors", in *BMVC*, 2017 (cit. on p. 116).

[32] R. Wang, S. Paris, and J. Popović, "Practical color-based motion capture", in *SIGGRAPH/Eurographics Symposium on Computer Animation*, 2011 (cit. on p. 116).

[33] W. Chen, H. Wang, Y. Li, H. Su, Z. Wang, C. Tu, D. Lischinski, D. Cohen-Or, and B. Chen, "Synthesizing training images for boosting human 3d pose estimation", in *3DV*, 2016 (cit. on p. 117).

[34] G. Varol, J. Romero, X. Martin, N. Mahmood, M. J. Black, I. Laptev, and C. Schmid, "Learning from synthetic humans", in *CVPR*, 2017 (cit. on p. 117).

[35] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation", in *CVPR*, 2016 (cit. on p. 117).

[36] Q. Nguyen and M. Kipp, "Annotation of human gesture using 3d skeleton controls.", in *LREC*, 2010 (cit. on p. 117).

[37] S. Maji, "Large scale image annotations on amazon mechanical turk", *EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2011-79*, 2011 (cit. on p. 117).

[38] S. Maji, L. Bourdev, and J. Malik, "Action recognition from a distributed representation of pose and appearance", in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, IEEE, 2011, pp. 3177–3184 (cit. on p. 117).

[39] D. Zoran, P. Isola, D. Krishnan, and W. T. Freeman, "Learning ordinal relationships for mid-level vision", in *ICCV*, 2015 (cit. on p. 117).

[40] W. Chen, Z. Fu, D. Yang, and J. Deng, "Single-image depth perception in the wild", in *NIPS*, 2016 (cit. on pp. 117, 119).

[41] C. J. Taylor, "Reconstruction of articulated objects from point correspondences in a single uncalibrated image", in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, IEEE, vol. 1, 2000, pp. 677–684 (cit. on pp. 117, 126).

[42] L. Bourdev and J. Malik, "Poselets: Body part detectors trained using 3d human pose annotations", in *Computer Vision, 2009 IEEE 12th International Conference on*, IEEE, 2009, pp. 1365–1372 (cit. on p. 117).

[43] G. Pons-Moll, D. J. Fleet, and B. Rosenhahn, "Posebits for monocular human pose estimation", in *CVPR*, 2014 (cit. on p. 117).

[44] G. Pavlakos, X. Zhou, and K. Daniilidis, "Ordinal depth supervision for 3d human pose estimation", *CVPR*, 2018 (cit. on pp. 117, 128).

[45] C. Wang, J. M. Buenaposada, R. Zhu, and S. Lucey, "Learning depth from monocular videos using direct methods", in *CVPR*, 2018 (cit. on p. 119).

[46] C. Wang, Y. Wang, Z. Lin, A. L. Yuille, and W. Gao, "Robust estimation of 3d human poses from a single image", in *CVPR*, 2014 (cit. on p. 120).

[47] X. Zhou, Q. Huang, X. Sun, X. Xue, and Y. Wei, "Towards 3d human pose estimation in the wild: A weakly-supervised approach", in *ICCV*, 2017 (cit. on p. 120).

[48] E. Marinoiu, D. Papava, and C. Sminchisescu, "Pictorial human spaces: A computational study on the human perception of 3d articulated poses", *IJCV*, 2016 (cit. on p. 121).

[49] S. Branson, G. Van Horn, and P. Perona, "Lean crowdsourcing: Combining humans and machines in an online system", in *CVPR*, 2017 (cit. on pp. 122, 125).

[50] M. Sanzari, V. Ntouskos, and F. Pirri, "Bayesian image based 3d pose estimation", in *ECCV*, 2016 (cit. on p. 128).

[51] G. Rogez, P. Weinzaepfel, and C. Schmid,
"LCR-Net: Localization-Classification-Regression for Human Pose",
in *CVPR*, 2017 (cit. on p. 128).

[52] B. Tekin, P. Marquez Neila, M. Salzmann, and P. Fua, "Learning to fuse 2d and 3d image cues for monocular body pose estimation", in *ICCV*, 2017 (cit. on p. 128).

[53] H. Fang, Y. Xu, W. Wang, X. Liu, and S.-C. Zhu, "Learning pose grammar to encode human body configuration for 3d pose estimation", in *AAAI*, 2018 (cit. on p. 128).

[54] M. R. I. Hossain and J. J. Little, "Exploiting temporal information for 3d pose estimation", *ArXiv:1711.08585*, 2017 (cit. on p. 128).

[55] B. Liu and V. Ferrari, "Active learning for human pose estimation",
in *ICCV*, 2017 (cit. on p. 131).

Chapter 9

HOW MUCH DOES MULTI-VIEW SELF-SUPERVISION HELP 3D POSE ESTIMATION?

The contents of this chapter are adapted from the manuscript:

M. R. Ronchi, O. Mac Aodha and P. Perona *"How Much Does Multi-View Self-Supervision Help 3D Pose Estimation?"*

URL: `http://www.vision.caltech.edu/~mronchi/projects/MultiView3DPose`

3D Pose estimation from a single image is challenging due to both the inherent ambiguity of the task, and the difficulty of collecting large and varied supervised training datasets. Self-supervised learning has emerged as an alternative solution where the aim is to learn features that encode pose information without requiring explicit supervision. This feature learning step is typically framed as solving a pretext task that is related to pose, such as temporal alignment of video frames or multi-view reconstruction. However, directly comparing current approaches is difficult due to different datasets and experimental settings used. In this work, we standardize this comparison by performing a detailed evaluation of multi-view self-supervised feature learning methods. Through experiments on Human3.6M, we observe several important issues that arise when using multi-view information as a training signal, including the impact of the amount of supervised pose data, the 3D pose reference frame used, the power of different pose decoders, among others. We conclude with recommendations and by highlighting open questions.

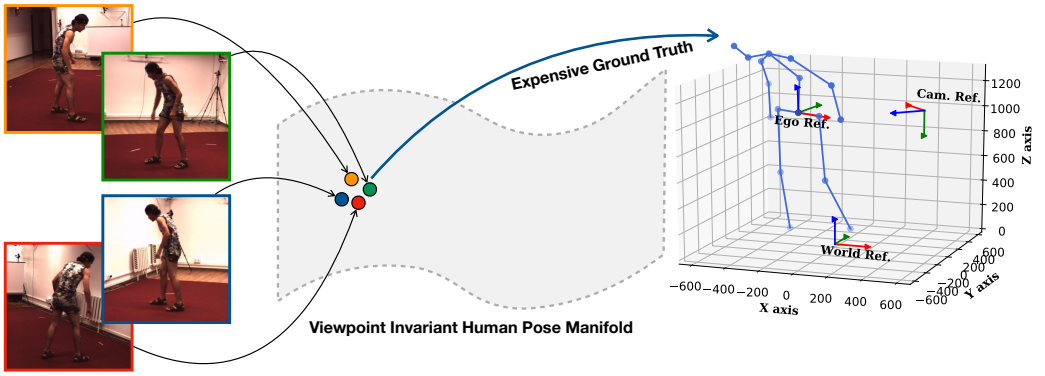


Figure 9.1: **Multi-view self-supervision for the task of 3D human pose estimation.** We wish to learn an embedding function that embeds images of the same pose, taken from different camera viewpoints, to the same place in feature space without requiring any supervised pose data (left). With a small amount of expensive to collect 3D pose supervision, we can train a pose decoder that projects from the embedding space to 3D pose space (right). Depending on the application of interest, human pose can be represented in a camera centered reference frame (*camera-centric*) or in an intrinsic reference frame (*ego-centric*).

9.1 Introduction

The ability to perform robust 3D human pose estimation opens up the possibility of richer interactions between humans and autonomous systems. By accurately predicting human pose, these future systems will be able to safely collaborate with humans in close proximity to each other.

Unfortunately, while innovations in network architectures, losses, and larger supervised training datasets have significantly improved the performance of 2D pose estimation [1–5], 3D pose estimation from a single image still remains a challenging problem.

Traditional approaches to computing 3D human pose from monocular two-dimenaional images and video relied on hand crafting models of the human body and designing ad-hoc detectors and trackers [6–9]. However, recent progress in deep networks has kindled interest in utilizing purely learning-based approaches [10–12].

A shortcoming of these methods is that they are data hungry, and annotating images with 3D pose ground-truth is both tedious and expensive, and especially hard to do for large varied training sets of *in-the-wild* 3D human poses. We are thus interested in exploring methods to recover 3D pose that require the least amount of supervision, both at training and inference time.

To address this lack of training data, several self-supervised learning approaches have been proposed that aim to learn features that encode information about human pose without requiring expensive to obtain ground-truth data. The quality of these features can then be evaluated by measuring how well they predict pose given a limited amount of supervised finetuning data.

Existing self-supervised methods exploit different forms of structure that is present in the raw image data to learn informative features. This can include: temporal information in the context of video [13, 14], the relationship between 2D and 3D pose [15], and multi-view images [16, 17]. These methods can be framed as distance metric learning problems, where the goal is to learn an image embedding space in which pairs of images containing similar 3D poses are closer in the learned space, while pairs that contain different poses are further apart.

In contrast to other forms of self-supervision, one of the main advantages of time-synchronized multi-view approaches [16, 17] is that they explicitly have information, at training time, about a subset of frames that are known to contain the same poses: images from the same individual captured at the same time point from different cameras. However, during training they throw away information about camera viewpoint as they force the learned embedding to encode the same pose from different viewing angles in the same location in the feature space. This is acceptable when the goal is to reason about pose in an *ego-centric* reference frame, but it poses challenges when one wants to extract pose with respect to the input camera, *camera-centric* reference frame. We highlight the difference between the two representations of pose in Figure 9.1.

In addition, while using videos recorded from multiple viewpoints provides information about positive image pair correspondeces for free, choosing negative pair frames is still a non-trivial task and worse, there is a danger that similar poses from different time points will be incorrectly selected as negative pairs [13, 17, 18].

For some applications, such as action recognition and image retrieval, representing pose with an intrinsic representation might suffice. However, 3D pose estimation from monocular images and robotic control are the principal applications we have in mind, and they both benefit from a camera-aware representation of pose. Thus, one of our goals is to study the ability of recovering either representation of pose from the learned viewpoint-invariant embedding.

In this study, we explore issues related to training self-supervised multi-view embedding based approaches for single view 3D human pose estimation. Through extensive experiments on Human3.6M [19], we show that there are several important design decisions that can significantly impact the accuracy of 3D pose estimation, among which some are the choice of reference frame used for evaluation, the capacity of the pose decoder, and the selection of training samples. By measuring the effect of each these aspects, and others, we make recommendations for how to best close the gap between self-supervised and fully supervised training.

9.2 Related Work

Supervised 3D Pose Estimation

The goal of 2D human pose estimation is to predict the 2D locations of a pre-defined set of keypoints on the human body in the image space. In the case of 3D pose estimation, we also want to predict the 'depth' of each keypoint (either with respect to the camera or in a specified 3D coordinate space). This is an inherently ill-posed problem when only given a single image as input. The existing literature for single image 3D pose estimation can be broadly divided into two main approaches: end-to-end, and two-stage (*i.e.*'lifting').

End-to-end approaches take a 2D image as input and directly regress the 3D keypoints from the pixel data. Common representations for the predicted pose include 3D coordinates [10, 20], volumetric predictions [21, 22], coefficients for probabilistic pose models [23], and 3D meshes for full body shape and pose encoding [24, 25]. Semi-supervised end-to-end training also enables the use of additional 2D keypoint data that has no paired 3D information [26, 27]. Multi-view information has also been shown to improve semi-supervised training [28, 29].

For lifting approaches, the goal is to take a set of 2D keypoints as input and then predict the missing depth dimension. These methods do not train a full deep feature extractor from the input image but instead use compact fully connected models to regress the missing dimensions [30]. In addition to direct regression of missing keypoints other lifting approaches have explored keypoint distance matrix estimation [31], direct retrieval [12, 32], and adversarial learning [33, 34].

We explore self-supervised learning of 3D pose aware feature representations using distance metric learning. Once trained, our feature extraction networks can be combined with any representation of 3D pose during supervised finetuning.

Training Data for 3D Pose Estimation

While 2D pose estimation has benefited from large supervised training datasets *e.g.*[35, 36], acquiring ground-truth 3D pose data is much more difficult. Standard options include capturing data in controlled settings with motion capture markers [19, 37], multi-camera setups [38, 39], or using paired depth cameras [40]. However, all these approaches require additional hardware and calibration and are difficult to deploy in outdoor settings. Training data can also be generating synthetically, resulting in ground-truth dense depth for free [41–43]. However, it is challenging to generate varied 3D poses with realistic scene appearance and interactions without significant manual intervention. As full 3D pose information for 'in-the-wild' image collections is very challenging to acquire. One proposed solution is to crowd-source weak 3D information in the form of estimated relative keypoint distances from the camera [44–47]. The disadvantage of this approach is that it can result in noisy labels as in many cases the relative distance of some pairs of keypoints can be hard to disambiguate.

Our goal is to learn features that encode 3D pose information without requiring any direct 3D supervision. Once trained, we show that these features can be finetuned with a relatively small amount of 3D supervision, resulting in accurate 3D pose predictions.

Self-Supervised 3D Pose Estimation

To overcome the lack of training data for 3D pose estimation, there is a growing interest in self-supervised approaches for learning features that encode 3D pose. These methods can be categorized based on their requirements for additional data at training time or the assumptions they make about how the images are related.

Given no 2D or 3D pose information at training time, but instead a set of time synchronized multi-view images with known camera extrinsics, [16] proposed an image reconstruction based approach to learn a features for 3D pose estimation. As they have the known transformations between the cameras during training the learned latent space encodes camera viewpoint information as well as pose. [48] proposed a similar approach but assumed they had access to 2D pose during training and performed the reconstruction in 2D pose space, rather than in image space. In contrast, other embedding based approaches that also assume availability of time synchronized cameras are unable to recover camera viewpoint information as

they explicitly train embeddings that are invariant to viewpoint [17, 49]. [50] still assumed access to time synchronized views from neighboring viewpoints but used a pretrained pose estimation network, along with multi-view epipolar constraints, to learn how to predict 3D pose. This overcomes the requirement of having known camera extrinsics.

Another source of self-supervision which has been exploited in the context of 2D pose estimation is temporal information. [18] trained a Siamese architecture to predict if two frames were temporally close or far, with the assumption that nearby frames in time are more similar in pose to further away ones. [13, 14] also utilized time information with the assumption that the same action performed by two different people will have a similar temporal ordering. These models learn to align pairs of video of the same action by enforcing consistency in both directions for the predicted frame-level embeddings. While they still require multiple videos of the same action, unlike the previous multi-view approaches, these videos do not have to be time synchronized or depict the same individual.

The last main category of self-supervision is having access to 2D keypoint information from non-time synchronized image collections. [15] proposed a lifting based approach that used a series of self-consistency constraints in both 2D and 3D along with an adversarial loss to encourage that projected 3D posses are valid 2D poses. However, these lifting approaches do not explicitly train an image encoder that can be used for any arbitrary image.

One of the main challenges for many of the above embedding based approaches for self-supervised pose representation learning is the sampling of positive and negative pairs of frames during training. Positive pairs of frames are usually selected based on some criteria such as the same time-point imaged from a different viewpoint or two frames that are close in time. However, due to the repeated nature of poses over time, selecting negative pairs of frames (*i.e.*ideally those containing very different poses) is challenging. One solution is to filter out potential negative pairs based on their similarity in the embedding space during training [13, 17, 18].

In this work we explore design decision related to multi-view self-supervised learning for single view 3D pose estimation. We evaluate the impact of each of the decisions on the quality of the learned features and measure how much they effect the predicted 3D poses.

9.3 Method

The proposed pipeline for semi-supervised monocular 3D pose estimation is composed of three modular components, depicted in Figure 9.2:

1. **Pose encoder:** We use a Time Contrastive Network [49] to learn a viewpoint-invariant representation of human poses, Section 9.3.1, using un-edited and un-calibrated videos from Human3.6M [19].

2. **Pose decoder:** We use a fully connected architecture [51] to learn the mapping from vectors in the learned embedding to the output space of human poses (ego-centric or camera-centric), Section 9.3.2.

3. **Viewpoint decoder:** We use a neural network architecture combining convolutional and fully connected layers to learn the reference frame orientation of the predicted poses, Section 9.3.3.

The entire pipeline is trained in two successive phases: *i)* we train the encoder in a completely self-supervised way on the pretext task of time-contrastive metric learning, and *ii)* we jointly fine-tune the pose and viewpoint decoders, using any small amount of 3D ground-truth pose supervision available.

Compared to related work in the same area of research, our algorithm does not require any camera intrinsic or extrinsic parameters during the self-supervised training phase, as opposed to [16], or the multi-view videos to be pre-segmented in order to avoid cyclic movements of the portrayed subjects like [14, 49]. This is a fundamental improvement, which makes the pipeline truly self-supervised and amenable to using with completely uncurated multi-view video sequences.

At inference time, our pipeline can be used for monocular 3D pose estimation in images or for video prediction by using it in a frame-by-frame fashion. Depending on the application of interest, it is possible to predict poses both in ego-centric coordinates, which discard the viewpoint information, or in camera-centric coordinates. This can be useful, since we observed that the data versus accuracy trade-off for the two pose representations is different, see Figure 9.3. Thanks to its design, our method requires very little supervision in terms of 3D pose ground-truth, and no supervision at all in terms of the camera intrinsic or extrinsic parameters for the baseline model.

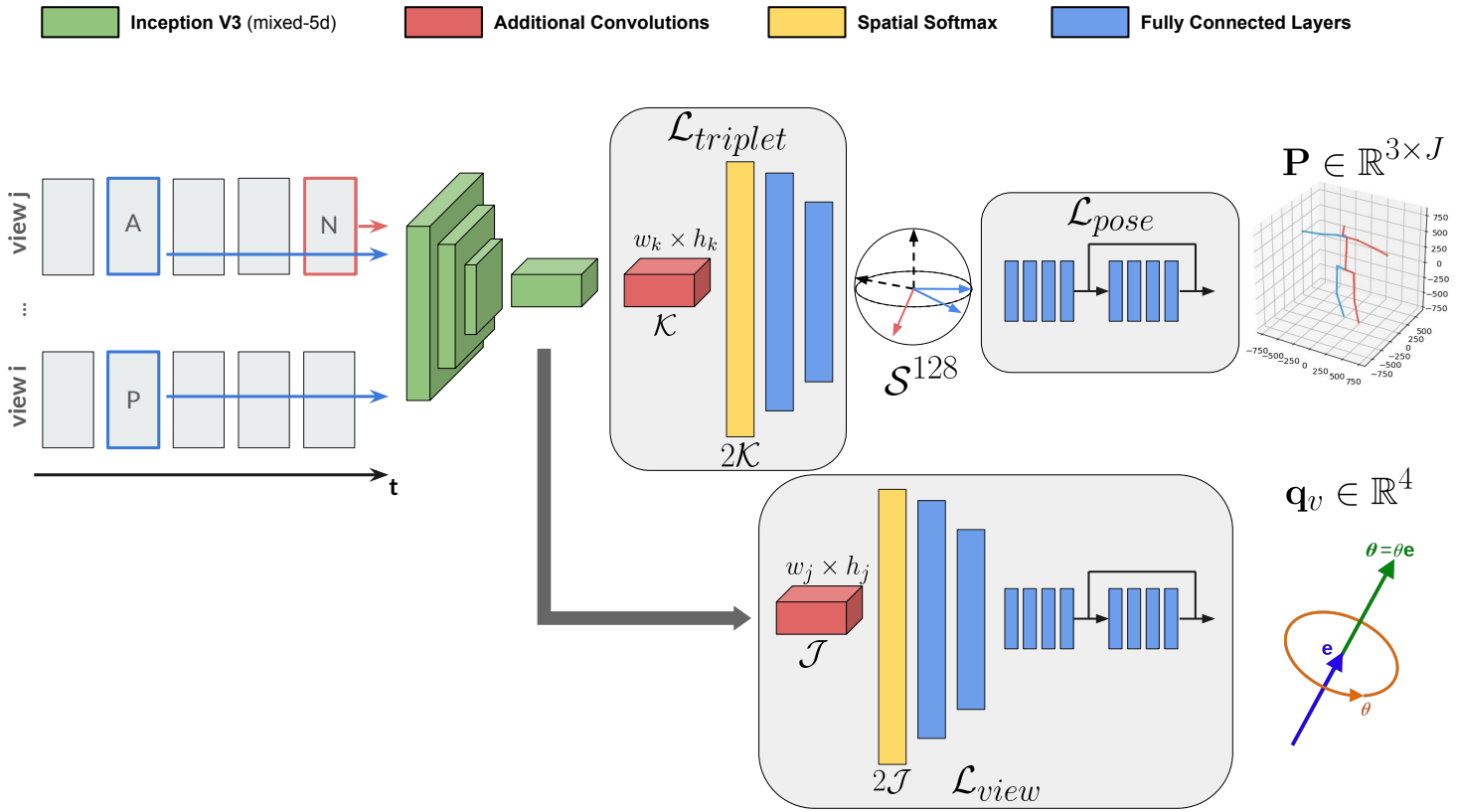


Figure 9.2: **Encoder-decoder pipeline for monocular 3D human pose estimation.** The proposed pipeline for semi-supervised monocular 3D pose estimation is based on three components: *i)* a Time Contrastive Network encoder which embeds an image from a video into a viewpoint-invariant manifold (Section 9.3.1); *ii)* a pose and *iii)* a viewpoint decoders (Section 9.3.2 and Section 9.3.3), which output, respectively, the 3D poses in ego-centric or camera-centric coordinates, and the rotation needed to transform one reference frame into the other, Eq. 9.2. The encoder is trained using multi-view videos from the Human3.6M dataset [19], while the decoders are trained using individual frames with ground-truth pose.

9.3.1 Time Contrastive Network Encoder

The goal of the encoder portion of our pipeline is to extract features from the input images and embed them in a lower dimensional viewpoint-invariant space in which all the nuisance factors related to viewpoint have been discarded, while the intrinsic content of an image is preserved. Since our input images portray human subjects performing a sequence of actions, we hope that the retained information can be useful for estimating 3D human pose.

The input data consists of 360×360 pixel images $\mathbf{I}$ extracted from the videos of the Human3.6M dataset [19]. The training set contains 150 videos obtained from two repetitions of 15 actions by 5 subjects recorded from four different viewpoints, while the test set contains two subjects for a total of 60 videos. Following the practice of related work [16, 26], we down-sample all the videos in the dataset from 50Hz to 10Hz and further sub-sample the test set by selecting one pose every 200th frame, resulting in a final size of 312,188 images for the training set and 2,844 images for the test set.

The network is parametrized as a function $h : \mathbb{R}^{360\times360} \rightarrow \mathbb{R}^{128}$ which outputs an embedding vector $\mathbf{e} = h(\mathbf{I})$. The architecture, inspired by [49] with some slight modifications, is composed of a convolutional backbone taken from an Inception Network V3 [52] up to the internal layer "mixed-5d", trained on ImageNet [53] for the image classification task, to which we add two sets of 512 convolutional filters of spatial resolution 35×35, followed by a Spatial Softmax Layer [54] and a set of fully connected layers that reduce the final dimension of our embedding to 128. The outputs of the last convolutional layer are L2 normalized, meaning that our embedding space is the unit sphere in $\mathbb{R}^{128}$.

We train the encoder in a time-contrastive fashion [49] using a triplet loss [55, 56] formulation that requires that two frames taken at the same t_i timestamp, but different viewpoints v_n and v_m be encoded in the embedding space at a closer euclidean distance than two frames from the same viewpoint v_n, but at different moments of time, t_i and t_j:

$$\mathcal{L}_{triplet} = \begin{cases} ||h(\mathbf{I}_{t_i}^{v_n}) - h(\mathbf{I}_{t_i}^{v_m})||_2^2 + \alpha < ||h(\mathbf{I}_{t_j}^{v_n}) - h(\mathbf{I}_{t_i}^{v_n})||_2^2 \\ \forall t_i, t_j, v_n, v_m \end{cases} , \tag{9.1}$$

where $\alpha > 0$ is a real number representing an arbitrary margin enforced between positive and negative pairs.

The necessary assumption for the training algorithm to be successful is that two frames taken at the same timestamp are visually dissimilar, but semantically similar, while two frames far away in time are visually similar (as they come from the same viewpoint), but semantically dissimilar.

The implication is that the way in which these triplets are sampled is fundamental. For every video (and its available viewpoints), we select a short time window centered around a random frame, and randomly sample a certain number of frames within that window, called *anchor frames*. For every anchor frame, we use the corresponding frames from the other viewpoints as positives, and for each anchor-positive pair a semi-hard negative is selected by choosing from all the available negatives (frames from the same viewpoint as the anchor) the one that is closest to the positive in the embedding space [57]. The window within which random frames are sampled is very important as it impacts the ability to pick high quality semi-hard negatives.

9.3.2 Pose Decoder

We represent the ego-centric pose of a person, Figure 9.1, with a vector $\mathbf{P}_{ego} \in \mathbb{R}^{3\times J}$, where each entry specifies the 3D location of the joint j, $\mathbf{P}^{(j)}_{ego} = [x_j, y_j, z_j]$ with respect to an intrinsic reference system in which the x-axis is aligned with the pelvis girdle and the y-axis is aligned to the spine. The camera-centric pose of a person portrayed in an image is instead represented with a vector $\mathbf{P}_{cam} \in \mathbb{R}^{3\times J}$. However, $\mathbf{P}_{cam}$ is defined with respect to the coordinate reference system centered in the camera that took the picture, also visible in Figure 9.1.

The goal of the pose decoder is to map vectors $\mathbf{e}$ from the viewpoint-invariant embedding space to the output space of 3D human poses. We implement it as a fully connected neural network parametrizing the function $f : \mathbb{R}^{128} \rightarrow \mathbb{R}^{3\times J}$, such that $f(\mathbf{e}) = \mathbf{P}$, (ego-centric or cam-centric). This decoder is non-convolutional, since all the image processing has been done by the Time Contrastive Network encoder and is trained using a simple eucledian loss, between the predicted and ground-truth 3D pose, $\mathcal{L}_{pose}(\hat{\mathbf{P}}, \mathbf{P}) = ||\hat{\mathbf{P}} - \mathbf{P}||_2$, supervised using individual frames from the Human3.6M dataset [19]. In Figure 9.8 and Figure 9.9, we analyze the impact of using decoders of different complexity.

If permitted by the application of interest, using ego-centric pose coordinates is preferable, as it requires a smaller amount of expensive ground truth, as exemplified in Figure 9.1. We investigate the implication of using different pose representations with respect to the reconstruction error in Figure 9.3 and Figure 9.4.

9.3.3 Viewpoint Decoder

If the application of interest requires a camera-centric representation, the pose decoder's task is much more complex. In fact, it has to jointly decode both the pose and camera viewpoint from the embedding space, which has been trained to throw away viewpoint specific information. In Section 9.4, we look in depth at the pose decoder's ability to perform such task across several experiments.

To alleviate the above problem, we introduce a branch in our pipeline that is used for disentangling the prediction of the camera-centric pose into a separate ego-centric component, predicted by the pose decoder, and a rigid reference system rotation, predicted by the viewpoint decoder. Not only this facilitates the task of the pose decoder, but also it allows the viewpoint decoder to leverage image appearance features from a lower level of the architecture which might contain useful information about viewpoint that would have otherwise been discarded by the encoder network in the training procedure described in Section 9.3.1.

We define the viewpoint rotation matrix $\mathbf{R}_v \in \mathbb{R}^{3\times3}$, as the rigid coordinate system rotation that brings one pose representation into the other:

$$\mathbf{P}_{cam} = \mathbf{P}_{ego} \cdot \mathbf{R}_v. \tag{9.2}$$

A visualization of the different reference frames is shown in Figure 9.1.

The goal of the viewpoint decoder is thus, given an input image, to predict the viewpoint orientation matrix $\mathbf{R}_v$ that transforms the ego-centric pose prediction into its alternative camera-centric representation.

Estimating (explicitly or implicitly) the viewpoint orientation matrix with a neural network is a very complicated task, due to the fact that regression lacks the closed geometry property of the $SO(3)$ space of 3D rotations in $\mathbb{R}^3$. A multitude of recent approaches have provided suggestions on how using different representations of rotation impacts the performance of the regression task [58–60].

Inspired by [58, 61], we adopted a representation of rotation based on quaternions. This is particularly useful because *i)* unit quaternions are one to one with $SO(3)$ and *ii)* the vector part of a quaternion can be interpreted as a coordinate vector, therefore the algebraic operations of the quaternions reflect the geometry of $\mathbb{R}^3$.

As a result, we represent the viewpoint orientation with a vector $\mathbf{q}_v = [w, x, y, z] \in \mathbb{R}^4$ such that $\mathbf{q}_v = w + x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$, where $\mathbf{i}$, $\mathbf{j}$ and $\mathbf{k}$ are such that $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1$, and $||\mathbf{q}_v||_2 = 1$.

We implement the viewpoint decoder as a neural network approximating the function $g : \mathbb{R}^{512\times35\times35} \rightarrow \mathbb{R}^4$ such that $g(\mathbf{b}) = \mathbf{q}_v$. Differently from the pose decoder, g should not leverage the image processing done by the Time Contrastive Network encoder, as it needs to retain features from the image that are viewpoint dependent (and not invariant) in order to predict the orientation.

The input to the viewpoint decoder is the tensor $\mathbf{b} \in \mathbb{R}^{512\times35\times35}$, obtained from the layer "mixed-5d" of the Inception-V3 [52] backbone. Up to this layer, the encoder weights are initialized with the pre-trained values from the ImageNet classification task and *are not* updated during the self-supervised training procedure described in Section 9.3.1. Therefore, they should have retained the information needed for the viewpoint prediction task, which is instead discarded by the layers belonging to the Time Contrastive Network, trained using Eq. 9.1. We then add a set of 512 convolutional filters of spatial resolution 35×35, followed by a Spatial Softmax Layer [54] and a set of fully connected layers, using the architecture from [30], that reduce the final output dimension to 4.

Using the Hamilton product [62] to combine the outputs of the two decoders produces the camera-centric representation of 3D pose associated to an input image:

$$\mathbf{P}_{cam} = \mathbf{q}_v^* \cdot \mathbf{P}_{ego} \cdot \mathbf{q}_v, \tag{9.3}$$

where $\mathbf{P}_{ego}$ is the output of the ego-centric pose decoder, $\mathbf{q}_v$ is the output quaternion obtained by the viewpoint decoder and $\mathbf{q}_v^*$ is its conjugate.

The viewpoint decoder is trained jointly with the pose decoder using individual frames from the Human3.6M dataset [19] with one of two strategies. In the *disentangled training*, each branch is trained separately and the overall loss is the sum of the two individual decoder losses equally weighted:

$$\mathcal{L}_{decoders} = \mathcal{L}_{pose} + \mathcal{L}_{view},$$

where $\mathcal{L}_{pose}$ is defined in Section 9.3.2, and $\mathcal{L}_{view}$ is the negative dot product between the predicted and ground-truth quaternion $\mathcal{L}_{view} = -\langle \mathbf{q}_v, \hat{\mathbf{q}}_v \rangle$.

Alternatively, in the *entangled training*, the output of the two decoders is combined using Eq. 9.3, and the training loss is the eucledian distance loss using camera-centric ground-truth poses, $\mathcal{L}_{decoders}(\hat{\mathbf{P}}_{cam}, \mathbf{P}_{cam}) = ||\hat{\mathbf{P}}_{cam} - \mathbf{P}_{cam}||_2$.

The main difference between the two approaches is that the latter does not require to explicitly compute the ground-truth quaternion $\mathbf{q}_v$ corresponding to $\mathbf{R}_v$, even though it is always possible to obtain it from the camera-centric pose $\mathbf{P}_{cam}$, Section 9.3.2.

9.3.4 Implementation Details

Our baseline approach is presented in Figure 9.2, and consists of a pipeline composed of the Time Contrastive Network and the pose decoder. We leave to future investigations the experimentation of combining the outputs of the pose and viewpoint decoders.

The Time Contrastive Network is trained for a total of 100K iterations with a learning rate of 0.0001 using RAdam [63] with a decay factor of 0.96 applied every 10K iterations. The pose decoder has the same architecture as [51] with one stage and an internal layer size of 1024, and is trained for a total of 150K iterations using a learning rate of 0.001 decayed once at iteration 100K by a factor of 0.96. Contrary to [51], we are not using this network for 3D lifting, but instead for predicting a mapping from our 128 dimensional feature space into the output pose space. The encoder weights are kept frozen during the training of the pose decoder (with the exception of training the fully supervised baseline), so that we do not modify the previously learned embedding. All encoders we compare against are initialized with ImageNet [53] pretrained weights.

For the experimental evaluation we implemented the following baseline methods:

ImageNet

To verify if the self-supervised procedure is useful at all, we compare its performance to a network with the same architecture pretrained for image classification on ImageNet [53].

Camera Aware TCN

Using the geometry aware distance function of Rhodin [16] allows the encoder to predict a different embedding for every image and compare them by mapping them into the same part of the manifold using knowledge of the relative position between each pair of cameras. We modify the loss in Eq. 9.1 using the information about the relative rotation of the cameras as follows:

$$||h(\mathbf{I}_{t_i}^{v_n}) - h(\mathbf{I}_{t_i}^{v_m}) \cdot \mathbf{R}_{v_{m\to n}}||_2^2 + \alpha < ||h(\mathbf{I}_{t_j}^{v_n}) - h(\mathbf{I}_{t_i}^{v_n}) \cdot \mathbf{R}_{v_{m\to n}}||_2^2. \tag{9.4}$$

As a result, the embeddings are camera-centric, but their comparison is done in a ego-centric space.

Oracle TCN

One of the fundamental problems of our methodology is to determine how to sample images during training. This method uses the same architecture of our baseline, but samples according to the 3D pose ground-truth. In practice, we construct our batches by randomly sampling frames from multiple subjects in the batch. However, different from our previous method we then use the distance in 3D pose space to determine how to construct valid triplets. This provides signal (that is not available in standard self-supervised training) that brings the representation of images of different people closer to each other in the embedding space. We use the metric N-MPJPE (normalized mean per joint position error) [19] to determine the distance between two poses, and put a threshold such that if N-MPJPE($\mathbf{P}^{t_j}_{ego}, \mathbf{P}^{t_j}_{ego}$) > 50$mm$ we treat them as anchor and negative.

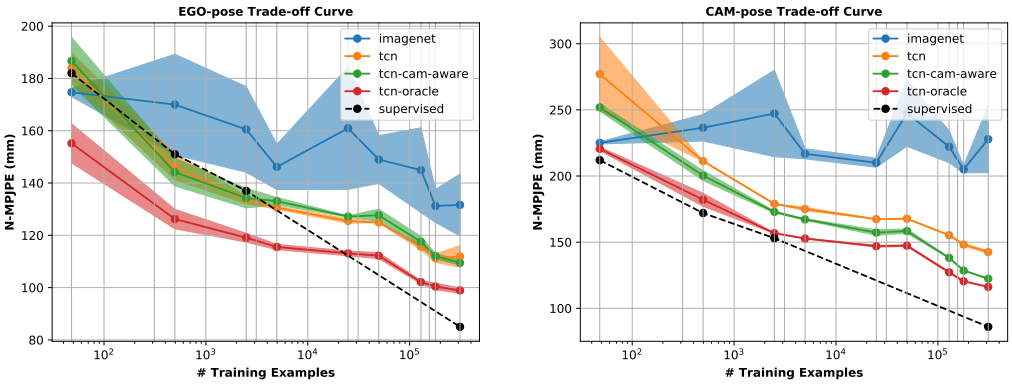
Supervised

We compare our self-supervised baseline to a fully-supervised end-to-end network in which the 3D pose is used to minimize the euclidean distance between predicted poses and ground-truth annotations. In this version, there is no distinction between embedder and decoder, but just one unique network with the same number of parameters. This strong baseline changes the weights of the backbone.

9.4 Experiments

We conduct a detailed experimental evaluation of distance metric based multi-view self-supervised feature learning for single image 3D pose estimation with the goal of answering the following questions:

- Are the features learned via multi-view self-supervision predictive of 3D pose?
- Does the learned feature space capture both ego and camera-centric pose?
- How much does additional viewpoint information help at training time?
- What impact does triplet selection have during training?
- Are the learned features invariant to identity?
- What is the impact of the dimensionality of the learned feature space?
- Are simple 3D pose decoders sufficient for supervised finetuning?



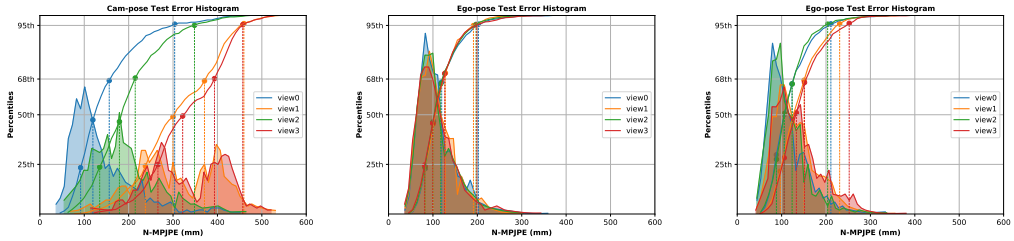
(a) Ego-centric pose (b) Camera-centric pose

Figure 9.3: **Amount of supervision versus reconstruction error trade-off curve for Ego versus Camera-Centric pose representation.** We compare several approaches to multi-view self-supervised feature learning by finetuning the learned features with different amounts of ground-truth 3D pose data. The task of ego-centric pose estimation in (a) is easier than that of camera-centric (b), but we observe a similar ranking across both tasks. Here the error bars represent the difference between the best and last models obtained during finetuning.

9.4.1 Ego versus Camera-Centric 3D Pose Finetuning

For our first experiment, we measure how much pose information do the features of different multi-view self-supervised learning methods capture. We extract a 128 dimensional embedding vector for each image in the training set, and finetune a fully connected pose decoder using an architecture similar to Martinez [30], while freezing the weights of the encoder network. By finetuning with different amounts of supervision, we can quantify how well do the features learned by each self-supervised approach generalize to the task of monocular 3D pose estimation.

In Figure 9.3-(a), we see that most approaches improve as the amount of ego-centric finetuning data increases. However, we also observe that most of the self-supervised approaches are worse than the fully supervised baseline trained end-to-end. In general, as more finetuning data is provided the 3D pose performance improves. Interestingly, features extracted from the same encoder architecture that has been pretrained on ImageNet [53] (`imagenet`) does not perform as well as the self-supervised baselines. Also the fully supervised baseline (`supervised`) still performs competitively when there is limited training data. However, the encoder and decoder for this network are both finetuned during training. Qualitative predictions are displayed in Figure 9.10.



(a) Cam: PT 4, FT 1 (b) Ego: PT 4, FT 1 (c) Ego: PT 2, FT 1

Figure 9.4: **Generalization of multi-view self-supervision to new viewpoints.** In each of the three cases, we pretrain (PT) our backbone `tcn` encoder with either two or four camera views, finetune (FT) the pose decoder with data from only one camera (view 0), and then test then pose decoder using data from all four views. (a) The pose decoder fails to generalize to new views when predicting camera-centric pose. (b) For ego-centric pose, where the features are truly viewpoint invariant, there is little impact of limited viewpoint finetuning. (c) If we only self-supervise the backbone with data from two views we observe a drop in performance in ego-centric prediction compared to using more views, as in (b).

By comparing the performance of ego-centric and camera-centric representations in Figure 9.3, we observe some interesting differences. While the conventional `tcn` encoder and the geometry-aware `tcn-cam-aware` [16] encoder perform comparably for ego-centric pose prediction, Figure 9.3-(a), the `tcn-cam-aware` features are better suited when jointly decoding camera information as well as pose, as is the case for the camera-centric evaluation task, Figure 9.3-(b). Surprisingly, this ability does not result in a drop in ego-centric performance, at least for anything but very small amounts of finetuning data. It is worth pointing out that while the `tcn-cam-aware` encoder is effective, it relies on having knowledge of the relative position between each pair of cameras [16] during the self-supervised training phase. This would be very challenging to acquire in the more general framework of moving cameras, limiting the applications in which such methodology can be deployed.

9.4.2 Impact of Viewpoints During Training

We explore the impact on performance of the availability of additional camera viewpoint information when both training the encoder and finetuning the pose decoder. In our experiments, we varied the number of camera viewpoints available (by selecting from the four views in Human3.6M [19]) to either the self-supervised network or pose decoder when finetuning. In Figure 9.4-(a), we observe that the camera-centric pose decoder fails to generalize to images from views other than the one it was finetuned on (view 0 in this case). In contrast, the features for the

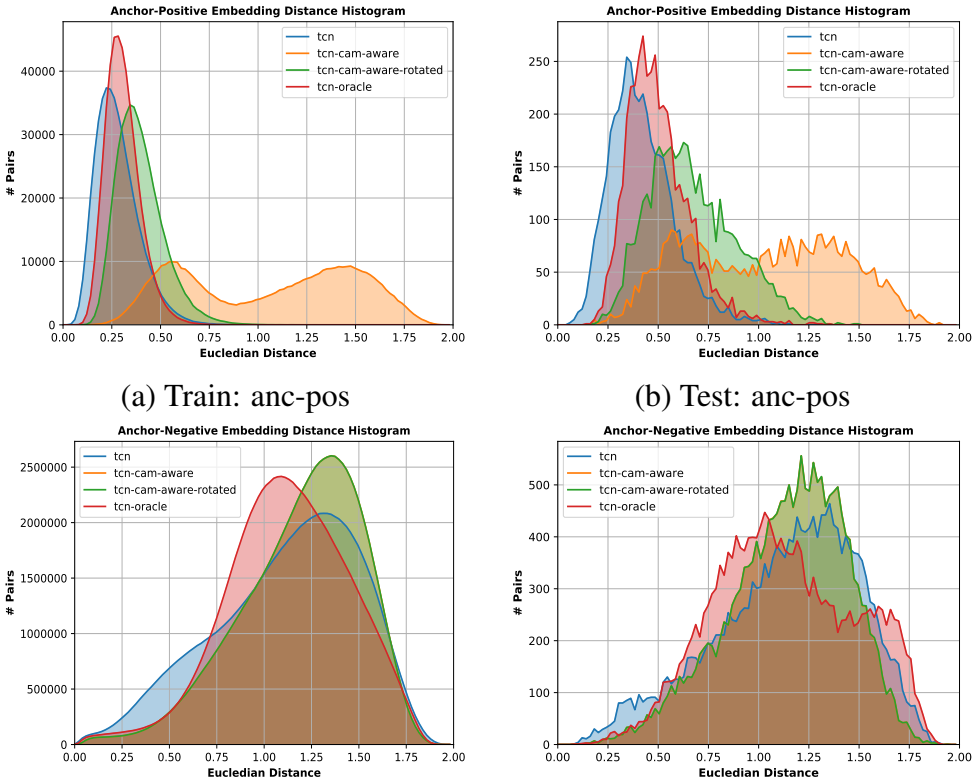
ego-centric decoder are invariant to viewpoint, and result in a similar distribution of errors for images from different cameras at test time, Figure 9.4-(b). When only self-supervising the backbone with images from two views instead of four (using views 0 and 2), we observe that there is a drop in performance for the ego-centric decoder, Figure 9.4-(c). We explored all possible permutations of viewpoints, and the results were consistent.

9.4.3 Image Selection During Training

Learning informative self-supervised 3D pose features relies on pushing features for pairs of images that contain different poses apart in the feature space and pulling features with similar poses towards each other. The only strong signal we have for multi-view training is the subset of images taken at the same time from different viewpoints, that we know, by construction, to contain the same pose. As a result, there is a strong likelihood that over training our baseline method will incorrectly sample both false positives and false negatives.

With the method `tcn-oracle`, we present results for our baseline when using an 'oracle' image sampler, that uses knowledge of 3D pose distance when constructing positive and negative training examples when training the backbone encoder, so that false positives and false negatives cannot occur. Obviously, this information would not be available at training time in the regular framework, but we evaluate it here to get an upper limit on the potential performance of our embedding approach. In Figure 9.3 we observe that `tcn-oracle` outperforms the the simple `tcn` baseline, regardless of the pose representation adopted. It does not reach the performance of the supervised baseline `supervised`, but we hypothesis this is explained by the fact that the supervised baseline is trained end-to-end, so even the backbone features are optimized.

To further illustrate the difference between the simple sampling used by the baseline `tcn` and the oracle sampler we plot the distances in the embedding space between anchor-positive and anchor-negative pairs of frames. By comparing the baseline `tcn` encoder and the `tcn-oracle` encoder in Figure 9.5 ((a) vs (c) and (b) vs (d)), we see that there is less overlap between the anchor-positive and anchor-negative frames for the oracle sampler. This is expected, as the oracle sampler is explicitly trained with ground-truth pose information, but indicates in a quantitative way the value of better methods for selecting pairs of frames during self-supervised training [13, 18].



(a) Train: anc-pos

(b) Test: anc-pos

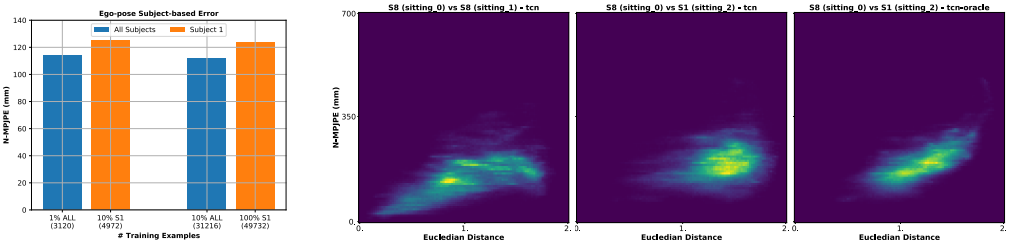
(c) Train: anc-neg

(d) Test: anc-neg

Figure 9.5: **Distribution of pairwise distances in the learned embedding space.** We show the distribution of pairwise distances for all the anchor positive (anc-pos) and anchor negative (anc-neg) pairs of frames in the training (a and c) and test (b and d) sets. To obtain meaningful anchor positive distances in the embedding space the `tcn-cam-aware` method needs to be rotated before comparing the image pairs, `tcn-cam-aware-rotated`. This is not necessary for the anchor negative distances, as they are sampled from the same camera view.

9.4.4 Identity Invariant Features

During the self-supervision stage, we would ideally learn image features that are invariant to identity, in such a way that the same 3D pose exhibited by two different people should project to the same location in the embedding space. To test this hypothesis, we finetune a pose decoder using the learned features from the `tcn` encoder with ground-truth ego-centric pose and vary the amount of *i)* training subjects and *ii)* training poses, while keeping the number of subjects fixed to one. We observe in Figure 9.6-(a), that using 3K frames across all the train subjects (1% All) is more effective in terms of reducing the pose reconstruction error compared to using 5K frames from one subject (10% S1). Similarly, if we increase both finetuning sets by a factor of ten we observe the same relationship (10% All versus 100% S1).



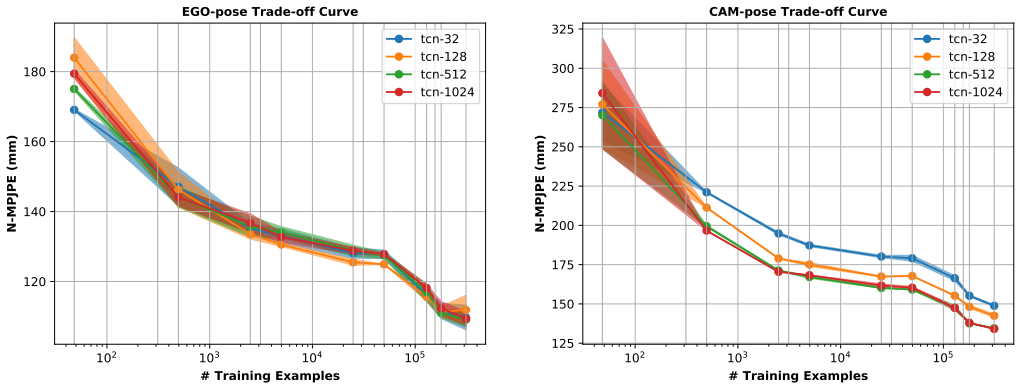
(a) Identity versus train (b) Pose distance versus embedding distance

Figure 9.6: **Quantitative and qualitative analysis of the invariance to identity of the learned embedding.** (a) Increasing the amount of data across subjects, as opposed to for a single subject, has a larger impact on performance. (b) We compare the distance in the learned feature space and their 3D ego-centric pose distance for pairs of images sampled from two different videos of a same action. We use three different settings: *i)* `tcn` / same subject, *ii)* `tcn` / different subjects, and *iii)* `tcn-oracle` / different subjects. The relationship between pose and feature distance is better captured for the same individual: *i)* versus *ii)*. However, *iii)* the `tcn-oracle` features present stronger correlation even for different individuals.

In fact, we can see that finetuning with only 3K frames from all the training subjects is superior to using 50K frames from only one subject. If the embedding space was truly invariant to identity we would not expect to see this trend.

In Figure 9.6-(b), we visualize the relationship between distance in the 3D ego-centric pose space and distance in the learned feature space. The heatmaps are obtained by randomly selecting pairs of frames over two video sequences and plotting the distance values in both spaces against each other. To aid visualization, we binned the distances to have an equal number of bins between a minimum and maximum N-MPJPE value of $[0, 700]$ mm for the pose space, and minimum and maximum eucledian distance value of $[0, 2]$ in the embedding space.

For a well learned embedding, there would be a strong correlation between the two distances, so that pairs of frames that are further apart in the embedding space have different 3D poses, and vice-versa. We study this behavior when sampling frames from different videos and compare the `tcn` and `tcn-oracle` features. We can see that, when the frames are sampled from two videos of the same subject performing a similar action, the `tcn` features show the desired correlation. However, that is not the case if the pairs are sampled from videos of different subjects performing a similar action, as there are multiple pairs of images with similar pose but a large of distances in the feature space. This indicates that the `tcn` features still encode some identity information. On the other hand, the `tcn-oracle` features show the desired correlation in the latter case, proving to be more invariant to subject identity.



(a) Ego-centric pose (b) Camera-centric pose

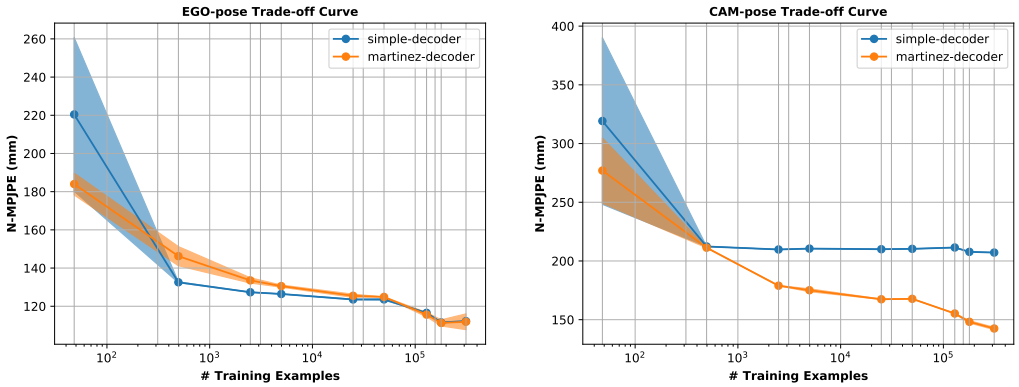
Figure 9.7: **Sensitivity of the data versus error trade-off curve to the embedding dimension.** (a) Changing the dimensionality of the output layer of the pose encoder does not impact performance on the ego-centric pose prediction task. (b) However, we see that a larger dimensionsionality can improve the camera-centric prediction, up to a point.

9.4.5 Embedding Dimensionality

In our baseline method, we used a default embedding dimension of 128. In Figure 9.7, we vary this dimensionality to see if it has an impact on the perforance on the task of 3D pose prediction, and if that changes based on the specific pose representation adopted. For each of these experiments, we simply change the number of neurons in the first layer of the pose decoder so that it is the same as the embedding dimensionality and keep all other settings un-changed.

For ego-centric pose prediction, Figure 9.7-(a), we observe no noticeable difference except for when there is very small amounts of fine-tuning data. This is not the case for the camera-centric pose prediction, Figure 9.7-(b), in which increasing the dimension of the embedding results in a consistent improvement in the accuracy of the predicted 3D pose, up to a point (as there is no noticeable difference between 512 and 1024).

We believe this is due to the fact that having a decoder with more parameters can help with the harder task of recovering the camera-centric pose from the viewpoint-invariant embedding. However, if the dimension is too large, performance for very small amounts of fine-tuning data available becomes much worst, as there is not enough data to fit the larger number of parameters.



(a) Ego-centric pose (b) Camera-centric pose

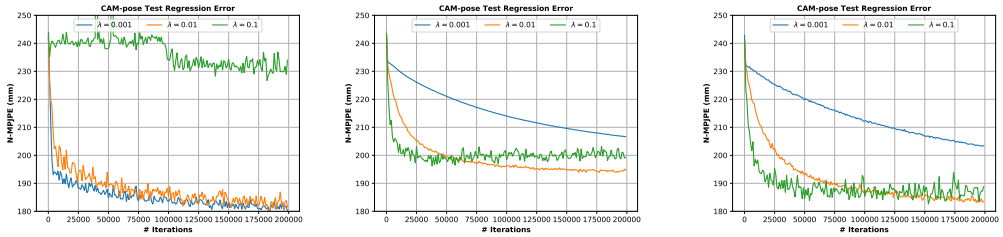
Figure 9.8: **Impact of the pose decoder architecture on the amount of data versus reconstruction error trade-off curve.** We compare the performance of a simple decoder consisting only of a linear layer between the embedding and the output poses with a more complex decoder inspired by Martinez [30]. By comparing the ego-centric and camera-centric plots we see that the simple pose decoder is able to decode ego-centric pose as effectively as the more complex decoder. However, in the case of camera-centric pose, it does not have enough capacity to predict the correct pose (and viewpoint) and even fails to benefit from additional training data.

9.4.6 Impact of the Pose Decoder

To better understand how much 3D pose information is directly encoded in the learned feature space we compare the performance of two different pose decoders: *i)* a simple linear regressor, consisting of one neural network layer going from the embedding space to the output pose space, and *ii)* the default multi-layer fully connected network inspired by the residual architecture used in Martinez [30] used in all the previous experiments. We finetune the features with different amounts of 3D supervised pose data using the same hyper parameters for both decoders.

In Figure 9.8-(a), we observe that the simple pose decoder performs equivalently to the more complex one for the task of ego-centric pose estimation, across almost all amounts of available training data. However, it fails to predict camera-centric pose even as the amount of training data is increased, Figure 9.8-(b). This further supports the idea that the learned feature space is invariant to camera viewpoint and as a result a more complex decoder is necessary to successfully learn how to decode both pose and camera viewpoint at the same time from the embedding space.

In fact, the simple regressor does not have enough capacity to perform that task in the allotted number of training iterations.



(a) M-decoder (5% S1) (b) S-decoder (5% S1) (c) S-decoder (100% ALL)

Figure 9.9: **Learning rate sensitivity for different pose decoder architectures.** (a) The complex pose decoder inspired by [51] (M-decoder) is more sensitive to larger learning rates but shows similar convergence properties for smaller ones. (b and c) The simple pose decoder (S-decoder) performs less well with our default learning rate of 0.001, but fails to approach the same performance of the larger one, even with better hyper-parameters. This indicates that carefully choosing hyper-parameters is very important, as convergence and final performance can vary significantly across different pose decoders.

To verify if the previous observation is the justification of the observed discrepancy in camera-centric performance between the two decoders, we trained the two architectures using multiple learning rate values for twice as many iterations, for a total of 200,000.

In Figure 9.9-(b), we observe that while the simple decoder benefits from the increase in training iterations when using the default learning rate of 0.001, it still performs much worse than the larger one, and has not yet converged. By comparing with Figure 9.9-(c), we also see more clearly the effect on the reconstruction error of adding additional training data. For the learning rate of 0.001, if more data is available even more iterations are required to minimize the reconstruction error.

In comparison, the complex decoder in Figure 9.9-(a) converges much faster and to a smaller error, also when using a smaller amount of finetuning data (5% of subject 1). It is also more robust to the learning rate, as long as it is not too large.

The results presented in our additional analysis indicate that the camera-centric prediction task has a much higher sensitivity to hyper-parameters such as the embedding dimension, Figure 9.7, or the decoder architecture and learning rate, Figure 9.9. We believe this is due to the fact that the encoder was trained to retain only ego-centric (viewpoint invariant) information, and that while the results show that it is still possible, recovering camera-centric coordinates has a much more complicated optimization landscape.

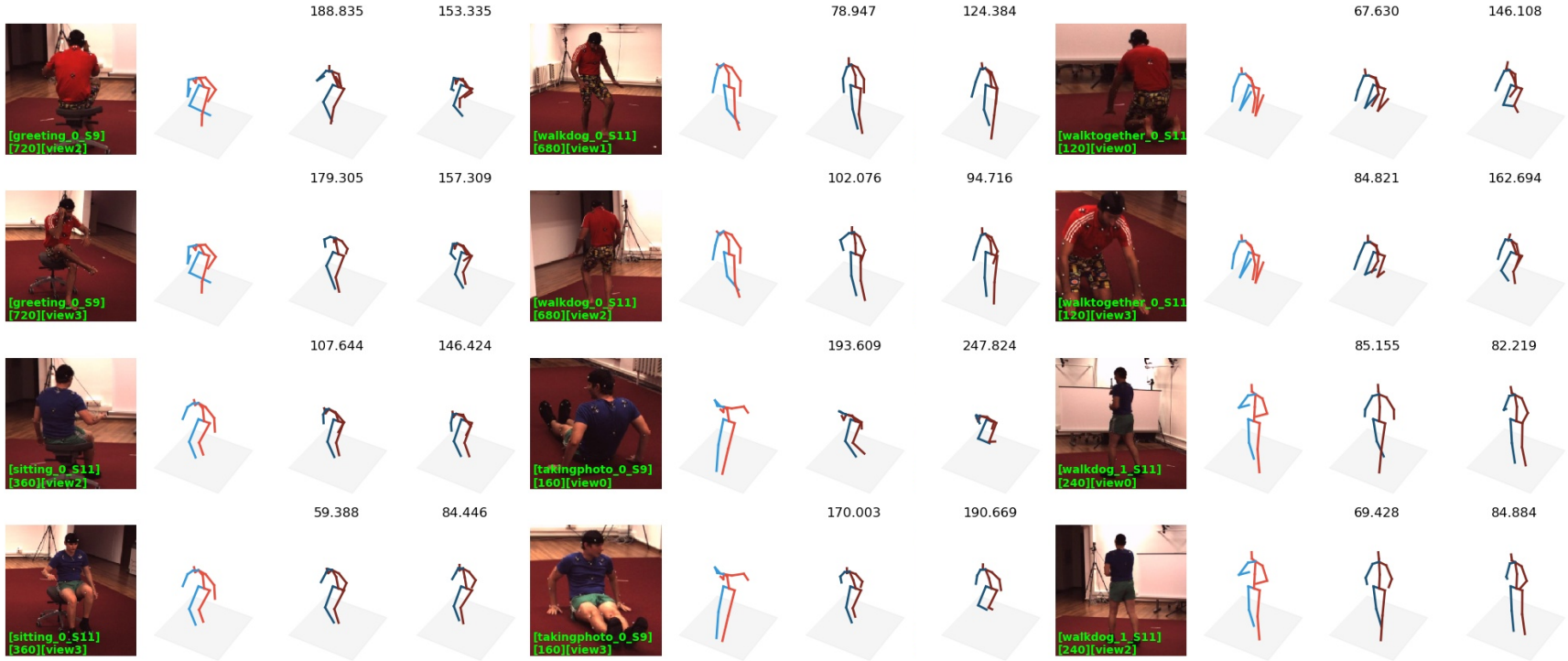


Figure 9.10: **Inference time ego-centric 3D pose predictions on the Human3.6M dataset.** Ego-centric pose predictions obtained by finetuning the pose decoder with two different self-supervised feature learners. For six test set frames, we show on each column: *i)* the input image, *ii)* the ground-truth pose, *iii)* the pose prediction trained on the `tcn-oracle` encoder, and *iv)* the pose prediction trained on the `tcn` encoder. For every example, we show the outputs obtained from two different viewpoints and can observe that the pose decoder prediction is ego-centric for both input images. We can see an improvement in the predicted output poses for the `tcn-oracle` method, particularly in the cases of complicated poses and significant occlusion. Note how The mean per joint prediction error (MPJPE) in mm is reported above each pair of predictions.

9.5 Discussion and Recommendations

In this Section we distill the observations from the previous experimental results and make recommendations for future work.

Camera-centric predictions are more difficult: Predicting in ego-centric pose space is consistently easier than camera-centric for the `tcn` based approaches we evaluated, see Figure 9.3. Depending on the down-stream application, ego-centric pose may be sufficient. However, camera-centric pose is aligned directly to the image and ego-centric pose can easily be extracted from the camera-centric pose. Going the opposite direction, from ego to camera pose, requires correspondences between the predicted pose and image space or knowledge of the camera pose relative to the person. While geometry-aware embeddings [16] make camera-centric pose recovery easier, they require knowledge of the camera extrinsics during the self-supervision stage. We will investigate in the future if, and to what extent, is the viewpoint decoder branch in our proposed pipeline able to predict the viewpoint information without needing that knowledge in the self-supervision stage.

Image selection at training time matters: Distance metric approaches like the ones evaluated here fundamentally rely on the availability of informative pairs during training. This is challenging for self-supervised learning, as we do not know which frames contain the same, or different, 3D poses. The results of the `tcn-oracle` method, Figure 9.3-(a), show that there is a lot more performance to be extracted by better sampling. More work needs to be done to investigate what other information can be used to improve this crucial step.

Current embeddings are not identity invariant: Our experiments show that less ground-truth supervision from more individuals is much more valuable than more data from one individual, see Figure 9.6-(a). This is not necessarily surprising, but we conjecture that a more effective embedding space would be invariant to the identity and appearance of the people in the images. Instead, we observed large differences in the statistics of the embeddings across different subjects, Figure 9.6-(b). This is likely due to the limitations of current multi-view datasets, in which there is no way to obtain correspondences between different subjects performing the same action, but future work would benefit from automatically factoring out this information.

9.6 Conclusion

We performed a detailed experimental evaluation of multi-view self-supervised feature learning for single image 3D pose estimation. While self-supervised learning offers the promise of learning informative features with limited supervision, we showed that in case of 3D pose estimation there are still several open problems that prevent self-supervised approaches from closing the gap to fully supervised baselines. These problems include issues related to image sampling during training, the 3D prediction frame of reference, and generalization to new views and unseen people. Through ablation experiments we showed that addressing these problems will result in a large reduction in 3D pose error.

References

[1] A. Toshev and C. Szegedy,
"Deeppose: Human pose estimation via deep neural networks",
in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1653–1660 (cit. on p. 138).

[2] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler, "Joint training of a convolutional network and a graphical model for human pose estimation", in *NIPS*, 2014 (cit. on p. 138).

[3] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation", *ArXiv preprint arXiv:1603.06937*, 2016 (cit. on p. 138).

[4] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh,
"Convolutional pose machines", in *CVPR*, 2016 (cit. on p. 138).

[5] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh,
"Realtime multi-person 2d pose estimation using part affinity fields",
in *CVPR*, 2017 (cit. on p. 138).

[6] J. O'rourke and N. I. Badler, "Model-based image analysis of human motion using constraint propagation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 522–536, 1980 (cit. on p. 138).

[7] D. Hogg, "Model-based vision: A program to see a walking person",
Image and Vision computing, vol. 1, no. 1, pp. 5–20, 1983 (cit. on p. 138).

[8] K. Rohr, "Incremental recognition of pedestrians from image sequences",
in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 1993, pp. 8–13 (cit. on p. 138).

[9] J. M. Rehg and T. Kanade, "Visual tracking of high dof articulated structures: An application to human hand tracking",
in *European conference on computer vision*, Springer, 1994, pp. 35–46 (cit. on p. 138).

[10] S. Li and A. B. Chan, "3d human pose estimation from monocular images with deep convolutional neural network", in *ACCV*, 2014
(cit. on pp. 138, 140).

[11] X. Zhou, M. Zhu, S. Leonardos, K. G. Derpanis, and K. Daniilidis,
"Sparseness meets deepness: 3d human pose estimation from monocular video", in *CVPR*, 2016 (cit. on p. 138).

[12] C.-H. Chen and D. Ramanan,
"3d human pose estimation= 2d pose estimation+ matching", in *CVPR*, 2017
(cit. on pp. 138, 140).

[13] T. Milbich, M. Bautista, E. Sutter, and B. Ommer, "Unsupervised video understanding by reconciliation of posture similarities", in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4394–4404 (cit. on pp. 139, 142, 153).

[14] D. Dwibedi, Y. Aytar, J. Tompson, P. Sermanet, and A. Zisserman, "Temporal cycle-consistency learning", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1801–1810 (cit. on pp. 139, 142, 143).

[15] C.-H. Chen, A. Tyagi, A. Agrawal, D. Drover, S. Stojanov, and J. M. Rehg, "Unsupervised 3d pose estimation with geometric self-supervision", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5714–5724 (cit. on pp. 139, 142).

[16] H. Rhodin, M. Salzmann, and P. Fua, "Unsupervised geometry-aware representation for 3d human pose estimation", in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 750–767 (cit. on pp. 139, 141, 143, 145, 149, 152, 160).

[17] R. Mitra, N. B. Gundavarapu, A. Sharma, and A. Jain, "Multiview-consistent semi-supervised learning for 3d human pose estimation", *ArXiv:1908.05293*, 2019 (cit. on pp. 139, 142).

[18] O. Sumer, T. Dencker, and B. Ommer, "Self-supervised learning of pose embeddings from spatiotemporal relations in videos", in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4298–4307 (cit. on pp. 139, 142, 153).

[19] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1325–1339, Jul. 2014 (cit. on pp. 140, 141, 143, 144, 145, 146, 148, 150, 152).

[20] D. Pavllo, C. Feichtenhofer, D. Grangier, and M. Auli, "3d human pose estimation in video with temporal convolutions and semi-supervised training", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7753–7762 (cit. on p. 140).

[21] G. Pavlakos, X. Zhou, K. G. Derpanis, and K. Daniilidis, "Coarse-to-fine volumetric prediction for single-image 3d human pose", *ArXiv preprint arXiv:1611.07828*, 2016 (cit. on p. 140).

[22] G. Varol, D. Ceylan, B. Russell, J. Yang, E. Yumer, I. Laptev, and C. Schmid, "Bodynet: Volumetric inference of 3d human body shapes", in *ECCV*, 2018 (cit. on p. 140).

[23] D. Tome, C. Russell, and L. Agapito, "Lifting from the deep: Convolutional 3d pose estimation from a single image", *CVPR*, 2017 (cit. on p. 140).

[24] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose", in *CVPR*, 2018 (cit. on p. 140).

[25] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black, "Keep it smpl: Automatic estimation of 3d human pose and shape from a single image", in *ECCV*, 2016 (cit. on p. 140).

[26] X. Zhou, Q. Huang, X. Sun, X. Xue, and Y. Wei, "Towards 3d human pose estimation in the wild: A weakly-supervised approach", in *ICCV*, 2017 (cit. on pp. 140, 145).

[27] I. Habibie, W. Xu, D. Mehta, G. Pons-Moll, and C. Theobalt, "In the wild human pose estimation using explicit 2d features and intermediate 3d representations", in *CVPR*, 2019 (cit. on p. 140).

[28] H. Rhodin, J. Spörri, I. Katircioglu, V. Constantin, F. Meyer, E. Müller, M. Salzmann, and P. Fua, "Learning monocular 3d human pose estimation from multi-view images", in *CVPR*, 2018 (cit. on p. 140).

[29] H. Qiu, C. Wang, J. Wang, N. Wang, and W. Zeng, "Cross view fusion for 3d human pose estimation", in *ICCV*, 2019 (cit. on p. 140).

[30] J. Martinez, R. Hossain, J. Romero, and J. J. Little, "A simple yet effective baseline for 3d human pose estimation", in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2640–2649 (cit. on pp. 140, 148, 151, 157).

[31] F. Moreno-Noguer, "3d human pose estimation from a single image via distance matrix regression", in *CVPR*, 2017 (cit. on p. 140).

[32] H. Yasin, U. Iqbal, B. Kruger, A. Weber, and J. Gall, "A dual-source approach for 3d pose estimation from a single image", in *CVPR*, 2016 (cit. on p. 140).

[33] W. Yang, W. Ouyang, X. Wang, J. Ren, H. Li, and X. Wang, "3d human pose estimation in the wild by adversarial learning", in *CVPR*, 2018 (cit. on p. 140).

[34] B. Wandt and B. Rosenhahn, "Repnet: Weakly supervised training of an adversarial reprojection network for 3d human pose estimation", in *CVPR*, 2019 (cit. on p. 140).

[35] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2d human pose estimation: New benchmark and state of the art analysis", in *CVPR*, 2014 (cit. on p. 141).

[36] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context", in *Computer Vision–ECCV 2014*, Springer, 2014, pp. 740–755 (cit. on p. 141).

[37] L. Sigal, A. O. Balan, and M. J. Black, "Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion", *IJCV*, 2010 (cit. on p. 141).

[38] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh, "Panoptic studio: A massively multiview system for social motion capture", in *ICCV*, 2015 (cit. on p. 141).

[39] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko, W. Xu, and C. Theobalt, "Monocular 3d human pose estimation in the wild using improved cnn supervision", in *3DV*, 2017 (cit. on p. 141).

[40] T. Shu, M. S. Ryoo, and S.-C. Zhu, "Learning social affordance for human-robot interaction", *IJCAI*, 2016 (cit. on p. 141).

[41] W. Chen, H. Wang, Y. Li, H. Su, Z. Wang, C. Tu, D. Lischinski, D. Cohen-Or, and B. Chen, "Synthesizing training images for boosting human 3d pose estimation", in *3DV*, 2016 (cit. on p. 141).

[42] G. Varol, J. Romero, X. Martin, N. Mahmood, M. J. Black, I. Laptev, and C. Schmid, "Learning from synthetic humans", in *CVPR*, 2017 (cit. on p. 141).

[43] D. Tome, P. Peluse, L. Agapito, and H. Badino, "xR-EgoPose: Egocentric 3D Human Pose from an HMD Camera", in *ICCV*, 2019 (cit. on p. 141).

[44] C. J. Taylor, "Reconstruction of articulated objects from point correspondences in a single uncalibrated image", in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, IEEE, vol. 1, 2000, pp. 677–684 (cit. on p. 141).

[45] L. Bourdev and J. Malik, "Poselets: Body part detectors trained using 3d human pose annotations", in *Computer Vision, 2009 IEEE 12th International Conference on*, IEEE, 2009, pp. 1365–1372 (cit. on p. 141).

[46] G. Pavlakos, X. Zhou, and K. Daniilidis, "Ordinal depth supervision for 3d human pose estimation", *CVPR*, 2018 (cit. on p. 141).

[47] M. R. Ronchi, O. Mac Aodha, R. Eng, and P. Perona, "It's all relative: Monocular 3d human pose estimation from weakly supervised data", *ArXiv preprint arXiv:1805.06880*, 2018 (cit. on p. 141).

[48] X. Chen, K.-Y. Lin, W. Liu, C. Qian, and L. Lin, "Weakly-supervised discovery of geometry-aware representation for 3d human pose estimation", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 895–10 904 (cit. on p. 141).

[49] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, and S. Levine, "Time-contrastive networks: Self-supervised learning from pixels", 2017 (cit. on pp. 142, 143, 145).

[50] M. Kocabas, S. Karagoz, and E. Akbas, "Self-supervised learning of 3d human pose using multi-view geometry", *ArXiv preprint arXiv:1903.02330*, 2019 (cit. on p. 142).

[51] J. Martinez, M. J. Black, and J. Romero, "On human motion prediction using recurrent neural networks", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2891–2900 (cit. on pp. 143, 149, 158).

[52] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826 (cit. on pp. 145, 148).

[53] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge", *IJCV*, 2015 (cit. on pp. 145, 149, 151).

[54] C. Finn, X. Y. Tan, Y. Duan, T. Darrell, S. Levine, and P. Abbeel, "Deep spatial autoencoders for visuomotor learning", in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2016, pp. 512–519 (cit. on pp. 145, 148).

[55] T.-T. Do, T. Tran, I. Reid, V. Kumar, T. Hoang, and G. Carneiro, "A theoretically sound upper bound on the triplet loss for improving the efficiency of deep distance metric learning", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 404–10 413 (cit. on p. 145).

[56] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823 (cit. on p. 145).

[57] A. Mishchuk, D. Mishkin, F. Radenovic, and J. Matas, "Working hard to know your neighbor's margins: Local descriptor learning loss", in *NeurIPS*, 2017 (cit. on p. 146).

[58] S. Liao, E. Gavves, and C. G. Snoek, "Spherical regression: Learning viewpoints, surface normals and 3d rotations on n-spheres", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9759–9767 (cit. on p. 147).

[59] S. Mahendran, H. Ali, and R. Vidal,
"3d pose regression using convolutional neural networks",
in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2174–2182 (cit. on p. 147).

[60] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li,
"On the continuity of rotation representations in neural networks",
in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5745–5753 (cit. on p. 147).

[61] D. Pavllo, D. Grangier, and M. Auli, "Quaternet: A quaternion-based recurrent model for human motion",
ArXiv preprint arXiv:1805.06485, 2018 (cit. on p. 147).

[62] W. R. Hamilton, "Xi. on quaternions; or on a new system of imaginaries in algebra", *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 33, no. 219, pp. 58–60, 1848 (cit. on p. 148).

[63] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, "On the variance of the adaptive learning rate and beyond",
ArXiv preprint arXiv:1908.03265, 2019 (cit. on p. 149).

Conclusions

Chapter 10

SUMMARY OF THESIS AND CONTRIBUTIONS

The research conducted throughout my PhD and presented in this Thesis investigated elements of the statement of work put forward in Chapter 2 in the context of social interactions between robots and human agents.

We provided a definition of social robots and characterized what perceptual abilities they would need to master to effectively function in a world rich of visual stimuli and to operate alongside humans, interacting both with people and objects.

We described ways in which Computer Vision can support the development and improvement of social robots' visual skills with specific focus on tasks that require interactions with humans.

We investigated the problems of scene understanding, and 3D human pose and motion estimation from monocular images using weakly supervised training data.

In its entirety, the work of this Thesis addressed the following questions:

- What information can be captured from a static scene that best helps inferring its context?
- Is it possible to recover information from the real world that is lost in the projection to monocular images?
- How can one extract information from images and videos on how humans are positioned and move?
- Can one develop algorithms that work accurately and have good generalization ability, even for tasks in which there is little high-quality training data?

Let us summarize the fundamental, novel observations supported by this Thesis:

Visual actions performed by humans and objects of interaction in images follow a very long tailed distribution. Humans portrayed during everyday actions typically perform many similar tasks using a small number of objects. Currently, only about 140 human interactions can be visually identified, and most of them are rare occurances. This highlights the importance of developing learning algorithms

that are robust to small amounts of training data. In fact, while robots can already perform a few tasks very well, other complex interaction tasks may be hard to learn because they occur less frequently and are not easily observable.

Images of human faces contain information on a scene's spacial setting. The distance at which a picture was taken can be obtained from a single monocular image of a human's face by analyzing the changes in specific visual cues caused by perspective distortion. This can be particularly useful to social robots, since interactions with humans should occur at different physical distances to support comfort and to respect social norms.

Current benchmarks and aggregate performance metrics miss real-world variability. In the task of multi-instance 2D human pose estimation *i)* algorithms with similar aggregate scores actually display strikingly different behavior when analyzing errors along multiple dimensions, and *ii)* biases and limited variability of the training data cause current metrics to overestimate the performance in real-world scenarios. Novel improved metrics and evaluation procedures can allow to better capture the true visual abilities of state-of-the-art algorithms, and provide the directions of improvements necessary for the deployment of social robots.

3D human pose and motion can be recovered from monocular images without using any 3D ground-truth data. Although people can assume a wide range of highly non-rigid poses, the human body is anatomically constrained and its parts' proportions, allowed motions and joint angles are well understood. When 3D ground-truth data is either lacking or available only in small quantities, one can leverage such understanding, together with additional constraints related to alternative types of supervisory signals, to learn models that are able to regress the full 3D pose of the human body and predict its motions simply from monocular 2D images. The above observation is supported by succesfully reconstrucing: *i)* short 3D body motions using only 2D keypoints from images of sports actions taken at various times from known camera viewpoints; *ii)* the static 3D pose of people from images using only 2D keypoints and relative depth labels for pairs of keypoints; *iii)* the static 3D pose of people from 2D images using synchronized video from multiple unknown camera viewpoints. This is fundamental for providing social robots with the ability to understand the 3D pose of humans with whom they are interacting, and open up the possibility of richer interactions in scenarios for which it is unrealistic to expect full 3D pose supervision.

Chapter 11

FUTURE STEPS TOWARDS SOCIAL ROBOTS

In Chapter 9, we described a methodology for learning an embedding of the pose of the human body using only training data in the form of synchronized videos from multiple uncalibrated cameras. The embedding is viewpoint-invariant and can be used to decode poses of the human body in both an ego-centric and camera-centric representation with a small metric error on the 3D reconstruction.

The synchronized videos, however, were taken in an indoor environment and with fixed cameras, while the proposed methodology does not require such limiting assumptions. A future direction of research we plan to pursue is to quantitatively verify the validity of our approach when training is performed on more complex, *in-the-wild* video collections.

To this end, we designed a stereo ego-centric video capturing rig, shown in Figure 11.1, composed of two Go-Pro Hero Session cameras with a 120° horizontal FOV which can record videos at 1080p and 60fps. The cameras are positioned in a stereo setup with a baseline in the range of $[60, 120]$mm to provide an approximate depth range of about $[.5, 5]$m, sufficient for capturing most of the relevant interactions between humans which we wish to observe.

In Figure 11.2, we show the data collection setup: two or more people wearing the video recording rigs will be moving while filming one or multiple actors performing a task, and possibly also interacting with each other in some way.

This will provide a multi-view setup between the pairs of cameras on different rigs, in which the camera locations and their relative orientation change in time, from a variety of viewpoints, dynamic backgrounds and illumination conditions. Further, the change in pose of each rig could be tracked using additional sensors, such as Inertial Measurement Units (IMUs). At the same time, the use of a calibrated stereo setup between the two cameras on each individual rig will allow collecting the ground-truth 3D position of the parts of the human body[1], and can be used to estimate the performance of the 3D pose estimation algorithm.

[1]The 2D locations of body parts in each monocular image can be annotated by humans or estimated through a state-of-the-art algorithm.

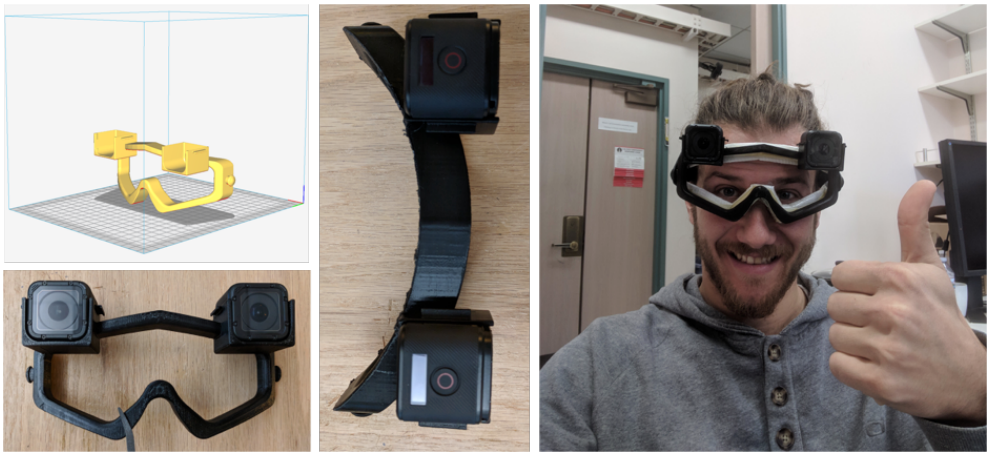


Figure 11.1: **Stereo and ego-centric video capturing rig.** The rig designed for capturing stereo ego-centric videos of human actions and social interactions.[2]

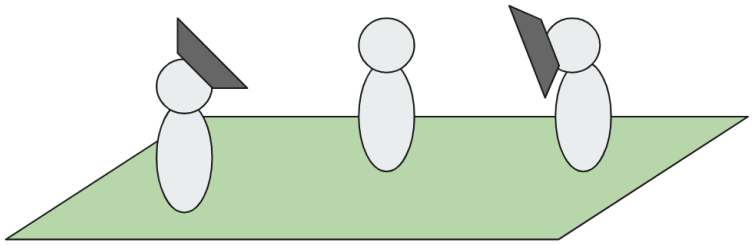


Figure 11.2: **Video capture setup for collecting a dataset of ego-centric videos of actions and social interactions.** The setup allows recording multiple human subjects, performing an action or interacting, with two or more recording rigs in relative motion to obtain valuable multi-view signal, while the stereo pair on each recording rig provides the 3D ground-truth.

We believe that verifying the robustness and generalization ability of the proposed learning algorithm on such a novel and truly *in-the-wild* video collection, could be very impactful. In fact, using a portable rig with the proposed design, one can easily obtain multi-view and stereo training videos of human actions and interactions from practically any location: on the top of mountains or under water, or in very trafficked environments, such as shopping malls, stadiums and parks.

Another research direction we are planning to pursue is studying in depth how the learned embedding can be modified to encode complex and highly articulated motions of the human body beyond its static 3D poses. In fact, similarly to how humans transition through different poses while moving, we would like to understand if and how the dynamic state of the human body can be represented by following trajectories in the learned embedding space.

[2]Thanks to my friend Pablo Guerrero for his assistance with the CAD design and 3D printing of the video capturing rig.

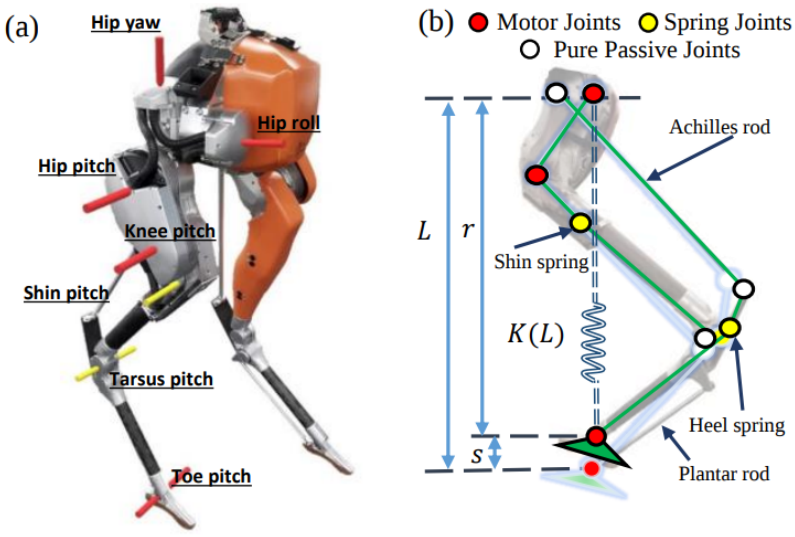


Figure 11.3: **Dynamic model of the leg of the robot Cassie.** Figure taken from [98], showing (a) the leg joints and (b) the leg model of the robot Cassie [99].

In Chapter 7 we showed that linear interpolations between vectors in the learned embedding produced realistic looking *movemes*. We would like to verify if such observation *i)* applies also to the embedding learned with the algorithm introduced in Chapter 9, which is trained using data which is easier to collect and gives better performance, *ii)* holds true for more complex non-linear trajectories and *iii)* can be extended to model longer actions which are composed of sequences of *movemes*.

In the work presented in Chapter 7 and Chapter 9, a point in the embedding space encoded a single image representing a human pose. We would like to investigate ways to encode multiple frames from consecutive timestamps into a single vector of the embedding, and analyze whether this results in an improved embedding, not only in terms of single-frame 3D pose estimation, but also for predicting human dynamics.

Finally, we would like to explore the framework in which our work can be applied for controlling the visuomotor skills of under-actuated dynamic bipedal robots that can walk and run in similar fashion to humans or animals, such as Cassie [99], Figure 11.3.

Recent work [82–84, 100] has shown that image features, extracted with convolutional neural networks, paired with self regression or reinforcement learning can be used to control manipulator robots and have them perform simple movements or object relocation tasks.

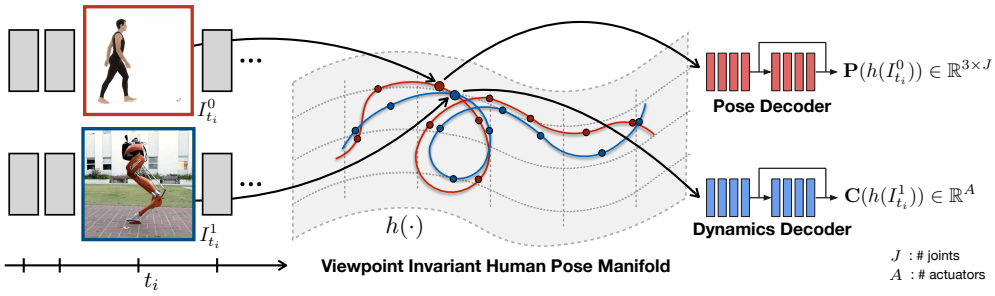


Figure 11.4: **Playing "Simon Says" in the viewpoint-invariant embedding.** First, we learn a manifold of human poses in such a way that the videos of a person performing a movement, and those of Cassie being manually controlled to imitate that movement, follow similar trajectories in the embedding space, regardless of their viewpoint. Secondly, we learn a "dynamics decoder" that can map the learned embedding vectors to Cassie's actuators' control signals. The goal is to study the extent to which the dynamics decoder can be trained with few examples and generalize to novel viewpoints and movements, as is the case for the pose decoder in the task of 3D pose estimation.

We would like to study how the methodology introduced in Chapter 9 can be expanded and applied to the highly complex case of controlling an underactuated robot imitating human body motions, like walking or squatting. This presents several new challenges, such as describing the dynamic model of the robot and dealing with instabilities due to the unknown surface contact points. While introducing complex feedback control algorithms for such types of robots is the main focus of state-of-the-art robotics research [98], we would like to investigate whether expanding the state variable with learned features, extracted from images of the robot moving successfully, can improve such control algorithms. This would add information to the state representation about the image appearance that is relevant to the movement that is being controlled, with the hope of making it more stable, secure, and efficient.

In Figure 11.4 we exemplify one possible direction of a further investigation. Similarly to how the "pose decoder" introduced in Chapter 9 is able to produce a 3D pose of the human body from the viewpoint invariant embedding space, we would like to learn a "dynamics decoder" that can output a control signal for Cassie's actuators, resulting in Cassie moving accordingly to its visual input in a way that could enable it to play the game of "Simon Says".

The embedding of Chapter 9 has the advantage of being able to retain an egocentric representation of the human body. This could allow learning more easily a unique control policy invariant to the viewpoint from which the robot is viewing the movement it is trying to imitate, while discarding other confounding factors, such as background elements, lighting and motion blur.

BIBLIOGRAPHY

[1] S. Semaw, P. Renne, J. W. Harris, C. S. Feibel, R. L. Bernor, N. Fesseha, and K. Mowbray, "2.5-million-year-old stone tools from gona, ethiopia", *Nature*, vol. 385, no. 6614, p. 333, 1997 (cit. on p. 2).

[2] *The first butchers*, `https://www.sapiens.org/evolution/homo-sapiens-and-tool-making`, Accessed: 2019-12-10 (cit. on p. 2).

[3] D. R. Harris and B. Wood, *The origins and spread of agriculture and pastoralism in eurasia*. UCL press London, 1996, vol. 744 (cit. on p. 2).

[4] G. Hillman, "Late pleistocene changes in wild plant-foods available to hunter-gatherers of the northern fertile crescent: Possible preludes to cereal cultivation.", *The origins and spread of agriculture and pastoralism in Eurasia*, pp. 159–203, 1996 (cit. on p. 2).

[5] Y.-I. Sato, "Origin of rice cultivation in the yangtze river basin", *The origins of pottery and agriculture*, pp. 143–150, 2002 (cit. on p. 2).

[6] A. Snir, D. Nadel, I. Groman-Yaroslavski, Y. Melamed, M. Sternberg, O. Bar-Yosef, and E. Weiss, "The origin of cultivation and proto-weeds, long before neolithic farming", *PLoS One*, vol. 10, no. 7, e0131422, 2015 (cit. on p. 2).

[7] V. L. Bengtson and R. Settersten Jr, *Handbook of theories of aging*. Springer Publishing Company, 2016 (cit. on p. 2).

[8] M. Gurven and H. Kaplan, "Longevity among hunter-gatherers: A cross-cultural examination", *Population and Development review*, vol. 33, no. 2, pp. 321–365, 2007 (cit. on p. 2).

[9] R. E. Sojka, D. L. Bjorneberg, and J. Entry, "Irrigation: An historical perspective", 2002 (cit. on p. 2).

[10] T. Freeth, Y. Bitsakis, X. Moussas, J. H. Seiradakis, A. Tselikas, H. Mangou, M. Zafeiropoulou, R. Hadland, D. Bate, A. Ramsey, *et al.*, "Decoding the ancient greek astronomical calculator known as the antikythera mechanism", *Nature*, vol. 444, no. 7119, p. 587, 2006 (cit. on p. 2).

[11] H. Alessandrino and B. Baldi, "De gli automati, overo, machine se moventi libri due", (cit. on p. 2).

[12] M. E. Moran, "The da vinci robot", *Journal of endourology*, vol. 20, no. 12, pp. 986–990, 2006 (cit. on p. 2).

[13] M. Rosheim, *Leonardo's lost robots*.
Springer Science & Business Media, 2006 (cit. on p. 2).

[14] Y. N. Harari, *Homo deus: A brief history of tomorrow*. Random House, 2016 (cit. on p. 3).

[15] J. K. Mitchell, "The last of a veteran chess player",
Chess Monthly, vol. 1, pp. 3–4, 1857 (cit. on p. 2).

[16] C. Bailly, *Automata: The golden age: 1848-1914*.
Robert Hale Limited, 2003 (cit. on p. 3).

[17] M. Krzyzaniak, "Prehistory of musical robots",
Journal of Human-Robot Interaction, vol. 1, no. 1, pp. 78–95, 2012
(cit. on p. 3).

[18] *1865 - le petit journal*, `https://gallica.bnf.fr/ark:/12148/bpt6k589123j.image.r=Manzetti.f3.langEN`,
Accessed: 2019-12-4 (cit. on p. 3).

[19] *1849 flute-playing automaton innocenzo manzetti (italian)*,
`http://cyberneticzoo.com/robots/1849-flute-playing-automaton-innocenzo-manzetti-italian`,
Accessed: 2019-12-4 (cit. on p. 3).

[20] K. Capek, *Rur (rossum's universal robots)*. Penguin, 2004 (cit. on p. 3).

[21] *Invention and meaning of the word "robot"*,
`https://web.archive.org/web/20120204135259/http://capek.misto.cz/english/robot.html`, Accessed: 2019-12-4
(cit. on p. 3).

[22] I. Asimov and L. McKeever, *The complete robot*.
Doubleday New York, 1982 (cit. on p. 3).

[23] I. Asimov, *Robot dreams*. Wiley Online Library, 2001 (cit. on p. 3).

[24] ——, *I, robot*. Spectra, 2004, vol. 1 (cit. on p. 3).

[25] ——, *Robot visions*. ibooks, 2013 (cit. on p. 3).

[26] A. Deed, *The mechanical man*,
`https://www.imdb.com/title/tt0337377/`, 1921 (cit. on p. 3).

[27] S. Kubrick, *2001: A space odyssey*,
`https://www.imdb.com/title/tt0062622/`, 1968 (cit. on p. 3).

[28] J. Cameron, *The terminator*,
`https://www.imdb.com/title/tt0088247/`, 1984 (cit. on p. 3).

[29] A. Proyas, *I, robot*,
`https://www.imdb.com/title/tt0343818/`, 2004
(cit. on pp. 3, 5).

[30] I. O. for Standardization, "Iso 8373: 2012 (en): Robots and robotic devices—vocabulary", 2012 (cit. on p. 4).

[31] C. Breazeal, K. Dautenhahn, and T. Kanda, "Social robotics", in *Springer handbook of robotics*, Springer, 2016, pp. 1935–1972 (cit. on pp. 4, 5).

[32] E. B. Goldstein, *Sensation and perception*. Cengage Learning, 2009 (cit. on p. 4).

[33] L. Thaler and M. A. Goodale, "Echolocation in humans: An overview", *Wiley Interdisciplinary Reviews: Cognitive Science*, vol. 7, no. 6, pp. 382–393, 2016 (cit. on p. 4).

[34] K. Weir, "The dawn of social robots", *Monitor on Psychology*, vol. 49, no. 1, p. 50, 2018 (cit. on p. 5).

[35] R. Held *et al.*, *Perception: Mechanisms and models: Readings from scientific american*. WH Freeman, 1972 (cit. on p. 5).

[36] J. Bruner, *A study of thinking*. Routledge, 2017 (cit. on p. 5).

[37] D. C. Dennett, *Consciousness explained*. Penguin uk, 1993 (cit. on p. 6).

[38] G. Mackie and P. Burighel, "The nervous system in adult tunicates: Current research directions", *Canadian journal of zoology*, vol. 83, no. 1, pp. 151–183, 2005 (cit. on p. 6).

[39] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey", *ArXiv preprint arXiv:1905.05055*, 2019 (cit. on p. 6).

[40] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features", in *European conference on computer vision*, Springer, 2006, pp. 404–417 (cit. on p. 7).

[41] D. G. Lowe *et al.*, "Object recognition from local scale-invariant features.", in *Iccv*, vol. 99, 1999, pp. 1150–1157 (cit. on p. 7).

[42] P. Viola, M. Jones, *et al.*, "Rapid object detection using a boosted cascade of simple features", (cit. on p. 7).

[43] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection", 2005 (cit. on p. 7).

[44] P. Dollár, S. Belongie, and P. Perona, "The fastest pedestrian detector in the west", 2010 (cit. on p. 7).

[45] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection", *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 8, pp. 1532–1545, 2014 (cit. on p. 7).

[46] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models", *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2009 (cit. on pp. 7, 8).

[47] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks", in *Advances in neural information processing systems*, 2012, pp. 1097–1105 (cit. on pp. 7, 13).

[48] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database", in *2009 IEEE conference on computer vision and pattern recognition*, Ieee, 2009, pp. 248–255 (cit. on pp. 7, 13).

[49] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context", in *Computer Vision–ECCV 2014*, Springer, 2014, pp. 740–755 (cit. on pp. 7, 8, 10).

[50] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, *The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results*, http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html (cit. on p. 7).

[51] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587 (cit. on p. 7).

[52] R. Girshick, "Fast r-cnn", in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448 (cit. on p. 7).

[53] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks", in *Advances in neural information processing systems*, 2015, pp. 91–99 (cit. on p. 7).

[54] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn", in *ICCV*, 2017 (cit. on p. 7).

[55] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788 (cit. on p. 7).

[56] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector", in *European Conference on Computer Vision*, Springer, 2016, pp. 21–37 (cit. on p. 7).

[57] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection", in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988 (cit. on p. 7).

[58] G. Johansson, "Visual perception of biological motion and a model for its analysis", *Perception & psychophysics*, vol. 14, no. 2, pp. 201–211, 1973 (cit. on p. 8).

[59] M. A. Fischler and R. A. Elschlager, "The representation and matching of pictorial structures", *IEEE Transactions on computers*, no. 1, pp. 67–92, 1973 (cit. on p. 8).

[60] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition", *International journal of computer vision*, vol. 61, no. 1, pp. 55–79, 2005 (cit. on p. 8).

[61] Y. Yang and D. Ramanan, "Articulated pose estimation with flexible mixtures-of-parts", in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, IEEE, 2011, pp. 1385–1392 (cit. on p. 8).

[62] S. Zuffi, O. Freifeld, and M. J. Black, "From pictorial structures to deformable structures", in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2012, pp. 3546–3553 (cit. on p. 8).

[63] X. Chen and A. L. Yuille, "Articulated pose estimation by a graphical model with image dependent pairwise relations", in *Advances in Neural Information Processing Systems*, 2014, pp. 1736–1744 (cit. on p. 8).

[64] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2d human pose estimation: New benchmark and state of the art analysis", in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2014 (cit. on p. 8).

[65] S. Johnson and M. Everingham, "Clustered pose and nonlinear appearance models for human pose estimation", in *Proceedings of the British Machine Vision Conference*, doi:10.5244/C.24.12, 2010 (cit. on p. 8).

[66] A. Toshev and C. Szegedy,
"Deeppose: Human pose estimation via deep neural networks",
in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1653–1660 (cit. on p. 8).

[67] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy, "Towards accurate multi-person pose estimation in the wild", *ArXiv preprint arXiv:1701.01779*, 2017 (cit. on p. 8).

[68] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh,
"Convolutional pose machines", in *CVPR*, 2016 (cit. on p. 8).

[69] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation", *ArXiv preprint arXiv:1603.06937*, 2016 (cit. on p. 8).

[70] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields",
ArXiv preprint arXiv:1611.08050, 2016 (cit. on p. 8).

[71] S. Kreiss, L. Bertoni, and A. Alahi,
"Pifpaf: Composite fields for human pose estimation", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019,
pp. 11 977–11 986 (cit. on p. 8).

[72] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments", *PAMI*, 2014 (cit. on p. 8).

[73] G. Varol, J. Romero, X. Martin, N. Mahmood, M. J. Black, I. Laptev, and C. Schmid, "Learning from synthetic humans", in *CVPR*, 2017 (cit. on p. 8).

[74] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model",
ACM transactions on graphics (TOG), 2015 (cit. on p. 8).

[75] S. Carey, *The origin of concepts*. Oxford University Press, 2009
(cit. on p. 8).

[76] R. Jain, R. Kasturi, and B. G. Schunck, *Machine vision*, vol. 5 (cit. on p. 8).

[77] P. W. Battaglia, J. B. Hamrick, and J. B. Tenenbaum, "Simulation as an engine of physical scene understanding", *Proceedings of the National Academy of Sciences*, vol. 110, no. 45, pp. 18 327–18 332, 2013
(cit. on p. 9).

[78] A. Lerer, S. Gross, and R. Fergus, "Learning physical intuition of block towers by example", *ArXiv preprint arXiv:1603.01312*, 2016 (cit. on p. 9).

[79] J. He, A. Lehrmann, J. Marino, G. Mori, and L. Sigal,
"Probabilistic video generation using holistic attribute control",
in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 452–467 (cit. on p. 9).

[80] C. Vondrick, H. Pirsiavash, and A. Torralba,
"Generating videos with scene dynamics",
in *Advances In Neural Information Processing Systems*, 2016, pp. 613–621 (cit. on p. 9).

[81] T. Xue, J. Wu, K. Bouman, and B. Freeman, "Visual dynamics: Probabilistic future frame synthesis via cross convolutional networks",
in *Advances in neural information processing systems*, 2016, pp. 91–99 (cit. on p. 9).

[82] P. Agrawal, A. V. Nair, P. Abbeel, J. Malik, and S. Levine,
"Learning to poke by poking: Experiential learning of intuitive physics",
in *Advances in Neural Information Processing Systems*, 2016,
pp. 5074–5082 (cit. on pp. 9, 173).

[83] C. Finn, X. Y. Tan, Y. Duan, T. Darrell, S. Levine, and P. Abbeel,
"Deep spatial autoencoders for visuomotor learning",
in *2016 IEEE International Conference on Robotics and Automation (ICRA)*,
IEEE, 2016, pp. 512–519 (cit. on pp. 9, 173).

[84] C. Finn, I. Goodfellow, and S. Levine,
"Unsupervised learning for physical interaction through video prediction",
in *Advances in Neural Information Processing Systems 29*,
D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds.,
Curran Associates, Inc., 2016, pp. 64–72.
[Online]. Available: `http://papers.nips.cc/paper/6161-unsupervised-learning-for-physical-interaction-through-video-prediction.pdf` (cit. on pp. 9, 173).

[85] J. Martinez, M. J. Black, and J. Romero,
"On human motion prediction using recurrent neural networks",
in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2891–2900 (cit. on p. 9).

[86] D. Pavllo, D. Grangier, and M. Auli, "Quaternet: A quaternion-based recurrent model for human motion",
ArXiv preprint arXiv:1805.06485, 2018 (cit. on p. 9).

[87] S. Gidaris and N. Komodakis, "Object detection via a multi-region and semantic segmentation-aware cnn model",
in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1134–1142 (cit. on p. 10).

[88] X. Zeng, W. Ouyang, J. Yan, H. Li, T. Xiao, K. Wang, Y. Liu, Y. Zhou, B. Yang, Z. Wang, *et al.*, "Crafting gbd-net for object detection",
IEEE transactions on pattern analysis and machine intelligence, vol. 40, no. 9, pp. 2109–2123, 2017 (cit. on p. 10).

[89] V. Ramakrishna, D. Munoz, M. Hebert, J. A. Bagnell, and Y. Sheikh, "Pose machines: Articulated pose estimation via inference machines", in *European Conference on Computer Vision*, Springer, 2014, pp. 33–47 (cit. on p. 10).

[90] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using convolutional networks", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 648–656 (cit. on p. 10).

[91] Z. Tu and X. Bai, "Auto-context and its application to high-level vision tasks and 3d brain image segmentation", *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 10, pp. 1744–1757, 2009 (cit. on p. 10).

[92] X. Chu, W. Yang, W. Ouyang, C. Ma, A. L. Yuille, and X. Wang, "Multi-context attention for human pose estimation", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1831–1840 (cit. on p. 10).

[93] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks", in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497 (cit. on p. 10).

[94] C.-Y. Wu, C. Feichtenhofer, H. Fan, K. He, P. Krahenbuhl, and R. Girshick, "Long-term feature banks for detailed video understanding", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 284–293 (cit. on p. 10).

[95] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories", in *2004 conference on computer vision and pattern recognition workshop*, IEEE, 2004, pp. 178–178 (cit. on p. 13).

[96] G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category dataset", 2007 (cit. on p. 13).

[97] Y. LeCun, Y. Bengio, *et al.*, "Convolutional networks for images, speech, and time series", *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995 (cit. on p. 13).

[98] X. Xiong and A. D. Ames, "Coupling reduced order models via feedback control for 3d underactuated bipedal robotic walking", in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, IEEE, 2018, pp. 1–9 (cit. on pp. 173, 174).

[99] *The robot cassie*, `https://www.agilityrobotics.com/robots#cassie`, Accessed: 2019-12-11 (cit. on p. 173).

[100] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, and S. Levine, “Time-contrastive networks: Self-supervised learning from pixels”, 2017 (cit. on p. 173).