Looking Deep at People: Towards Understanding and Generating Humans in Images with Deep Learning

Rodrigo Andrade de Bem

Wolfson College
University of Oxford

A thesis submitted for the degree of Doctor of Philosophy

Michaelmas 2018

Abstract

Understanding and generating people in images and videos is a long-standing goal in computer vision. A significant effort has been devoted to these tasks by the research community along the last decades, greatly motivated by a large number of potential applications, like surveillance, human-machine interaction, action and behaviour recognition, motion capture, video reenactment, and computer graphics animation. Also driving the necessity of this remarkable endeavour, one can mention the difficulties for tackling such problems, generated for instance by the endless combinations of environments, visual appearances, and postures in which humans can appear in images. Besides that, the high-dimensionality of the human body, the inherent noise of visual data and the ill-posed characteristics of the problems are also relevant issues. Nonetheless, meaningful advances in the field were achieved recently using deep learning.

This thesis pursues further advances towards understanding and generating people in visual data by the development of new discriminative and generative deep learning methods. The main contributions are:

i) A deep learning framework for 2D human pose estimation, which allows for mean-field inference over part-based models;

ii) A conditional deep generative model that achieves state-of-the-art results on generating images of humans conditioned on body posture; and

iii) A structured semi-supervised deep generative model that jointly performs pose estimation and image generation, *understanding* and *generating* people in images in a single framework.

Looking Deep at People: Towards Understanding and Generating Humans in Images with Deep Learning

Rodrigo Andrade de Bem

Wolfson College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Michaelmas 2018

To my wife and my daughter,
Caroline and Manuela,
for their love and patience.

Acknowledgements

My work on this thesis was supported by several people that were essential from before the beginning until the very end. First things first! I want to start thanking all my family, relatives, and friends but in particular my parents, Raul and Anelita, for their unconditional love and encouragement since my very early years. Likewise, my brother Eduardo and my sister Daniela, for being more than siblings but close friends, always present whenever I needed. To my wife and partner in crime Caroline, I am not sure what else to say, other than acknowledge that this really would not be possible without you. To my little pirate Manuela, that hopefully will read this in the future, I would like to say that watching you grow gracefully (much more gracefully than this thesis) during these four years was nothing but overwhelmingly joyful.

Thanks to my former supervisors, Profs. Marcelo Pias, Silvia Botelho, and Anna Reali, for their confidence to recommend me as a DPhil candidate; to my friends from FURG for their support; and to FURG and CAPES as institutions for their sponsorship. My thanks too to Profs. Andrew Zisserman, Nando de Freitas, and David Murray for their valuable contributions during my Transfer and Confirmation of Status examinations. Especially, I am immensely thankful and will be forever grateful for the opportunity granted to me by my supervisor Prof. Philip Torr. It was really a once-in-a-lifetime experience!

I am glad for my time in the TVG group. It was a pleasure to work and spend time with all you guys. In particular, my thanks to Stuart Golodetz, Michael Sapienza, Sadeep Jayasumana, Bernardino Romera-Paredes, Siddharth Narayanaswamy, Thalaiyasingam Ajanthan, Adnane Boukhayma, Saumya Jetley, Luca Bertinetto, Anurag Arnab, Arnab Ghosh, Oscar Rahnama, Daniela Massiceti, Ondrej Miksik, Alban Desmaison, Tom Joy, Viveka Kulharia, Qizhu Li, Christian Schroeder, Nick Lord, Stephan Liwicki, Jack Valmadre, Puneet Dokania, Tommaso Cavallari, Shuai Zheng, Julien Valentin, Vibhav Vineet, Harkirat Behl, Nantas Nardelli, Arslan Chaudhry, Namhoon Lee, Yuge Shi, for all support, guidance and companionship. My special thanks to Justin Hutchence for his incredible assistance.

Last but not least, thanks to all the dear friends, especially from Wolfson, HBC and St. Andrew's, which made our time in Oxford more special. In particular, Fabio and Natalia; Leonardo and Flávia; Zohar and Avigail; Pedro and Talita; Alan and Leanne; Paulo and Estela; Adam Mahdi; Aline and Alexandre; Ruth and Ricardo; James and Marion; Debbie and Adam; Ana and David; Issac and Gillian; Charles and Yvonne; Nilton and Fabiana; Charles and Val. We will really miss you all guys!

Abstract

Understanding and generating people in images and videos is a long-standing goal in computer vision. A significant effort has been devoted to these tasks by the research community along the last decades, greatly motivated by a large number of potential applications, like surveillance, human-machine interaction, action and behaviour recognition, motion capture, video reenactment, and computer graphics animation. Also driving the necessity of this remarkable endeavour, one can mention the difficulties for tackling such problems, generated for instance by the endless combinations of environments, visual appearances, and postures in which humans can appear in images. Besides that, the high-dimensionality of the human body, the inherent noise of visual data and the ill-posed characteristics of the problems are also relevant issues. Nonetheless, meaningful advances in the field were achieved recently using deep learning.

This thesis pursues further advances towards understanding and generating people in visual data by the development of new discriminative and generative deep learning methods. The main contributions are:

i) A deep learning framework for 2D human pose estimation, which allows for mean-field inference over part-based models;

ii) A conditional deep generative model that achieves state-of-the-art results on generating images of humans conditioned on body posture; and

iii) A structured semi-supervised deep generative model that jointly performs pose estimation and image generation, *understanding* and *generating* people in images in a single framework.

Contents

List of Figures

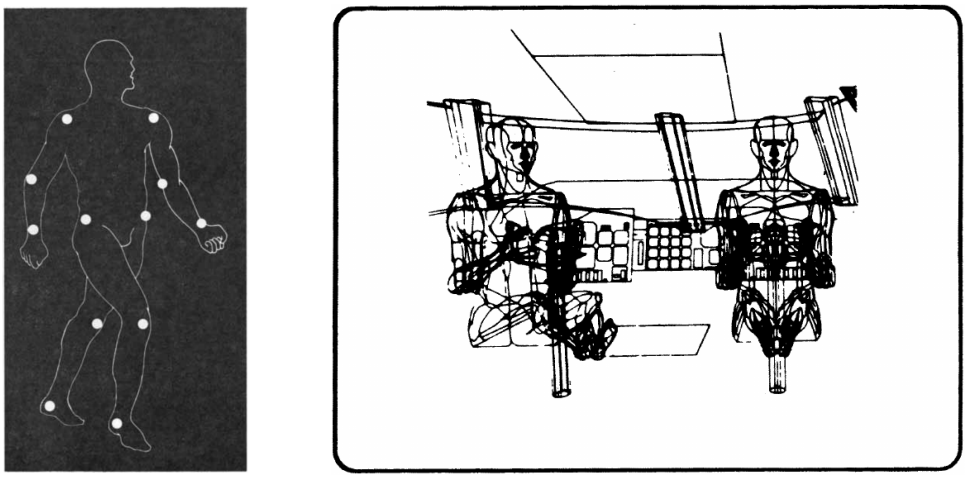
List of Tables

1 Introduction

Visual analysis of humans is a long-standing goal in computer vision, being one of the most active areas in the field for the last decades (Moeslund et al., 2011). It covers problems such as detection and tracking of people, facial and gestural analysis, and action recognition.

Among them, human pose estimation is one of the most investigated subjects (Moeslund et al., 2001; Moeslund et al., 2006). Such a problem, which goes back to Johansson's moving light display (MLD) (Johansson, 1973; Johansson, 1975), illustrated in Fig. 1.1a, is sometimes referred to in the computer vision literature as articulated pose estimation or body pose recovery (Sigal, 2014). It is the process of estimating the configuration of the human body from visual data. On it, the body is considered as an articulated object, composed of several connected rigid parts. Its configuration denotes the position of these components in the 2D (image plane) or in the 3D (world) space.

Another related topic actively investigated is the generation of visual data containing people. It corresponds to the process of synthesising images of humans, taking into account attributes such as body posture, appearance, shape, illumination, and the surrounding environment. Ideally, considering all these factors, a model should be capable of generating realistic image samples of people in real-world scenarios. Such creation of *virtual humans* (Hilton et al., 1999; Magnenat-Thalmann



(a) (b)

Figure 1.1: Pioneering approaches to understanding and generating humans. **(a)** Illustration of the moving light display (MLD) from Johansson (1975). In one experiment, two dancers perform in a dark room with 12 lights "outlining" their bodies. According to Johansson (1975), subjects watching the film could tell in a "fraction of a second" that two people were moving in the scene. **(b)** Boeing's "first man" illustration (Fetter, 1982). This articulated synthetic human with seven segments was developed and used for the studies of the Boeing 747 instrument panel. Created in 1968, it was the first human figure build using computer graphic techniques.

et al., 2006) is traditionally a computer graphics undertake (Magnenat-Thalmann et al., 2005) since its origins in the 60's, with Boeing's "first man" (Boeing, 2018; Fetter, 1982), shown in Fig. 1.1b. However, it has been gradually embraced also by the computer vision community, since image-based techniques have been successfully adopted on matters like rendering and modelling (Kanade et al., 1997; Borshukov et al., 2005; Starck et al., 2007).

In this thesis, we investigate the *understanding* and the *generation* of humans in images. We associate *understanding* with the typical computer vision task of extracting information from visual data by mapping images to abstract representations, e.g., points, bounding boxes, skeletons, and segmentation masks. It is a relatively broad and subjective term since its meaning may vary according to the task or context that someone has in mind. For instance, the 2D positions of people in the ground may be sufficient information to "understand" the movement dynamics of pedestrians in a crossroad, however certainly not enough to infer their emotions. On the other hand, we relate *generation* to the analysis-by-synthesis

process of reproducing natural images from a set of parameters, e.g., illumination, shading, and camera position, which presents closer bonds with the computer graphics enterprise. Here, the meaning of the expression seems to be reasonably clear, perhaps because generating digital pictures and videos of people has become a pervasive activity in our daily lives.

Arguably, the human pose plays a central role regarding both issues, in a sense tying them together. Firstly, pose is highly correlated to semantically complex understanding aspects, such as performed activities and presented behaviours (Pantic et al., 2006; Poppe, 2010; Jhuang et al., 2013). Secondly, regarding the generation of realistic images, although visual appearance certainly has a fundamental relevance, the body posture is again essential since suitable postural configurations are crucial to grant fidelity to virtual humans (Akhter et al., 2015). Therefore, body pose as an attribute of humans in images has centrality in the present work, in which we research both *understanding* and *generation* under the light of deep learning methods (Goodfellow et al., 2016).

1.1 Motivation and Objectives

There are a high number of applications in several areas for understanding and generating people in natural images. Following we highlight some of them, referring to representative academic and commercial approaches when appropriate.

Movies, Entertainment and News. Some of the most popular applications are related to the special effects industry, which is responsible for forging human-like characters that are watched by millions of people in the movies. A well-known landmark in the area was the use of image-based rendering techniques on the Matrix trilogy (Borshukov et al., 2005). Additionally, companies such as Vicon (2018) and OptiTrack (2018) have developed robust commercial solutions for applications like motion capture, illustrated in Fig. 1.2. It is an excellent example of a task for mutual understanding and generation since estimated poses are transferred to animated characters. However, with few exceptions (Organic Motion, 2006;

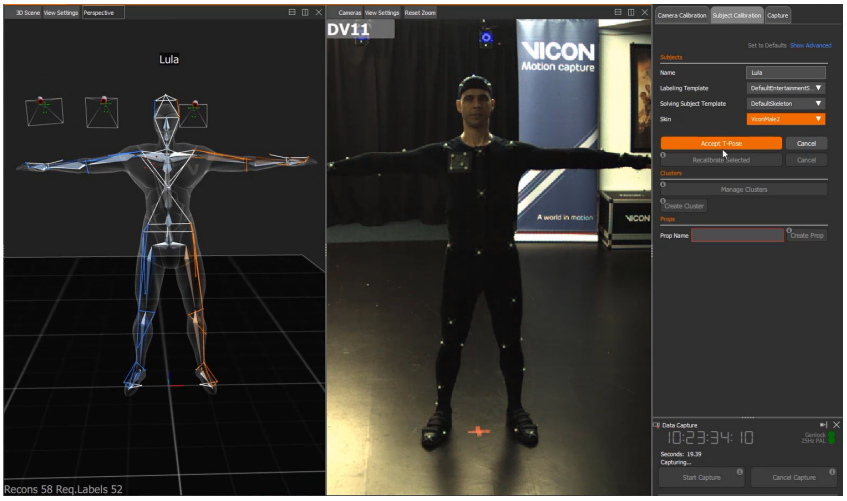
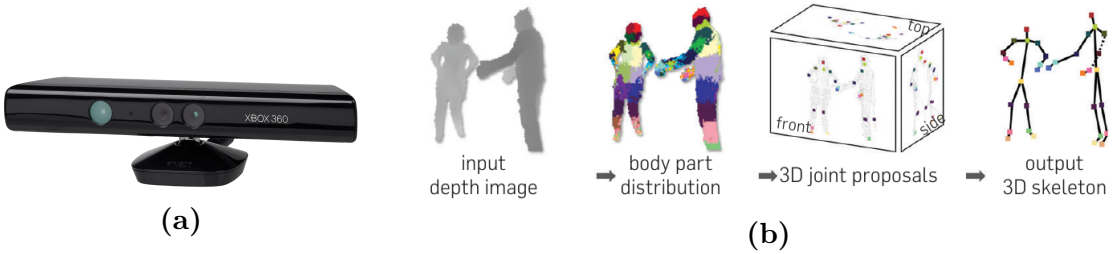


Figure 1.2: Illustration of a Vicon's multi-camera motion capture system (Vicon Motion Systems, 2018), frequently employed in movies' productions. The snapshot of the controlling software interface shows a person in the centre wearing a suit containing several optical markers which are tracked by the multiple cameras. In the left, we see the correspondent tracked body model. Such kind of multi-camera marker-based motion capture systems still are the most accurate ones currently available.

The Captury, 2018), most of the commercial solutions depend upon markers over the actor's bodies and manually developed virtual characters (3Lateral, 2018) to achieve the desired level of fidelity.

Other companies like SenseTime (2018) have ported body pose estimation and motion capture solutions to mobile platforms, focusing on entertainment solutions which are very flexible at the cost of accuracy and precision. Very recently, Xinhua News Agency (2018) launched a synthetic news anchor based on a machine learning solution not disclosed by the developers (Sogou, 2017). Although still performed in a constrained environment, such accomplishment sheds light over the potential possibilities for such methods, including eventual dangers. For instance, methods that can perform RGB video reenactment (Thies et al., 2016; Balakrishnan et al., 2018) have shown to be capable of generating synthetics videos used to spread detrimental fake news. Certainly, ethical discussing and regulations will need to address such issues soon.

Video Games. This multi-billion-dollar industry (CNBC, 2018) has importantly contributed to advances in the field, either by the creation of software such as



(a) (b)

Figure 1.3: Popularisation of affordable depth-cameras. (a) The second edition of the iconic Kinect, part of the Xbox 360 video game console, and now discontinued by Microsoft. It was greatly responsible for the increase in the use of depth-based methods in the computer vision community. **(b)** Overview of the relevant real-time 3D pose estimation method by Shotton et al. (2013), applied over single Kinect depth images and based on random forests classifiers. The approach highlights the importance of machine learning and large training datasets even before the currently widespread deep learning algorithms.

powerful game engines (Epic Games, 2018), or by the development of hardware devices such as graphics cards (NVIDIA, 2018), the current powerhouse of deep learning, and the Microsoft Kinect (Microsoft, 2018a) which consolidate the popularisation of affordable depth-cameras. The Kinect (see Fig. 1.3a), initially designed for a video game console, also had a significant impact on the research community particularly regarding the use of machine learning techniques for pose estimation, as exemplified in Fig. 1.3b (Shotton et al., 2011; Shotton et al., 2013). Regarding the generation of content, although the level of fidelity of synthetic images in the area is already impressive, it is still highly dependent upon the work of computer graphics artists (3Lateral, 2018). To turn such a process into a cheaper and less time-consuming task, it is desirable to improve the automatic generation of people in images, making them more realistic.

Robotics. The interaction between robots and humans has been the object of study for a long time in the robotics community (Goodrich et al., 2007). However, as robots increasingly become part of our daily lives, such topic gains even more relevance. To mention two applications, one could refer to autonomous cars and unmanned aerial vehicles (UAVs) or drones, both illustrated in Fig. 1.4. In both cases, it is not only essential to understand and anticipate the behaviour of people on the surroundings of the robots (Ohn-Bar et al., 2016; Schaffalitzky, 2016; Singh

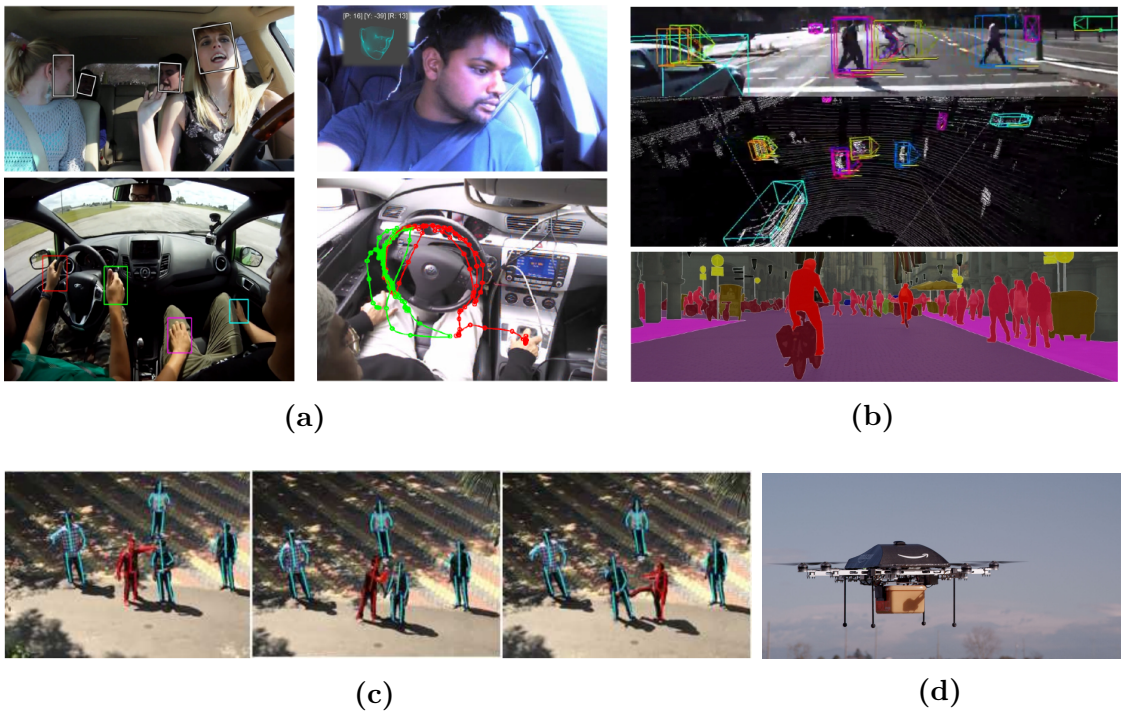


Figure 1.4: Applications in robotics. Ohn-Bar et al. (2016) propose a taxonomy of problems related to autonomous cars, in which humans are inside the vehicle, around it, or into surrounding vehicles. The authors use images from the VIVA dataset (LISA, 2015) to illustrate the scenario inside the vehicle in **(a)**, with hands and faces being detected and tracked. In **(b)**, the top and bottom images, respectively from the KITTI (Geiger et al., 2013) and Cityscapes (Cordts et al., 2016) datasets, show people around the car, either cyclists or pedestrians. **(c)** Pose estimation over an image captured by a drone (Singh et al., 2018). It is also an important application since delivery and retail companies have ambitious projects, like the Amazon PrimeAir (Amazon, 2018b), involving UAVs, such as the one illustrated in **(d)**, to directly deliver parcels to costumers.

et al., 2018; Amazon, 2018b), but also the capability of generating realistic images of humans, for instance, pedestrians in urban environments, is highly required to allow the creation of "virtual worlds" for simulation and training (Peters et al., 2009; Massive Software, 2017; Wang et al., 2018).

Assistive and Medical Technologies. The ageing of populations in many countries around the world is one of the reasons that assistive technologies have become increasingly important throughout the years. Monitoring older adults (Obdrzalek et al., 2012) and also hospital patients (Achilles et al., 2016) remotely, as demonstrated in Fig. 1.5a, is one of the critical applications in this area. Regarding image generation, some companies have focused on tools which allow patients to

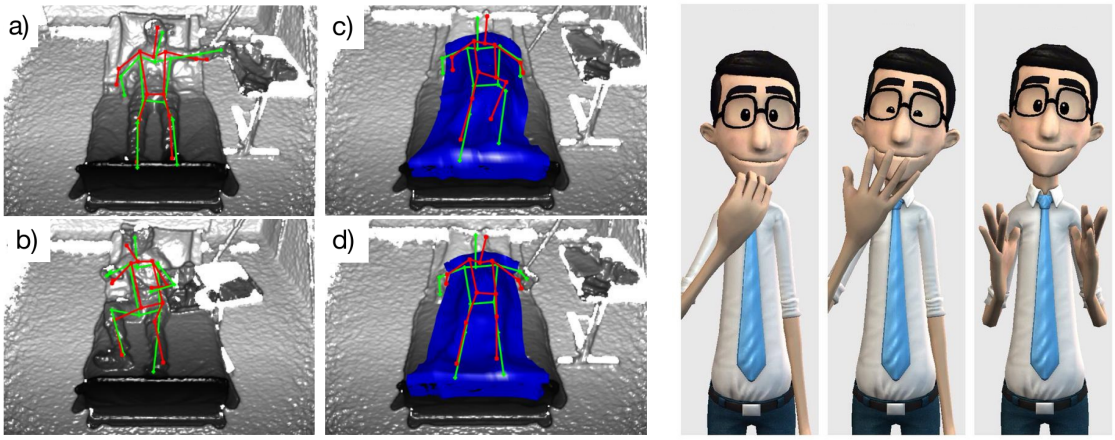


Figure 1.5: Illustration of assistive and medical applications. **(a)** Pose estimation is used to monitor patients in bed even when they are covered by a blanket (blue surface in images on the right) (Achilles et al., 2016). **(b)** The Hand Talk (2018) company has a commercial product in which text is automatically translated into LIBRAS, the Brazilian sign language, by a synthetic avatar.

preview the results of medical procedures, such as breast surgeries (Arbrea Labs, 2018). Despite the existence of advanced approaches for these and other problems in the areas, further developments are still needed, such as hand-tracking systems (Prisacariu et al., 2011) for translating of sign languages (see Fig. 1.5b) (Hand Talk, 2018; University of Surrey, 2018), and support for visually impaired to cope with people's behaviours around them (OxSight, 2018). For instance, in the United Kingdom, there are approximately 14 million people reported with some disability, making these kinds of technologies social and economically important (Department for Work and Pensions - GOV.UK, 2017).

Fashion. Some reports estimate the value of the global fashion industry at 3 trillion dollars (Fashion United, 2018). Every year, an increasing portion of this market is done through e-commerce (McKinsey&Company, 2017; McKinsey&Company, 2018), creating the necessity for better ways of matching outfits to the personal tastes and sizes of the consumers. Big retail companies, such as Amazon (2018a) and Alibaba (2018) are already investing in equipment and systems that work as personal stylists, detecting appropriate sizes, providing previews and even given

(a) **(b)**

Figure 1.6: (a) The Echo Look (Amazon, 2018a) is a camera that captures a picture of someones' outfit and provides personal style assistance. **(b)** From left to right, Margot, Shudu, and Zhi are the Balmain's computer graphics virtual supermodels (Balmain, 2018).

recommendations for buyers. See the Amazon Echo Look in Fig. 1.6a. In other cases, some fashion brands have invested in virtual supermodels to showcase their collections, as illustrated in Fig. 1.6b (Balmain, 2018). Finally, researchers have also explored the area employing machine learning to related problems and showing that further developments are still needed (Lassner et al., 2017).

Besides all the highlighted applications, and still not exhaustively, one could mention others like sports (Hilton et al., 2011), virtual/augmented/mixed realities (Facebook, 2018; Microsoft, 2018b), security surveillance (Haritaoglu et al., 2000), and military (Zyda et al., 2005). Thus, pursuing the *understanding* and *generation* of humans in natural images presents a high practical relevance concerning its applicability to real-world problems. We define the main objectives of the present thesis in the context of deep learning methodologies as:

1) to seek for the fundamentals of *understanding* people in visual data through 2D pose estimation in natural still images;

2) to pursue the *generation* of realistic natural still images of people; and

3) to aim for a comprehensive framework capable of jointly and simultaneously perform both previous tasks, *understanding* and *generation*, being not only capable

of mapping images to abstract representations (understanding) but also capable of mapping these representations back to the image-space (generation).

1.2 Challenges

The difficulties in solving the human pose estimation problem stem from diverse sources. Firstly, the human body itself, modelled as an articulated object, is a high-dimensional structure (often ranging from 30 to 70 dimensions, depending on the modelling choices) with many degrees of freedom (Sigal, 2014). To search for plausible configurations of it in this high-dimensional space is costly and often an ill-posed problem (Sminchisescu, 2006). The nature of visual data also presents inherent issues that make the task even more complicated. Noise in the measurements, effects of lighting, motion, manufacturing imprecision in the equipment and in the calibration processes, and the projection from the 3D world to the 2D image plane are sources of uncertainties and ambiguities which must be overcome. Lastly, another formidable challenge arises from a massive variety of sizes, shapes, appearances, and poses which the human body can assume. It also can be inserted into an endless number of different scenes, from chroma key studios to extremely cluttered environments.

Indeed, to tackle human pose estimation, constrained scenarios (or setups) are often established to facilitate the solution of the problem. For instance, one may consider the use of multiple cameras, indoor studios, artificial markers to identify body parts, and RGB-D cameras as ways to simplify the most general scenario, which is the markerless pose estimation[1] in a monocular RGB image. Although solving human pose estimation from monocular images is very difficult, there are relevant reasons that justify the focus on it. In many cases, only images from single RGB cameras are available, for instance in movies, still camera pictures, and TV or Internet videos. Depending on the application, the requirement of studios, special clothes or markers is inconvenient and sometimes impossible to arrange. In the

[1]From this point on, for simplicity, we will use the terms markerless pose estimation and pose estimation without distinction in the text.

analysis of sports performance, for example, the use of intrusive strategies can affect the execution of the athletes' movements. In other situations, the imposition of constraints and intrusions can be difficult or undesirable, such as in outdoor environments or controlled places, as medical facilities. Besides that, on robots such as autonomous cars, the use of single RGB sensors dramatically simplifies and reduces the cost of the data acquisition process, not requiring steps such as simultaneous calibration, image registration or synchronisation.

Lastly, the advent of low-cost RGB-D cameras certainly opened another track for research in the field. However, the environments in which such equipment can be used are restricted. The main issues are related to the sensors' range and sensitivity to lighting conditions, for instance, some RGB-D cameras are sensitive to infrared from natural light. Although there are relevant applications for all simplified scenarios, the markerless human pose estimation in monocular RGB images is still the most general one, with the broadest range of applications.

Concerning the generation of realistic RGB images, some of the challenges mentioned above are the same, such as the ill-posed 3D to 2D projections. However, instead of focusing on the "mapping" of images to abstract representations, such as body pose, it involves the explicit or implicit modelling of all the geometric and photometric intricacies of digital images formation. Such complexities are well studied and mastered in computer graphics but still mainly performed with some level of human intervention. To automatically synthesise artificial images to the point they cannot be distinguished from real ones may be considered as equivalent to succeed in a *visual Turing test* (Shan et al., 2013). Thus, a considerably complicated and consequently yet unsolved challenge (Fan et al., 2018). To perform such a task producing images of people is even harder since humans seem to be very familiarised to corporal traits, like faces, since their early ages (Valenza et al., 1996).

To generate an image of a person in a given pose, for instance, a method needs to take into account aspects such as appearance, shape, imaging process, illumination, and the surrounding environment. Ideally, considering all these factors, a generative model should be capable of generating realistic image samples of people

in real-world scenarios. It is clear that the definition of such models is highly complex and hard to extend to unconstrained images, due to intractable probability distributions and the high variability of all the aspects above. Although the literature presents detailed models (Franco et al., 2005; de La Gorce et al., 2011), in many cases simplifying assumptions are made in practice, such as independence between different factors. Such simplifications frequently lead to weak generative models that fail to capture statistical subtleties.

1.3 Approach

Regarding the understanding of people in images through 2D pose estimation, since the pioneering work of Johansson, the moving light display (MLD) proposed in the 70s (Johansson, 1973; Johansson, 1975), important advances were accomplished in the last four decades of scientific investigation. The first generation of research efforts, described in several surveys (Aggarwal et al., 1994; Cédras et al., 1995; Aggarwal et al., 1997; Aggarwal et al., 1998; Aggarwal et al., 1999; Gavrila, 1999; Moeslund et al., 2001; Wang et al., 2003), is mainly composed of deterministic and geometric methods, strongly relied on constraints and assumptions concerning the human body appearance, movements, and surrounding environment.

These restrictions started to be relaxed in the second wave of works, mainly based on probabilistic and machine learning techniques, which achieved remarkable results (Moeslund et al., 2001; Moeslund et al., 2006; Poppe, 2007; Metaxas, 2008; Sigal et al., 2010; Moeslund et al., 2011). In fact, some authors considered the human pose estimation to be almost solved in constrained scenarios, for example, in which there are a large number of calibrated camera views (more than 10), people wearing tight-fitting clothes, predictable and simple body movements, and a static environment (Sigal et al., 2010). However, the problem remains unsolved in uncontrolled conditions when using a single RGB camera. In this particular case, the challenges persist due to: variability in lighting, human body appearance, size and shape, background's appearance, and in the number of people in the scenes; the occurrence of complex and arbitrary people's motions, moving

camera and background, interactions among people, and non-trivial occlusions and self-occlusions.

It can be considered that the third generation of methods was initiated more recently, by the use of deep learning to tackle the problem (Taylor et al., 2010b). They are reflections of the recent ground-breaking advances of deep architectures in machine learning (Bengio, 2009; LeCun et al., 2015; Goodfellow et al., 2016). Specifically, the success of convolutional neural networks (CNNs) in image classification (Krizhevsky et al., 2012) has encouraged their use in many other computer vision problems. "Deep" human pose estimation methods have advanced the state-of-the-art in the field. Such CNN based approaches were applied to different variations of the problem, such as 3D pose estimation and 2D pose estimation in videos, which shows its versatility. These approaches, such as the ones by Elhayek et al. (2015), Wei et al. (2016), Cao et al. (2017), Kanazawa et al. (2017), Güler et al. (2018), and Trumble et al. (2018), to mention a few, have contributed significantly to the improvement of human pose estimation in less controlled scenarios. However, the further limitation and capabilities of deep learning methods are still to be explored.

While most advances regarding pose estimation and the *understanding* of people in images have relied on discriminative deep learning methods (Chen et al., 2014; Pishchulin et al., 2015; Pfister et al., 2015; Gkioxari et al., 2016; Rafi et al., 2016; Bulat et al., 2016; Insafutdinov et al., 2016; Belagiannis et al., 2017; Chu et al., 2017), less attention was initially given to *generation* of images through deep generative models, possibly because of all the difficulties involved in such task. Due to intractable probability distributions and the high variability of all factors influencing the generation of images, the definition of such models is highly complex and hard to extend to unconstrained images.

However, more recently, Variational Autoencoders (VAEs) (Kingma et al., 2014b; Rezende et al., 2014) and Generative Adversarial Nets (GANs) (Goodfellow et al., 2014) have rapidly become popular deep learning methods to build generative models. These approaches have introduced strategies to tackle inference and stochastic learning in the context of intractable probabilistic computations and

large datasets. GANs constitute a framework which corresponds to a minimax two-player game, where two neural networks are used to play the role of generative and discriminative models, respectively, without the need for approximate inference. On the other hand, VAEs are well-principled variational Bayesian lower bound estimators, capable of performing approximate inference and learning of generative models. Both methods present positive and negative attributes (Goodfellow et al., 2014) and several works can be found in the literature extending these approaches or even trying to benefit from the association of them (Mescheder et al., 2017; Wan et al., 2017). Two well-known limitations of deep generative models are their difficulty in generating high-resolution images, as well as disentangled representations, once they usually establish few assumptions on the structure of the probabilistic models. Nevertheless, the characteristics of VAEs and GANs have allowed their application to many problems, such as image-to-text synthesis (Massiceti et al., 2018), hand pose estimation (Wan et al., 2017), and clothing of people (Lassner et al., 2017).

In summary, this thesis pursues further advances in the understanding and generation of people in visual data by the development of new discriminative and generative deep learning approaches.

1.4 Contributions

The main contributions of the present thesis are:

i) a deep learning framework for 2D human pose estimation, which allows for mean-field inference over part-based models;

ii) a conditional deep generative model that achieves state-of-the-art results on generating images of humans conditioned on body posture; and

iii) a structured semi-supervised deep generative model that jointly performs pose estimation and image generation, *understanding* and *generating* people in images in a single framework.

1.5 Publications

The work in this thesis has been published or submitted for publication in the following papers:

1. *Rodrigo de Bem*, Anurag Arnab, Stuart Golodetz, Michael Sapienza, Philip Torr. **Deep Fully-Connected Part-Based Models for Human Pose Estimation.** ***Best paper runner-up*** in Asian Conference on Machine Learning (ACML), 2018.

2. *Rodrigo de Bem*, Arnab Ghosh, Adnane Boukhayma, Thalaiyasingam Ajanthan, Siddharth N., Philip Torr. **A Conditional Deep Generative Model of People in Natural Images.** In Winter Conference on Applications of Computer Vision (WACV), 2019.

3. *Rodrigo de Bem*, Arnab Ghosh, Thalaiyasingam Ajanthan, Ondrej Miksik, Siddharth N., Philip Torr. **A Semi-supervised Deep Generative Model for Human Body Analysis.** In European Conference on Computer Vision (ECCV), 9th International Workshop on Human Behavior Understanding (HBU), 2018.

4. *Rodrigo de Bem*, Arnab Ghosh, Ondrej Miksik, Adnane Boukhayma, Thalaiyasingam Ajanthan, N. Siddharth, Philip Torr. **DGPose: Deep Generative Models for Human Body Analysis.** Submitted to the International Journal of Computer Vision (IJCV), 2018.

Other close related publications are listed below:

1. Zohar Ringel, *Rodrigo de Bem.* **Critical Percolation as a Framework to Analyze the Training of Deep Networks.** In International Conference on Learning Representations (ICLR), 2018.

2. Adnane Boukhayma, *Rodrigo de Bem*, Philip Torr. **3D Hand Shape and Pose from Images in the Wild.** In Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

1.6 Thesis Outline

We organize this thesis as follows:

Chapter 2. We present a brief introduction to the fundamentals of deep convolutional neural networks and deep generative models. We refer to the relevant literature in the field and shed light over the points which are crucial to the development of our contributions.

Chapter 3. We present a 2D pose estimation method in which the human body is modelled as a fully-connected conditional random field (CRF). Over it, approximate inference is efficiently performed using the mean-field algorithm, implemented as a deep recurrent neural network (RNN). Our method achieves competitive results on two well-established benchmarks for 2D human pose estimation, the MPII (Andriluka et al., 2014), and the LSP (Johnson et al., 2010), outperforming state-of-the-art methods in particular cases.

Chapter 4. We propose a deep generative model of humans in natural images which keeps 2D pose separated from other latent factors of variation, such as background scene and clothing. Our single-stage end-to-end conditional-VAEGAN learns directly on the image space allowing the sampling of people with independent variations of pose and appearance. We validate our method on the Human3.6M dataset (Ionescu et al., 2014) and achieve state-of-the-art results on the ChictopiaPlus benchmark (Lassner et al., 2017), outperforming the closest related work in the literature, the ClothNet-Body network (Lassner et al., 2017).

Chapter 5. In this work, our structured semi-supervised approach allows for pose estimation to be performed by the model itself and relaxes the need for labelled data. Therefore, our method aims for the joint *understanding* and *generation* of people in natural images, since it is not only capable of mapping images to interpretable latent representations, but it is also capable of mapping these representations back to the image-space. We compare our models with relevant baselines, the ClothNet-Body (Lassner et al., 2017), and the Pose Guided Person Generation (Ma et al., 2017b) networks, demonstrating their merits on the Human3.6M (Ionescu et al., 2014), ChictopiaPlus (Lassner et al., 2017), and DeepFashion (Liu et al., 2016b) benchmarks.

Appendices. Here we present additional papers containing developed work that is supplementary to the main scope of the thesis.

2

Deep Discriminative and Generative Models Fundamentals

The main difference between discriminative and generative models in machine learning is that the former attempt to directly compute the posterior probability of labels given data. In contrast, the latter aim for the corresponding joint probability distribution, being able to indirectly calculate the posterior distribution mentioned earlier using the Bayes' rule.

Regarding deep learning methods, such definitions are still valid (Deng et al., 2016). However, different neural networks architecture may be employed to implement one model or another. Here, we briefly introduce the fundamentals of the two main building blocks of the discriminative and generative approaches presented in the current thesis. Respectively, deep convolutional neural networks (CNNs) and deep variational autoencoders (VAEs).

2.1 Deep Convolutional Neural Networks

One crucial task in machine learning is to extract features from the raw input data and to create feature vectors suitable for learning. Deep learning or deep architectures consist of a set of methods that allow a machine to be fed with raw data and to automatically discover the feature vectors needed for tasks such as

classification. These methods present multiple levels of internal representation, which are obtained by composing simple non-linear modules that transform the representation at one level (starting with the raw input) into a representation in a higher and slightly more abstract level (LeCun et al., 2015; Bengio, 2009; Goodfellow et al., 2016).

In a modular or layer-wise approach (LeCun et al., 2012; Goodfellow et al., 2016), a deep neural network architecture can be initially considered as a monolithic gradient-based learning machine. It computes the function $f(\mathbf{w}, \mathbf{z}^{(p)})$, where $\mathbf{z}^{(p)}$ is the p-th input vector (raw data), and $\mathbf{w}$ defines the vector of adjustable parameters (weights) of the network. In a supervised learning context, the weights are learned from pairs of training data in the set $\{(\mathbf{z}^{(p)}, \mathbf{d}^{(p)})\}_{p=1}^{P}$, where each sample $\mathbf{z}^{(p)}$ is associated with a label $\mathbf{d}^{(p)}$, which is the desired output of the network for that sample. A loss function $\mathcal{E}^{(p)} = \mathcal{L}(\mathbf{d}^{(p)}, \hat{\mathbf{d}}^{(p)})$, with $\mathcal{E}^{(p)} \in \mathbb{R}$, measures the difference between the output produced by the network $\hat{\mathbf{d}}^{(p)} = f(\mathbf{w}, \mathbf{z}^{(p)})$ and the correct output $\mathbf{d}^{(p)}$ for the p-th sample. The average error over the entire training set is denoted by $\mathcal{E}_{train} \in \mathbb{R}$. The learning process is then posed as the minimisation of the error $\mathcal{E}_{train}$ with respect to the weights $\mathbf{w}$ as

$$\underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{P} \sum_{p=1}^{P} \mathcal{L}(\mathbf{d}^{(p)}, f(\mathbf{w}, \mathbf{z}^{(p)})). \tag{2.1}$$

The optimisation problem is typically solved iteratively with variations of the gradient descent algorithm (LeCun et al., 1989). The way the inputs are propagated through the network is called *forward pass*, while the way the error gradients are propagated backward in the process of updating the weights is the *backward pass*. The manner of performing the backward pass is normally established by the backpropagation algorithm (LeCun et al., 2012).

Despite the initial holistic view, the function $f(.)$ is usually a composite of functions, corresponding to the network layers or modules, which can be arranged as a directed acyclic graph (DAG). In the simplest case, the network is a sequential stack of layers, where the forward pass of each layer is defined by the function

$$\mathbf{x}_n = f_n(\mathbf{w}_n, \mathbf{x}_{n-1}), \tag{2.2}$$

with $n = 1, ..., N$, where N is the total number of layers of the network; $\mathbf{x}_n$ is the output vector of the layer n; $\mathbf{w}_n$ is its vector of adjustable weights, which is a subset of $\mathbf{w}$; and $\mathbf{x}_{n-1}$ is its input vector, which is also the output vector of layer $n-1$. For each sample p in the training data, the first layer has the input $\mathbf{x}_0 = \mathbf{z}^{(p)}$.

Concerning the updating of the weights of the network, if the partial derivative of the error $\mathcal{E}^{(p)}$ with respect to a layer's output $\mathbf{x}_n$ is known, then the partial derivatives of $\mathcal{E}^{(p)}$ with respect to the layers' weights $\mathbf{w}_n$ and input $\mathbf{x}_{n-1}$, given the Equation 2.2, can be defined by the following backward equations

$$\frac{\partial \mathcal{E}^{(p)}}{\partial \mathbf{w}_n} = \frac{\partial f_n(\mathbf{w}_n, \mathbf{x}_{n-1})}{\partial \mathbf{w}_n} \frac{\partial \mathcal{E}^{(p)}}{\partial \mathbf{x}_n}, \tag{2.3}$$

$$\frac{\partial \mathcal{E}^{(p)}}{\partial \mathbf{x}_{n-1}} = \frac{\partial f_n(\mathbf{w}_n, \mathbf{x}_{n-1})}{\partial \mathbf{x}_{n-1}} \frac{\partial \mathcal{E}^{(p)}}{\partial \mathbf{x}_n}, \tag{2.4}$$

where $\frac{\partial f_n(\mathbf{w}_n, \mathbf{x}_{n-1})}{\partial \mathbf{w}_n}$ is the Jacobian of $f_n(.)$ with respect to $\mathbf{w}_n$ evaluated at the point $(\mathbf{w}_n, \mathbf{x}_{n-1})$, and $\frac{\partial f_n(\mathbf{w}_n, \mathbf{x}_{n-1})}{\partial \mathbf{x}_{n-1}}$ is the Jacobian of $f_n(.)$ with respect to $\mathbf{x}_{n-1}$, at the same point. The base of the backward pass of the backpropagation is the fact that in the last layer where $n = N$, for each sample p in the training data the output is $\mathbf{x}_n = \mathcal{E}^{(p)}$, and consequently $\frac{\partial \mathcal{E}^{(p)}}{\partial \mathbf{x}_n} = \frac{\partial \mathcal{E}^{(p)}}{\partial \mathcal{E}^{(p)}} = 1$. From this starting point, Equations 2.3 and 2.4 are applied to all layers, from layer N to layer 1.

Deep convolutional neural networks are architectures which can be considered as a class of the well-known multi-layer non-linear feedforward networks[1] (Haykin, 1998; Bishop et al., 2006). CNNs are biologically inspired by a model of the visual system proposed by Hubel et al. (1962), which based their findings on the study of a cat's visual cortex. The Neocognitron, proposed by Fukushima (1980), is the pioneer CNN computational model, however, the methodology was practically employed and popularised only 10 years later with the LeNet-5 by LeCun et al. (1989). Their CNN architecture achieved state-of-the-art results in handwritten digits recognition employing a gradient-based learning algorithm (Lecun et al., 1998). The main characteristics of the deep CNNs are local connections, shared weights, translational

[1] Although the term multi-layer perceptron has been vastly employed in the literature, some authors consider it is misused because the perceptron learning procedure is not in fact used by the multi-layer feedforward networks.

invariance, and many layers (LeCun et al., 2015). These attributes are obtained by means of two fundamental points: their architecture and the operations performed by their *convolutional*, *non-linear* and *pooling* layers.

Concerning the analysis of humans, we can highlight relatively few approaches in the literature which tackle the problem with ordinary multi-layer neural networks (Rosales et al., 2000a; Rosales et al., 2000c; Rosales et al., 2000b; Takahashi et al., 2000; Rosales et al., 2001). Although their use was relatively popular and successful in applications such as face detection and recognition, in general, the neural networks were considered as not capable of dealing well with the complexity of the full body pose (Soatto et al., 2008). However, in 2010, Taylor et al. (2010b) presented a first approach tackling the problem with deep CNNs.

2.2 Deep Variational Autoencoders

Variational Autoencoders (VAEs) (Kingma et al., 2014b; Rezende et al., 2014) and Generative Adversarial Nets (GANs) (Goodfellow et al., 2014) are deep learning methods to build generative models. These approaches have introduced strategies to tackle inference and stochastic learning in the context of intractable probabilistic computations and large datasets. GANs constitute a framework which corresponds to a minimax two-player game, where two neural networks are used to play the role of generative and discriminative models, respectively, without need for approximate inference. On the other hand, VAEs are well-principled variational Bayesian lower bound estimators, capable of performing approximate inference and learning of generative models. Both methods present positive and negative attributes, and several works can be found in the literature extending these approaches or even trying to benefit from the association of them (Mescheder et al., 2017; Wan et al., 2017). Two well-known limitations of deep generative models are their difficulty of generating high-resolution images, as well as disentangled representations, once they usually establish few assumptions on the structure of the probabilistic models. Nevertheless, the characteristics of VAEs and GANs have allowed their application to many problems, such as text-to-image synthesis (Mansimov et al., 2015), hand

pose estimation (Wan et al., 2017), clothing of people (Lassner et al., 2017), and image-to-image translation (Isola et al., 2017). Our deep generative models (Chapters 4 and 5) rely mainly on the VAE framework, using a GAN discriminator loss to improve the quality of image generation. Thus, following we describe the fundamentals of the VAE methodology, referring the reader to Goodfellow et al. (2014) for further details about GANs.

Maximum likelihood (ML) estimation in models with latent variables (missing data) is a well-known problem in machine learning. It can be defined as

$$\underset{\theta}{\operatorname{argmax}}\, L(\theta; \mathbf{x}), \tag{2.5}$$

$$L(\theta; \mathbf{x}) = p_\theta(\mathbf{x}) = \sum_{\mathbf{z}} p_\theta(\mathbf{x}, \mathbf{z}), \tag{2.6}$$

where $L(\theta; \mathbf{x})$ denotes the likelihood function, $\mathbf{x}$ is an observable random variable, $\mathbf{z}$ is a latent random variable and θ corresponds to the parameters of the joint probability distribution $p_\theta(\mathbf{x}, \mathbf{z})$, which is marginalised over the latent variable $\mathbf{z}$. The notation $p_\theta(.)$ denotes the conditional probability $p(.|\theta)$. The analogous maximisation for the log-likelihood may be defined as follows,

$$\underset{\theta}{\operatorname{argmax}}\ \log\, L(\theta; \mathbf{x}), \tag{2.7}$$

$$\log\, L(\theta; \mathbf{x}) = \log\, p_\theta(\mathbf{x}) = \log \left\{ \sum_{\mathbf{z}} p_\theta(\mathbf{x}, \mathbf{z}) \right\}. \tag{2.8}$$

An equally well-known problem is the fact that, even when $p_\theta(\mathbf{x}, \mathbf{z})$ belong to the exponential family of functions, the summation (in the discrete case) over the latent variable $\mathbf{z}$ remains inside the logarithmic function in the Equation 2.8. This results in a complicated and commonly non-closed form for the maximum likelihood solution.

Considering this issue, the Expectation-Maximisation (EM) algorithm (Dempster et al., 1977; Neal et al., 1998) may be employed as an iterative alternative for the ML estimation in the presence of latent variables. In a variational context, the log-likelihood can be decomposed as (Bishop et al., 2006),

$$\log\, p_\theta(\mathbf{x}) = \mathcal{L}(\phi, \theta; \mathbf{x}) + \mathrm{D}_{\mathrm{KL}}[q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})], \tag{2.9}$$

with

$$\mathcal{L}(\phi, \theta; \mathbf{x}) = \sum_{\mathbf{z}} q_\phi(\mathbf{z}|\mathbf{x}) \log\left\{\frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})}\right\}, \tag{2.10}$$

$$\mathrm{D}_{\mathrm{KL}}[q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})] = -\sum_{\mathbf{z}} q_\phi(\mathbf{z}|\mathbf{x}) \log\left\{\frac{p_\theta(\mathbf{z}|\mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{x})}\right\}, \tag{2.11}$$

where $\mathcal{L}(\phi, \theta; \mathbf{x})$ is a lower bound on $\log p_\theta(\mathbf{x})$ and $\mathrm{D}_{\mathrm{KL}}[q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})]$ is the Kullback-Leibler (KL) divergence between an introduced approximate distribution $q_\phi(\mathbf{z}|\mathbf{x})$ and the true posterior distribution $p_\theta(\mathbf{z}|\mathbf{x})$.

The EM algorithm, in fact, finds the maximum likelihood solution for $\log p_\theta(\mathbf{x})$ maximising its lower bound $\mathcal{L}(\phi, \theta; \mathbf{x})$. In the Expectation step, the lower bound is maximised w.r.t. $q_\phi(\mathbf{z}|\mathbf{x})$, while θ is kept fixed. The maximum is achieved when $q_\phi(\mathbf{z}|\mathbf{x}) = p_\theta(\mathbf{z}|\mathbf{x})$, which makes the KL-divergence equal to zero. In the Maximisation step, the lower bound is maximised w.r.t. θ, while $q_\phi(\mathbf{z}|\mathbf{x})$ is held fixed. This will cause the lower bound, and consequently the log-likelihood, to increase, unless it is already at a maximum. These two steps are repeated until convergence to the final θ parameters. Although effective, it is important to notice that the EM algorithm does not scale efficiently to large datasets, due to the Expectation step, that implies the re-evaluation of all data points in a training set at test time. This issue can be overcome with the use of variational autoencoders, in which a neural network is employed to compute $q_\phi(\mathbf{z}|\mathbf{x})$, as detailed following.

VAEs, as introduced by Kingma et al. (2014b), consist of two components, namely an *encoder* and a *decoder*, which normally correspond to neural networks such as MLPs and CNNs. For each data sample $\mathbf{x} \in \mathcal{D}$ in the training set, the encoder $q_\phi(\mathbf{z}|\mathbf{x})$ approximates the distribution of a latent random variable $\mathbf{z}$ given the data sample. The encoder parameters ϕ are the parameters of the neural network corresponding to it, which is commonly referred to as the "recognition network" or the "inference network". In its turn, the decoder is cast as the conditional probability distribution $p_\theta(\mathbf{x}|\mathbf{z})$ that is analogously correspondent to a neural network parameterised by θ, usually referred to as the "generative network", which aims to reconstruct the input sample given the latent variables. By defining a

prior distribution over $\mathbf{z}$, the decoder consequently defines a joint distribution $p_\theta(\mathbf{x}, \mathbf{z}) \propto p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})$. Inference and generative networks are jointly trained using stochastic optimisation for the maximisation, w.r.t. ϕ and θ, of the variational lower bound of the marginal likelihood over the training data, which can be defined as

$$\begin{aligned}
\log\, p_\theta(\mathcal{D}) &\geq \mathcal{L}(\phi, \theta; \mathcal{D}) \\
&= \sum_{\mathbf{x}^i \in \mathcal{D}} \mathcal{L}(\phi, \theta; \mathbf{x}^i) \\
&= \sum_{\mathbf{x}^i \in \mathcal{D}} \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^i)} \log \left\{ \frac{p_\theta(\mathbf{x}^i, \mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x}^i)} \right\} \text{(from Equation 2.10)} \\
&= \sum_{\mathbf{x}^i \in \mathcal{D}} \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^i)} [\log\, p_\theta(\mathbf{x}^i, \mathbf{z}) - \log\, q_\phi(\mathbf{z}|\mathbf{x}^i)] \\
&= \sum_{\mathbf{x}^i \in \mathcal{D}} \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^i)} [\log\, p_\theta(\mathbf{x}^i|\mathbf{z}) + \log\, p(\mathbf{z}) - \log\, q_\phi(\mathbf{z}|\mathbf{x}^i)] \\
&= \sum_{\mathbf{x}^i \in \mathcal{D}} \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^i)} [\log\, p_\theta(\mathbf{x}^i|\mathbf{z})] - \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^i)} [\log\, q_\phi(\mathbf{z}|\mathbf{x}^i) - p(\mathbf{z})] \\
&= \sum_{\mathbf{x}^i \in \mathcal{D}} \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^i)} [\log\, p_\theta(\mathbf{x}^i|\mathbf{z})] - \mathrm{D}_{\mathrm{KL}}[q_\phi(\mathbf{z}|\mathbf{x}^i)||p(\mathbf{z})]. \qquad (2.12)
\end{aligned}$$

The first term of Equation 2.12 corresponds to the decoder and measures the reconstruction error, while the second term is the KL-divergence between the approximate distribution generated by the encoder and the prior distribution over the latent variables, acting as a regulariser of ϕ, as it encourages the approximate posterior to be close to the prior. In order to make the cost function in Equation 2.12 fully differentiable, a re-parameterisation trick is employed by Kingma et al. (2014b), consisting of sampling from the posterior $\mathbf{z}^i \sim q_\phi(\mathbf{z}|\mathbf{x}^i)$, using $\mathbf{z}^i = g_\phi(\mathbf{x}^i, \epsilon) = \mu_i + \sigma_i \odot \epsilon$, where $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\odot$ denotes an element-wise product.

Siddharth et al. (2017) introduced an extension of (Kingma et al., 2014c) by defining an interpretable and semi-supervised subset $\mathbf{y}$ of the latent variables $\mathbf{z}$. This definition changes the recognition model (approximate posterior) and the generative model, respectively, to $q_\phi(\mathbf{z}, \mathbf{y}|\mathbf{x})$ and $p_\theta(\mathbf{x}, \mathbf{y}, \mathbf{z})$, allowing them to have arbitrary conditional dependency structures. To implement these modifications, a new objective function and a stochastic computational graph are defined (Schulman et al., 2015). Despite the recent methods in the literature tackling pose-transfer

and generation of humans in images, to our knowledge, no work has explored interpretable, structured and semi-supervised learning as we propose in Chapters 4 and 5.

3

A Deep Fully-Connected Part-Based Model for Human Pose Estimation

In this chapter, we propose a 2D multi-level appearance representation of the human body in RGB images, spatially modelled using a fully-connected graphical model. The appearance model is based on a CNN body part detector, which uses shared features in a cascade architecture to simultaneously detect body parts with different levels of granularity. We use a fully-connected conditional random field (CRF) as our spatial model, over which approximate inference is efficiently performed using the mean-field algorithm, implemented as a recurrent neural network (RNN). The stronger visual support from body parts with different levels of granularity, along with the fully-connected pairwise spatial relations, which have their weights learnt by the model, improve the performance of the bottom-up part detector. We adopt an end-to-end training strategy to leverage the potential of both our appearance and spatial models and achieve competitive results on the MPII (Andriluka et al., 2014) and LSP (Johnson et al., 2010) datasets.

3.1 Introduction

Human pose estimation is a fundamental problem in computer vision, with important applications in human-computer interaction, motion capture for films and games,

activity recognition and prediction, and augmented reality (Moeslund et al., 2011). Unconstrained monocular pose estimation presents several challenges. The human body, modelled as an articulated object, lies in a high-dimensional space in which finding possible configurations is costly. The projection from the 3D world to the 2D image plane is a source of uncertainties and ambiguities that must also be overcome. Lastly, one can mention the wide variety of sizes, shapes, appearances and poses that the human body can assume, and the endless number of different environments in which it can appear. For these reasons, the most general scenario of *markerless* pose estimation in monocular RGB images is often simplified by the introduction of constraints. The use of multiple cameras, indoor studios, artificial markers over body parts and depth sensors can all help to simplify the problem. Despite these strategies, in many cases images from single RGB cameras are the only ones available; in other cases, the imposition of constraints and intrusions can be difficult or undesirable, such as in outdoor environments, or controlled areas such as medical facilities. Besides that, for applications such as autonomous driving, the use of single RGB sensors dramatically simplifies and reduces the cost of the data acquisition process, obviating the need for steps such as simultaneous calibration, image registration or synchronisation. Markerless pose estimation in monocular RGB images thus remains a crucial goal of current research.

Recently, substantial advances have been made by using deep learning to tackle the problem, greatly contributing to the improvement of human pose estimation in less-controlled scenarios (Elhayek et al., 2015). To our knowledge, the first effort in this direction was made by Taylor et al. (2010b), who employed deep convolutional architectures for learning embeddings of images with people in similar poses, but with different clothes, background and other appearance changes. "Deep" human pose estimation methods have advanced the state-of-the-art (see Sec. 3.2), but occlusions, cluttered scenes and unusual poses, which may lead to false positives and negatives when locating body parts, still represent challenges.

In this chapter, we propose a deep CNN architecture, illustrated in Fig. 3.1, to tackle 2D markerless human pose estimation in monocular RGB images. Our

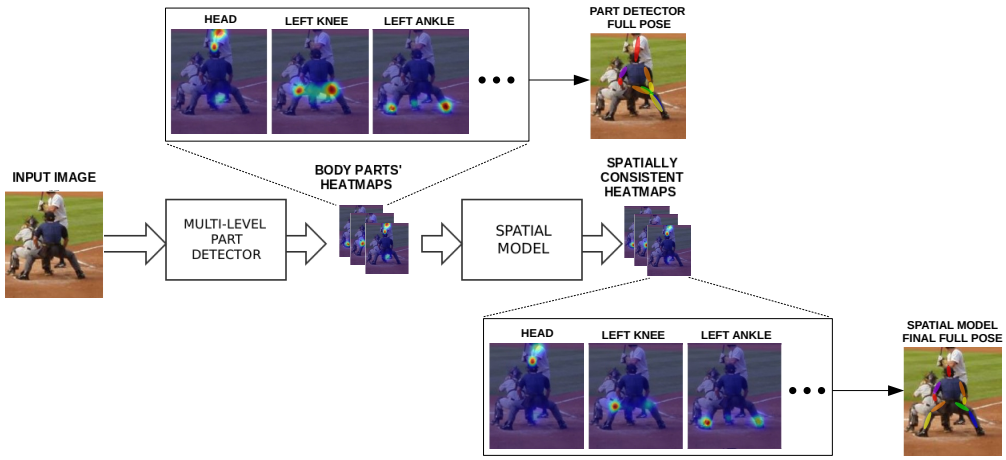


Figure 3.1: Our architecture consists of a part detector, which produces a heatmap for each body part, and a spatial model, which encourages spatial consistency over the detected body parts. A sample input image, along with sample heatmaps and full poses, illustrates how it can help with problems such as double-counting.

architecture is composed of a multi-level body part detector and a fully-connected CRF model. Both modules are jointly trained end-to-end. We show that our detector, producing a holistic representation of the human body, benefits from the use of auxiliary parts (e.g., rigid parts) to locate joints. It is also noticeable that the CRF model, using the spatial relationships between body parts, learns which are the ones that exert more influence on each other, thus encouraging the generation of consistent poses. We evaluate our approach on both the MPII (Andriluka et al., 2014) and LSP (Johnson et al., 2010) datasets, achieving competitive results and outperforming state-of-the-art methods in particular cases. In summary, our main contributions are:

i) A multi-level Gaussian representation for the human body and a novel, inexpensive, and simple weakly-supervised methodology for generating corresponding ground-truth annotations.

ii) A novel framework for performing deep learning and inference in part-based models. To our knowledge, the presented method is the first to allow mean-field approximate inference over a loopy, fully-connected, part-based model using a deep end-to-end architecture. In our CRF, each body part corresponds to a distinct binary random field, and all parts together compose the part-based model (e.g., the human body). Pairwise functions are defined only between binary random variables

in different random fields corresponding to *neighbouring* body parts: we dub these *inter-field* relations. Such relations are established according to the structure of the part-based model (e.g., fully-connected), differently from *intra-field* relations (i.e. between variables in the same field), as used in CRF models for image segmentation (Zheng et al., 2015; Krähenbühl et al., 2011).

iii) Competitive results on two well-established benchmarks for 2D human pose estimation, the MPII (Andriluka et al., 2014) and the LSP (Johnson et al., 2010) datasets, outperforming state-of-the-art methods in particular cases, e.g., unusual poses.

iv) Finally, although we present here the more general fully-connected setup, with joints' locations as features, our method allows for different part-based models' structures (e.g., tree-structured, star-structured, etc.), as well as for the use of multiple and different image features (e.g., colour, texture, and depth). Thus, our framework may be applied to different deep part-based models for pose estimation, and also to other problems in which part-based models are employed, such as object co-detection (Hayder et al., 2014) and multi-person detection (Liu et al., 2016a).

3.2 Related Work

Recent years have seen outstanding improvements in 2D human pose estimation, driven by the introduction of CNN-based methods. Early works in this period have established two main lines of approach: the direct regression of joints' sparse coordinates (Toshev et al., 2014), and the dense regression of joints' locations over the image plane (Jain et al., 2015). The latter class of approaches has received greater attention from researchers due to some of its advantages, such as invariance to translation, multi-modality and a natural capability to handle smoother cost functions.

Many methods in this class have adopted exclusively bottom-up strategies, where joints are detected without any explicit use of prior knowledge or spatial reasoning about the human body (Gkioxari et al., 2016; Rafi et al., 2016; Pfister et al., 2015; Belagiannis et al., 2017; Wei et al., 2016; Bulat et al., 2016). Other

works employ different forms of ad hoc, top-down refinements (Jain et al., 2015; Newell et al., 2016; Carreira et al., 2016; Hu et al., 2015; Lifshitz et al., 2016; Ke et al., 2018). Lastly, another group of approaches (Tompson et al., 2014; Chen et al., 2014; Tompson et al., 2015; Pishchulin et al., 2015; Insafutdinov et al., 2016; Dai et al., 2016; Chu et al., 2017) perform structured predictions relying on part-based models (Fischler et al., 1973). Such models (Felzenszwalb et al., 2005; Yang et al., 2011), which are in fact instances of probabilistic graphical models such as Markov random fields (MRFs) and conditional random fields (CRFs) (Koller et al., 2009; Lafferty et al., 2001), used to achieve state-of-the-art results on 2D pose estimation until the advent of CNN techniques. Our architecture is designed to jointly benefit from a bottom-up, CNN-based part detector and a part-based, fully-connected CRF model. We build upon knowledge and concepts from several lines of works in the literature, as we will now describe in more detail.

Multi-level appearance representations, which simultaneously gather coarse and fine visual cues, are well-known in the pose estimation literature (Pishchulin et al., 2013; Mikolajczyk et al., 2004). The underlying principle is that stronger and more diverse visual features from images can facilitate the location of joints. Concerning CNN-based approaches, local and global appearance models are employed by Fan et al. (2015), whilst Belagiannis et al. (2017) use auxiliary body parts to locate joints. In our work, we propose a simple weak annotation method to automatically derive the ground truth for the rigid parts and whole body from the manually annotated joints' coordinates.

Fully convolutional CNN architectures employing an efficient sliding-window technique (Fernando et al., 2015; Long et al., 2015) appear in the context of pose estimation in work by Tompson et al. (2014). We use shared features as a **multitask learning** strategy to benefit from auxiliary body parts when locating the joints. Caruana (1998) mentioned that secondary tasks (e.g. detection of rigid parts and whole body) might produce an *inductive bias* towards the correct learning of the main task, and also act as regularisers in many cases. Li et al. (2015) apply a multitask approach in their CNN architecture for simultaneous regression

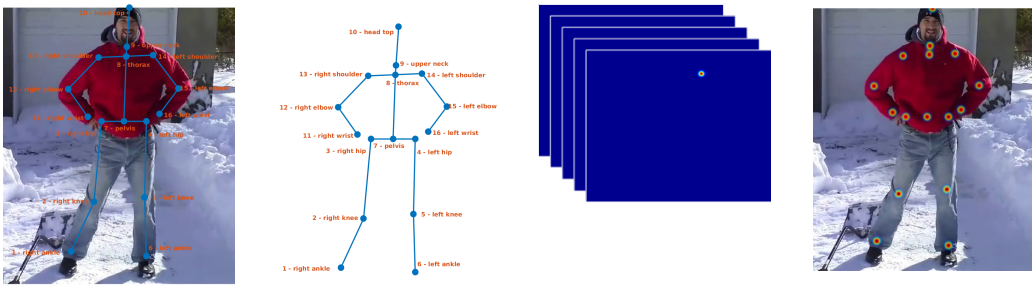
and detection of body parts. By contrast, we handle multiple tasks in a cascade architecture, inspired by the instance segmentation approach of Dai et al. (2016).

Cascaded CNN architectures, such as the one by Wei et al. (2016), have their foundations in cascaded or stacked classifiers (Heitz et al., 2009; Tu, 2008), in which classification outputs are sequentially refined through chains of classifiers.

Probabilistic graphical models for human pose estimation are explored jointly with CNNs by Tompson et al. (2014) and Tompson et al. (2015), where the authors introduced a method based on an MRF. Chen et al. (2014), Pishchulin et al. (2015), and Insafutdinov et al. (2016) explore unaries and pairwise terms in graphical model frameworks associated with deep architectures but not implemented in an end-to-end manner. Chu et al. (2017) achieve very good results by employing a CRF model on top of a pose estimation architecture (Newell et al., 2016). Our graphical model approach is closely related to the one by Kiefel et al. (2014), in which the authors proposed a mean-field inference algorithm over a CRF model for human pose estimation using HOG features (Dalal et al., 2005). Such algorithms were inspired by Krähenbühl et al. (2011), which showed how to perform efficient approximate inference over fully-connected CRF models for semantic segmentation. More recently, Zheng et al. (2015) showed how the mean-field algorithm could be formulated as a recurrent network for pixel-wise classification. However, it is important to note that the concept of a part does not exist in either (Zheng et al., 2015) or (Krähenbühl et al., 2011), preventing their direct application to part-based models. In this chapter, differently from previous approaches, we present a deep architecture to perform mean-field inference in part-based models applied to markerless human pose estimation.

3.3 Our Approach

As shown in Fig. 3.1, we propose a deep neural network architecture that consists of two modules: a multi-level body part detector and a spatial model. The detector finds human body parts at multiple levels of granularity (joints, rigid parts, and whole body), producing a dense heatmap for each part, whilst the spatial model



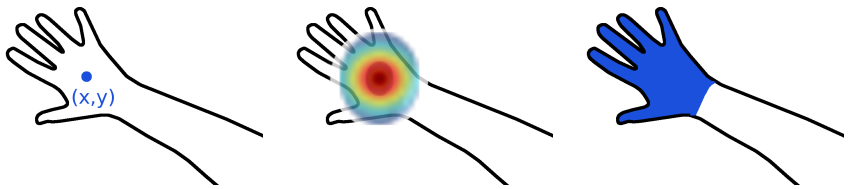
(a) **(b)** **(c)** **(d)**

Figure 3.2: (a) Annotated joints, highlighted and named, superimposed on a cropped image from the MPII Human Pose dataset (Andriluka et al., 2014). The links between the joints are not part of the annotations and were just included to make the visualisation easier. (b) The "skeleton" of the person along with the annotated positions and the names of the joints. (c) Result of the ground-truth mapping: from the joints' 2D annotations to a set of dense 2D heatmaps. (d) All the Gaussians representing the annotated joints superimposed on the original image.

encourages the prediction of consistent poses. We compose the two modules into a neural network and train it end-to-end using ground-truth heatmaps, constructed from manually-annotated positions in the case of the joints, and automatically generated with a simple, weakly supervised strategy for the rigid parts and the whole body (since they are not labelled on the employed datasets). The way in which we generate these heatmaps and the structure of the two network modules are described in the following subsections.

3.3.1 Multi-Level Gaussian Representation

We represent human pose using a set of oriented 2D Gaussian heatmaps, one for each joint (e.g., right shoulder) and rigid part (e.g., right forearm), and an additional one for the body as a whole. This approach builds upon recent successful approaches in the literature (Wei et al., 2016; Newell et al., 2016) that modelled joints alone as 2D Gaussians (see Fig. 3.2). The intuition behind adding rigid parts and the whole body to our representation is that in many images, they can be easier to see than the joints. Thus, these extra visual cues at multiple levels of granularity, i.e. semi-global and global (Pishchulin et al., 2013; Bourdev et al., 2009), act as inductive biases (Caruana, 1998) that allow us to better estimate the correct locations of the joints.



(a) (b) (c)

Figure 3.3: Range of body parts representations. (a) single 2D discrete point representation; (b) dense Gaussian representation; (c) dense per-pixel representation. Heatmaps with Gaussian representation stand on the central position of the spectrum. It takes into account inherent uncertainties about the location of the body parts and conveys much more spatial information that the point representation. However, with our weak annotation strategy, it is much cheaper to obtain in terms of annotation cost than the per-pixel labelling.

Among several options to represent body parts, we consider that heatmaps with Gaussian representations rest between two extremes, namely discrete 2D points (pose vector) and per-pixel annotations, as illustrated in Fig. 3.3. Such representation conveys more information than discrete 2D coordinates, yet still much cheaper to obtain than per-pixel part labelling. We proposed our simple weakly supervised and part type-specific strategy, based on annotated 2D joint locations for constructing heatmaps.

Formally, our representation consists of $P = J + R + B$ body elements, namely J joints (numbered $1 \dots J$), R rigid parts (numbered $J+1 \dots J+R$) and one element ($B = 1$) for the whole body (numbered $J + R + B$). Each body element p is represented using a 2D Gaussian around its centre $\boldsymbol{\mu}_p = (i_p, j_p)$, with covariance matrix Σ_p, i.e. $G_p(i,j) = \frac{1}{\sqrt{2\pi|\Sigma_p|}} \exp\left(-\frac{1}{2}\begin{bmatrix} i-i_p \\ j-j_p \end{bmatrix}^\top \Sigma_p^{-1} \begin{bmatrix} i-i_p \\ j-j_p \end{bmatrix}\right)$. In this, (i,j) denotes a spatial position within the heatmap, with $i \in \{1, ..., H\}$ and $j \in \{1, ..., W\}$, where H and W are the heatmap height and width, respectively. The covariance matrix for a body element p can be decomposed as $\Sigma_p = R_p \begin{bmatrix} \sigma_{p,i}^2 & 0 \\ 0 & \sigma_{p,j}^2 \end{bmatrix} R_p^\top$, in which $\sigma_{p,i}$ and $\sigma_{p,j}$ respectively denote the standard deviations of the Gaussian along the principal axes of the body element, and R_p denotes the rotation from image space to body element space that defines the Gaussian's orientation. The multi-level body representation is illustrated in Fig. 3.4.

(a) **(b)** **(c)**

Figure 3.4: The 2D Gaussian of each body part is contained in a different heatmap; however, for simplicity, all the Gaussians are superimposed together on the same image here. (a) 2D Gaussians over the joints. (b) 2D Gaussians over the rigid parts. (c) 2D Gaussians over the whole body.

3.3.2 Ground-Truth Heatmap Generation

As mentioned above, to train our network we need ground-truth heatmaps for the body elements, but existing datasets have only been annotated with 2D joint locations (Johnson et al., 2010; Andriluka et al., 2014; Gong et al., 2017). We thus propose a simple weakly supervised and part type-specific strategy of extending these to heatmaps:

Joints. Since joints have a limited spatial extent, we follow previous works (Wei et al., 2016; Newell et al., 2016) in modelling them as isotropic Gaussians (i.e. $\sigma_{p,i} = \sigma_{p,j}$, and $R_p = I$) that are centred at the ground-truth joint location and have a small standard deviation.

Rigid Parts. The centre $\boldsymbol{\mu}_p$ of a rigid part p is defined as the midpoint of the centres $\boldsymbol{\mu}_{p'}$ and $\boldsymbol{\mu}_{p''}$ of the joints it connects. We orient the Gaussian representing the part to align its i axis with the line connecting $\boldsymbol{\mu}_{p'}$ and $\boldsymbol{\mu}_{p''}$. We define $\sigma_{p,i}$ to be proportional to the Euclidean distance $||\boldsymbol{\mu}_{p'} - \boldsymbol{\mu}_{p''}||$, and set $\sigma_{p,j} = \kappa_p \sigma_{p,i}$, where κ_p is a part-specific ratio inspired by anthropometric measures (NASA, 1995).

Body. The body centre is defined to be the mean of the annotated joint centres. Principal component analysis (PCA) of the joint centres is used to obtain the orientation of the body in the image plane. We define $\sigma_{p,i}$ and $\sigma_{p,j}$ to be proportional to the distance between the extreme projections of the joint centres onto the two principal axes.

Finally, we employ two sets of ground-truth heatmaps, one containing all the people in the scenes and the other containing only the person of interest. The second set is only used by the last module of the part detector architecture (Fig. 3.5). A sample ground-truth set is illustrated in Fig. 3.4.

3.3.3 Multi-Level Body Part Detector

This initial module of our network aims to simultaneously detect all body elements under consideration, and thereby refine the detection of the joints by making use of the additional visual cues provided by the rigid parts and whole body. To achieve this, we adopt a cascade architecture where a common set of features are used to predict all body parts (see Fig. 3.5). This is reasonable due to the strong correlations between the visual appearance of the parts.

The proposed fully convolutional architecture produces one dense heatmap for each body part. Only two pooling layers are used over the network, reducing losses of spatial information. For our activation function, we use the Parametric ReLU (He et al., 2015). Another important point is the larger receptive field, which captures more contextual information around the joints.

The detector module as a whole implements a function of the form $f : \mathbb{R}^{H' \times W' \times 3} \times \mathbb{R}^n \mapsto \mathbb{R}^{H \times W \times P}$, which takes an RGB input image $\mathbf{Z}$ and a vector $\mathbf{w}$ of network weights, and yields an output tensor $\mathbf{X}$ that contains a dense heatmap for each of the P body elements under consideration. To learn $\mathbf{w}$, we train the network on a set of images $\{\mathbf{Z}^{(s)}\}$, with $s = 1, ..., S$. For each sample $\mathbf{Z}^{(s)}$ in the training set, we let $\mathbf{X}^{(s)} = f(\mathbf{Z}^{(s)}, \mathbf{w})$, and solve

$$\underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{S} \sum_{s=1}^{S} \sum_{p=1}^{P} \sum_{(i,j)} \mathcal{L}(\mathbf{X}_{i,j,p}^{(s)}; G_p(i,j)), \tag{3.1}$$

where $(i, j) \in H \times W$ denotes a spatial position within a heatmap; $\mathbf{X}_{i,j,p}^{(s)}$ denotes one value of the output tensor at a given coordinate; $G_p(i, j)$ is part p's 2D Gaussian, and $\mathcal{L}$ denotes mean squared error. Eq. 3.1 is minimised using stochastic gradient descent and, for each module in the cascade architecture, we use a similar loss

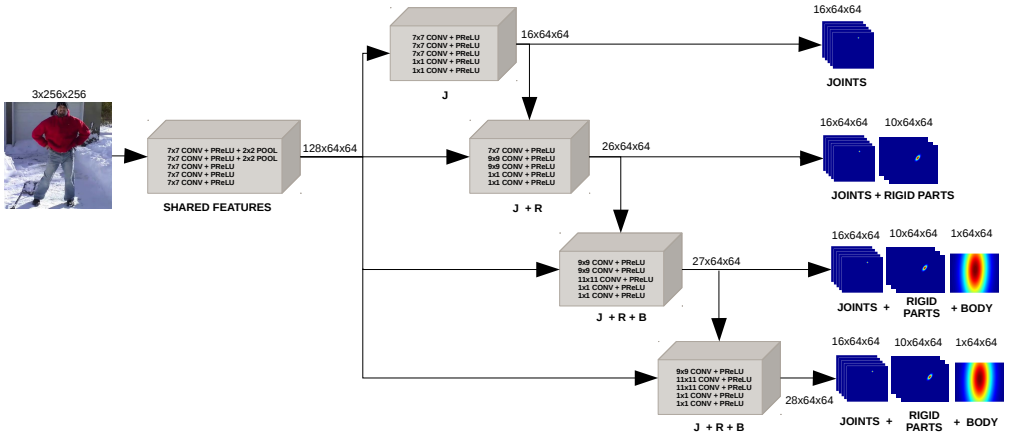


Figure 3.5: Multi-level body part detector basic architecture. A shared convolutional module produces CNN features that are used by subsequent modules to detect joints, rigid parts and the whole body. The initial detections of the joints (heatmaps) obtained by the module **J** – which can be seen as saliency maps (Itti et al., 1998) – are used to focus the *visual attention* (Borji et al., 2013) of the subsequent module **J+R**, which refines the previous predictions at the same time as it produces additional predictions for the rigid parts. The same principle is followed by the third module **J+R+B**, which refines the previous predictions and detects the whole body. The fourth module **J+R+B** is responsible for focusing on the person of interest, usually in the centre of the image. References to the backward pass of the network are not included for simplicity.

function, including only the terms corresponding to the body elements detected by that module.

3.3.4 Fully-Connected Conditional Random Field

Our fully-connected CRF model is defined over the multi-level appearance representation as a framework for performing learning and inference on loopy part-based models for human pose estimation. By contrast with others in the literature, in our approach, not only all body parts within the same level of granularity are fully-connected by means of pairwise relations, but also all parts among all levels of granularity are connected, as illustrated in Fig. 3.6. Redundancy and the strong correlation between rigid parts and joints are positive characteristics in our model. The full connectivity allows each body part to receive messages from all others. This reinforces their consistent location since, ultimately, the posture of the whole body

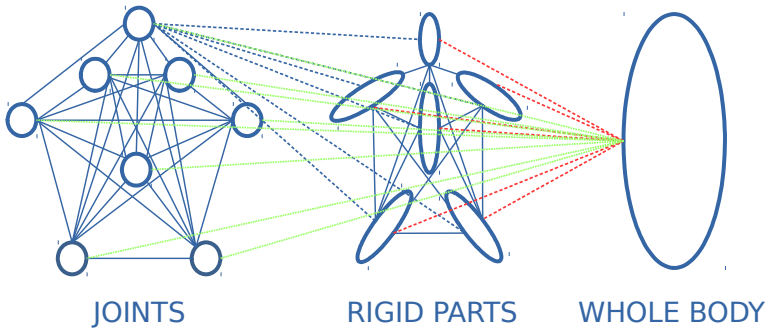


Figure 3.6: Spatial model illustration. The pairwise relations are illustrated as follows: **continuous blue lines** show *joint* ↔ *joint* and *rigid part* ↔ *rigid part* relations; **dashed blue lines** show *joint* ↔ *rigid part* relations; **dashed red lines** show *rigid part* ↔ *body* relations; **dotted green lines** show *joints* ↔ *body* relations. Not all relations are illustrated for clarity. Best viewed in colour.

is taken into account in the message passing. Moreover, instead of imposing a fixed simpler structure *a priori* (e.g., a tree), the strength of the pairwise relations and influence of the parts over each other, and consequently the underlying structure that emerges from that, is data-driven, i.e. learned from training data. Regarding the close relation between rigid parts and joints, as the former usually have a larger area in the images, they provide an inductive bias (Caruana, 1998) towards the correct location of the latter, since the joints are smaller and consequently more susceptible to being ambiguously or wrongly located (e.g., due to clutter or occlusion).

Formally, we introduce random binary variables $Y_{u,p}$ for $p = 1, ..., P$ and $u = 1, ..., H \times W$, each of which can take a value of either 0 or 1, respectively denoting the absence or presence of body part p at raster position u in an image $\mathbf{Z}$. Let $\mathbf{Y}$ be the vector composed of all such binary random variables $Y_{u,p}$. Let us now consider a factor graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V} = \{\mathcal{V}_1, ..., \mathcal{V}_P\}$, which divide the binary random variables in subsets $\mathcal{V}_p = \{Y_{u,p}\}$, each one corresponding to a body part; and edges $\mathcal{E} = (u, v)$, for $u \in \mathcal{V}_p$ and $v \in \mathcal{V}_{p'}$ with $p \neq p'$. In summary, an edge is established between every binary random variable of a given part and all the other random variables of all the other body parts in the representation. Given the observable image $\mathbf{Z}$ and the vector $\mathbf{Y}$, the pair $(\mathbf{Y}, \mathbf{Z})$ can be modelled as a CRF characterised by a Gibbs distribution $P(\mathbf{Y} = \mathbf{y}|\mathbf{Z})$. For the convenience

of notation, we will suppress the conditioning in the following equations. Thus, the energy of a given configuration $\mathbf{y}$ is given by

$$E(\mathbf{y}) = \sum_{p=1}^{P} \left(\sum_{u \in \mathcal{V}_p} \phi(y_{u,p}) + \sum_{p'=1}^{P} \sum_{u \in \mathcal{V}_p} \sum_{v \in \mathcal{V}_{p'}} \psi(y_{u,p}, y_{v,p'}) \right), \tag{3.2}$$

where the unary potentials $\phi(y_{u,p})$ give the likelihood for the presence of a part p in a given location u, and the pairwise potentials $\psi(y_{u,p}, y_{v,p'})$ measure the likelihood for the simultaneous location of parts p and p' in the corresponding positions u and v, respectively. In the proposed model, the unary terms correspond to the energies provided by the CNN-based body part detector (Sec. 3.3.3). The pairwise potentials are defined by weighted Gaussian kernels as

$$\psi(y_{u,p}, y_{v,p'}) = \sum_{k=1}^{K} \mathbf{w}_{p,p'}^{(k)} \boldsymbol{\mu}_{p,p'}^{(k)}(y_{u,p}, y_{v,p'}) \mathbf{k}_{p,p'}^{(k)}(\mathbf{f}_{u,p}, \mathbf{f}_{v,p'}), \tag{3.3}$$

[1]

where we have a linear combination of Gaussian kernels $\mathbf{k}_{p,p'}^{(k)}(.)$, weighted by $\mathbf{w}_{p,p'}^{(k)}$, and the compatibility functions $\boldsymbol{\mu}_{p,p'}^{(k)}(.)$. Each Gaussian kernel is applied on feature vectors $\mathbf{f}_{u,p}$ and $\mathbf{f}_{v,p'}$ from the image, which may encode information such as the location, colour or intensity of the parts. The weights measure the relative importance of each kernel. Finally, the compatibility function is a 2×2 matrix, as it weights how one binary label assignment for one part influences the assignment for the other part. It is important to notice that there are a different Gaussian kernel and compatibility functions between each pair of parts p and p', because of the different ways that different parts influence each other. This differs from the DenseCRF approach for semantic segmentation (Krähenbühl et al., 2011; Zheng et al., 2015), where a single compatibility function and Gaussian kernel are used for the entire model. In our case, we explicitly model the dependencies between different parts in the part-based model, e.g., human body parts. Although our method allows for multiple kernels between each pair of parts p and p', in the current formulation we only use the location of the parts as a feature ($K = 1$),

[1] The main functions and variables are written in bold here to facilitate reading.

thus the feature vectors $\mathbf{f}_{u,p}, \mathbf{f}_{v,p'}$ correspond to the 2D positions of parts p and p', respectively. The Gaussian kernel is then defined as

$$\mathbf{k}^{(k)}_{p,p'}(\mathbf{f}_{u,p}, \mathbf{f}_{v,p'}) = \exp\Big\{ -\frac{1}{2}\Big[(\mathbf{f}_{u,p} - \mathbf{f}_{v,p'}) - \bar{\mathbf{x}}_{p,p'}\Big]^T \Sigma^{-1}_{p,p'}\Big[(\mathbf{f}_{u,p} - \mathbf{f}_{v,p'}) - \bar{\mathbf{x}}_{p,p'}\Big]\Big\}, \quad (3.4)$$

where the vector $\bar{\mathbf{x}}_{p,p'}$ and the matrix $\Sigma_{p,p'}$ are, respectively, the 2D mean displacement and the covariance between the locations of parts p and p'. Thus the Gaussian kernel has maximum value when the distance between two parts is equal to their mean displacement and exponentially decays when their offset moves away from it.

The means $\bar{\mathbf{x}}_{p,p'}$ and the covariance matrices $\Sigma_{p,p'}$ of these Gaussian kernels are learned by maximum likelihood estimation over the training set. In Fig. 3.7 we show some of the learned parameters for the MPII dataset. In each graph, a sample position for one given part is defined to be at the centre of a 256×256 frame. The continuous lines show the normalised expected displacements calculated between each other body part and the sample part in the centre of the frame, whereas the dashed ellipses show the computed normalised standard deviations.

These parameters, and consequently the Gaussian kernels, relate directly to the message-passing step of the mean-field inference, which is interpreted as follows; given a sample body part (e.g., the central ones in Fig. 3.7), at each mean-field iteration it receives messages from all the other parts conveying their expectations about its position. All these expectations are combined with the unary energies through the learnable weights and compatibility matrices. It is important to notice that Fig. 3.7 only shows the part in the central position of the frames "receiving messages", however for each body part at each possible location of the image there is a corresponding binary random variable; thus the messages are in fact exchanged efficiently between all these random variables at each mean-field iteration.

3.3.5 Mean-Field Inference as an RNN in a Part-Based Model

Our final predicted human pose is the Maximum a Posteriori (MAP) estimate of the CRF defined in Eq. 3.2. This corresponds to the assignment $\mathbf{y}^*$ that minimises the

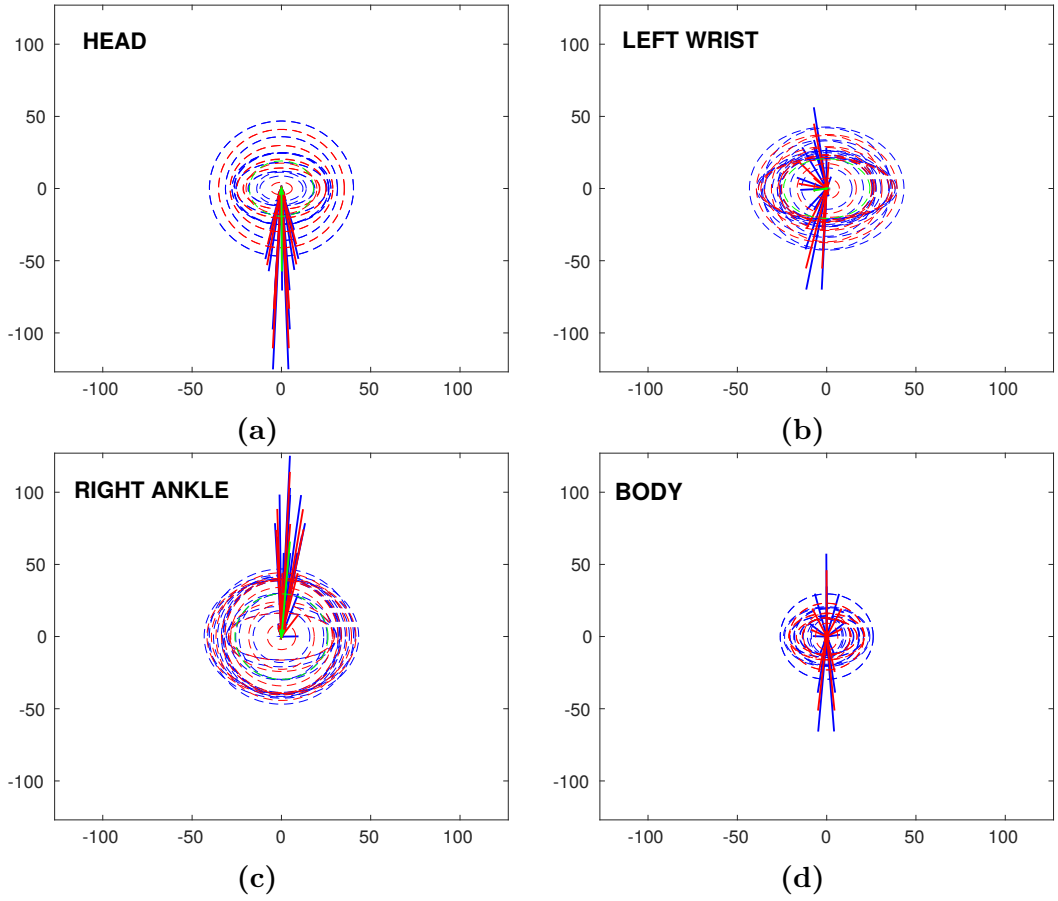


Figure 3.7: Spatial priors. Mean displacements and standard deviations between sample parts (i.e. head, left wrist, right ankle, and body) and all other parts in the model for the MPII dataset. Displacements and standard deviation ellipses between the sample parts and other joints are showed in blue, while the analogous parameters are showed in red for rigid parts and in green for the body. All parameters are normalised w.r.t. the size of the frame. Best viewed in colour.

energy $E(\mathbf{y})$. Exact inference over this loopy fully-connected CRF is intractable. As a result, we perform approximate mean-field inference in which a simpler distribution, expressed as a product of independent marginals, $Q(\mathbf{Y}) = \prod_u Q(Y_{u,p})$, is used to approximate the real distribution, $P(\mathbf{Y})$. The KL-divergence between $P(\mathbf{Y})$ and $Q(\mathbf{Y})$ is then iteratively minimised. The mean-field update equation (Koller et al., 2009; Kiefel et al., 2014) for our fully-connected CRF model is given by

$$Q(y_{u,p}) \propto \exp\Big\{ -\phi(y_{u,p}) - \sum_{p'=1}^{P}\sum_{k=1}^{K} \mathbf{w}_{p,p'}^{(k)} \sum_{l' \in \{0,1\}} \boldsymbol{\mu}_{p,p'}^{(k)}(y_{u,p}, l') \sum_{v \in \mathcal{V}_{p'}} \mathbf{k}_{p,p'}^{(k)}(\mathbf{f}_{u,p}, \mathbf{f}_{v,p'}) Q(l') \Big\}. \tag{3.5}$$

Eq. 3.5 contains a highly costly message-passing operation performed in the pairwise terms calculations. The definition of such pairwise relations as Gaussian kernels allows their computation by means of a convolution operation, which can be efficiently performed using a permutohedral lattice (Adams et al., 2010), reducing the complexity of the message-passing step from quadratic to linear with respect to the number of random variables.

It has recently been shown that mean-field iterative updates are differentiable (Zheng et al., 2015). Consequently, the mean-field inference algorithm can be represented as an RNN, where the estimate of $Q(\mathbf{Y})$ is refined at each time step. This allows us to learn the parameters of our spatial model and the underlying CNN jointly via end-to-end training. As shown in Fig. 3.8, our RNN receives as inputs the unary potentials from the CNN-based part detector and produces spatially consistent heatmaps (marginal distributions) as outputs. Its loss function is the same given by Eq. 3.1.

To define the overall behaviour of the network, let us consider the unary potential $U = -\phi(y_{u,p}) = \text{logit}(V)$. We apply the logit(.) function (inverse Sigmoid) over the body part detector outputs to compute the log-likelihood scores corresponding to the unary energies. Once the unaries are obtained, an initial normalisation step is executed by the softmax(.) function. The Softmax and Logit modules have their error differentials trivially back-propagated. Additionally, let us define an estimation of the marginal probabilities Q from the previous step, and finally, the forward-pass of the mean-field iteration module can be denoted by the function $f_\theta(U, Q)$, where the vector $\theta = \{\mathbf{w}^{(k)}_{p,p'}, \boldsymbol{\mu}^{(k)}_{p,p'}(l, l')\}$ represents the *learnable* CRF parameters. In this context, the following equations define the RNN behaviour, where T is the total number of iterations:

$$
\begin{aligned}
H_1(t) &= \begin{cases} \text{softmax}(U), & t = 0 \\ H_2(t-1), & 0 < t \leq T, \end{cases} \\
H_2(t) &= \begin{cases} f_\theta(U, H_1(t)), & 0 \leq t \leq T, \end{cases} \\
H_3(t) &= \begin{cases} 0, & 0 \leq t < T \\ H_2(t), & t = T. \end{cases}
\end{aligned}
\tag{3.6}
$$

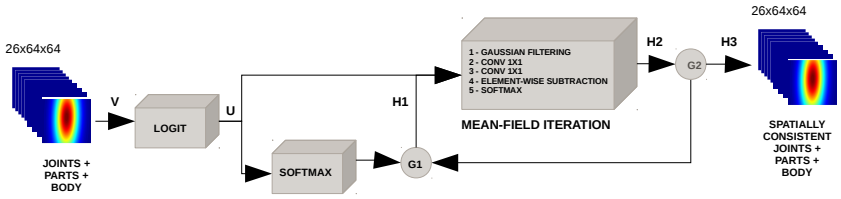


Figure 3.8: Spatial model architecture. The **mean-field iteration** module executes the mean-field update for T iterations. The additional modules pre-process the input unaries. Gates G1 and G2 switch outputs according to Eq. 3.6. References to the backward pass of the network are not included for simplicity.

Internally, the sequence of operations executed by the mean-field iteration module (e.g., Gaussian filtering, convolutions, element-wise subtraction, and softmax) correspond to the mean-field steps for CRF models defined by Eq. 3.5 (Krähenbühl et al., 2011; Kiefel et al., 2014; Zheng et al., 2015). We detail the full algorithm describing the main steps as follows:

Algorithm 1 Mean-field inference as a RNN in a part-based model.

$U_{u,p}(l) \leftarrow -\phi(y_{u,p} = l)$, with $l \in \{0, 1\}$ ▷ Initialising
$Z_{u,p} \leftarrow \sum_l \exp(U_{u,p}(l))$

$Q_{u,p}(l) \leftarrow \frac{1}{Z_{u,p}} \exp\left(U_{u,p}(l)\right)$ for all (u, p) ▷ Normalising

for T iterations **do**
  $\hat{Q}_{u,p}(l) \leftarrow 0$ ▷ Initialising
  for each neighbour p' of p $(p \sim p')$ **do**
    $\tilde{Q}^{(k)}_{u,p}(l) \leftarrow \sum_{v \in \mathcal{V}_{p'}} \mathbf{k}^{(k)}_{p,p'}(\mathbf{f}_{u,p}, \mathbf{f}_{v,p'}) Q_{v,p'}(l)$ for all Gaussian kernels k ▷ Message Passing
    $\check{Q}_{u,p}(l) \leftarrow \sum_{k \in K} \mathbf{w}^{(k)}_{p,p'} \tilde{Q}^{(k)}_{u,p}(l)$ ▷ Weighting Filter Outputs
    $\hat{Q}_{u,p}(l) \leftarrow \sum_{l' \in \{0,1\}} \boldsymbol{\mu}^{(k)}_{p,p'}(y_{u,p}, l') \check{Q}_{u,p}(l')$ ▷ Compatibility Transform
  end for
  $\breve{Q}_{u,p}(l) \leftarrow U_{u,p}(l) - \hat{Q}_{u,p}(l)$ ▷ Adding Unary Potentials
  $Q_{u,p} \leftarrow \frac{1}{Z_{u,p}} \exp\left(\breve{Q}_{u,p}(l)\right)$ ▷ Normalising
end for

Message passing. Message passing is achieved by convolving data-dependent Gaussian filters $\mathbf{k}^{(k)}_{p,p'}(\mathbf{f}_{u,p}, \mathbf{f}_{v,p'})$ to the marginal distributions $Q(l')$. In our method,

the spatial Gaussian kernels are computed taking into account the mean and covariance of the displacement between each pair of parts, learned with a maximum likelihood estimate. As the kernel weights are derived from the input data, no learning is required in this step. During the backpropagation, the error differentials with respect to the previous layer are sent backward through the same Gaussian kernels in reverse order.

Weighting the filter outputs. The relative weighting $\mathbf{w}_{p,p'}^{(k)}$ between the Gaussian kernels are learned in the RNN. This operation is executed using a 1×1 convolutional filter, leaving the dimension of the output unchanged w.r.t the input. The error differentials with respect to the weights of this layer are calculated in the same way as for conventional convolutional layers.

Compatibility Transform. In a similar fashion to the filter weighting of the previous step, the compatibility transform may be executed as a 1×1 convolution layer. The compatibility matrix $\boldsymbol{\mu}_{p,p'}^{(k)}(y_{u,p}, l')$ is learned taking into account the influence that each pair of parts have over each other w.r.t. the binary labels assignments. The error differentials are computed and backpropagated similarly to the previous layer.

Adding unary potentials and normalising. Finally, the output from the previous step is subtracted (element-wise) from the unary potentials $-\phi(y_{u,p})$. The error differentials at the output of this step are passed to both inputs by copying the differentials at the output with the appropriate sign. The normalisation step, similarly to the initialisation, is another Softmax layer with no parameters to be learned.

3.4 Experiments and Discussion

In this section, we evaluate the performance of the part detector and the spatial model, separately and together. Moreover, we compare our model with the state-of-the-art methods on the LSP (Johnson et al., 2010) and the MPII (Andriluka et al.,

2014) benchmarks, using the Percentage of Correct Keypoints (PCK) (Yang et al., 2011) and PCKh (Andriluka et al., 2014) metrics as described below.

3.4.1 LSP Dataset

The Leeds Sports Pose (LSP) (Johnson et al., 2010) and the Leeds Sports Pose Extended Training (LSPe) (Johnson et al., 2011) datasets are two well-know benchmarks for 2D human pose estimation. They are composed of 2,000 and 10,000 images, respectively, with roughly 150 pixels in length and containing sports scenes. Each image shows a single person with 14 manually annotated joints. We follow the protocol defined by Johnson et al. (2011) in which 1,000 images from the LSP dataset are used as the testing set, while 11,000 (1,000 from LSP and 10,000 from LSPe) are used as the training set. We refer to this setup simply as LSP in this thesis.

3.4.2 MPII Dataset

The MPII Human Pose dataset (Andriluka et al., 2014) has established a new level in comparison with the previous generation of benchmarks for 2D human pose estimation (Ferrari et al., 2008; Johnson et al., 2010; Johnson et al., 2011; Sapp et al., 2013). The MPII has largely extended the number of images, presenting approximately 25,000 images containing over 40,000 people in a great variety of poses and visual appearances. Moreover, it provides the rough human position in the image, the coordinates of a rectangle around the head, scale, the annotations for 16 joints and the information about the visibility of them. The images have variable sizes and can contain multiple people.

3.4.3 Metrics

We assess the performance of our models quantitatively using the standard PCK (Yang et al., 2011) and PCKh (Andriluka et al., 2014) metrics. These measures account for the percentage of joints in the test set which are correctly located by a 2D pose estimator, regarding a normalised threshold distance computed in pixels. To be considered as correctly located, a predicted joint must be within

the normalised threshold radius of the corresponding ground-truth coordinate. For the PCK metric, this normalisation is performed taking the size of the person's torso, whereas for the PCKh it is performed using the size of the head which is more stable in the image plane w.r.t. body pose variations.

3.4.4 Training Hyperparameters

In all experiments, we adopted the following parameters: mini-batch equal to 10; Adam optimiser (Kingma et al., 2014a) with learning rate equal to 10^{-5}; weight decay of 5×10^{-4} and no other form of regularisation; and network weights initialised using the robust initialisation proposed by He et al. (2015). We cropped a 256×256 image area, keeping the person of interest in the central position, reinforcing the focus on it using an extra input channel containing a central and fixed size 2D Gaussian (standard deviation equal to 15 pixels), which is concatenated to the input channels of the final module of the network. Regarding data augmentation, we randomly apply the following over the images: scaling $[0.5, 1.5]$, rotation $[-45°, 45°]$, horizontal flipping and RGB jittering (Krizhevsky et al., 2012). We have trained our network by adding the MPII training set to the LSP and LSP extended training sets, following related approaches (Bulat et al., 2016; Wei et al., 2016; Insafutdinov et al., 2016; Pishchulin et al., 2015; Belagiannis et al., 2017). At test time, we perform predictions over scales in $[0.5, 1.5]$, equally spaced with a step of 0.1, and sum them to obtain the final estimations. The variants of part-detector architecture and the contribution of the CRF model were evaluated over a single scale in a validation set. The architecture was implemented using the Caffe framework (Jia et al., 2014) and the experiments ran on an NVIDIA Titan X.

3.4.5 Multi-Level Body Part Detector Evaluation

To evaluate variants of the basic architecture illustrated in Fig. 3.5, and support our design choices, we have trained different arrangements of modules, with the parameters afore detailed. They are defined by bold capital letters, e.g., **J** represents one module that locates joints, **(J)(J+R)** represents two modules which predict

Ref. name	Arch.	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	Mean
A_1	**J**	92.0	77.8	64.4	52.4	69.0	65.0	53.2	67.7
B_1 SEQUENTIAL / WITH_JUST_FINAL_LOSS	**4×(J)**	92.6	81.2	72.2	63.2	79.2	75.2	65.4	75.6
B_2 WITH_JUST_FINAL_LOSS	**4×(J)**	93.2	81.4	69.6	61.2	80.0	75.0	63.6	74.9
B_3	**4×(J)**	93.2	81.4	73.0	66.6	80.6	76.4	68.6	77.1
C_1	**(J)+(J+P)+ (J+P+B)+(J+P+B)**	**95.0**	80.2	72.8	65.2	80.6	77.0	70.4	77.3
C_2 J_LOSSES_WEIGHTED {1,2,3,12}†	**(J)+(J+P)+ (J+P+B)+(J+P+B)**	94.6	**83.0**	**75.4**	65.8	79.2	77.8	72.0	78.3
D_1 J_LOSSES_WEIGHTED {1,2,3,12}†	**4×(J+P)**	94.8	82.6	73.8	66.0	79.4	77.4	74.2	78.3
E_1 J_LOSSES_WEIGHTED {1,2,3,12}†	**4×(J+P+B)**	92.8	78.4	68.4	60.0	77.0	72.8	70.6	74.3
F_1 J_LOSSES_WEIGHTED {1,2,3,12,24}†	**5×(J+P)**	93.6	82.8	73.2	**66.8**	**82.8**	**80.2**	73.2	**78.9**
G_1 J_LOSSES_WEIGHTED {1,2,3,12,24,48}†	**6×(J+P)**	93.0	81.6	72.0	64.4	81.2	78.0	**75.4**	77.9

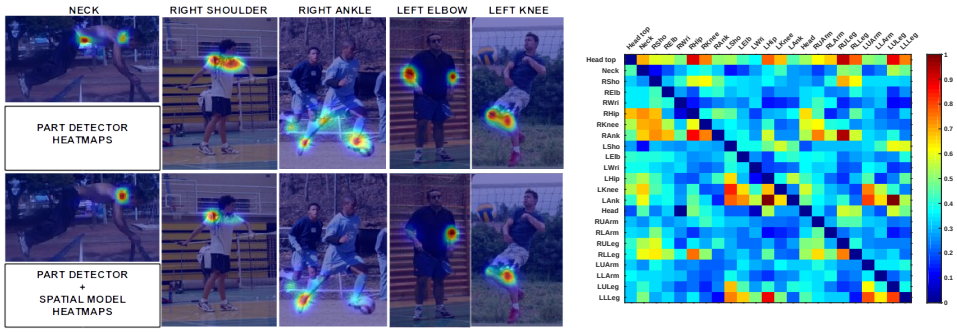
† Sets of relative weights of the modules losses.

Table 3.1: Comparison on the LSP dataset with PCK@0.2.

joints and joints along with rigid parts, respectively, and **4×(J+R+B)** denotes four modules predicting joints, rigid parts, and whole body. It is important to highlight that, except when stated, the variants are cascaded, and every module has a loss weight equal to 1. A summary of the variants is listed in Tab. 3.1, with their scores and reference names. In what follows, we analyse some of the evidence observed. Initially, the sets of architectures **A** and **B** not only show the benefits of multiple modules (**B_1** vs. **A_1**), but also show that cascaded modules with intermediate losses (**B_3**) are preferable to either sequential or cascade networks of the same capacity, but with just one final loss function (**B_1** and **B_2**). The set **C** corresponds to our basic architecture and shows evidence that the auxiliary body parts somehow improve the performance (**C_2** vs. **B_3**), particularly with the use of weights (**C_2** vs. **C_1**) attributing more importance to the joints and to final modules in the overall loss. To evaluate the effect of having the extra parts gradually added through the architecture, we employed sets **D** and **E**. The results suggest that it is equivalent to all modules predicting the same multi-level representation (**D_1** vs. **C_2**). Besides this, the prediction of the whole body does not seem to help in the location of the joints (**E_1**). We hypothesise that this happens because the person of interest is already centralised in most images of the LSP dataset. Finally, we have evaluated the addition of extra modules in sets **F** and **G**, finding the overall best performing architecture as **F_1**. We also tried other experiments over the sets, such as inverting the order of the modules in **C_1** and **C_2**, adding

Reference name	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	Mean
$\mathbf{F_1}$	93.6	82.8	73.2	66.8	**82.8**	**80.2**	73.2	78.9
$\mathbf{F_1}$+SM	**93.8**	**83.4**	**74.8**	**70.2**	81.8	77.0	**75.2**	**79.5**

Table 3.2: Comparison on the LSP dataset with PCK@0.2. SM = Spatial Model.



(a) **(b)**

Figure 3.9: (a) **Heatmaps from the LSP dataset.** The first row shows heatmaps obtained from the part detector ($\mathbf{F_1}$), while the second row shows heatmaps from the part detector plus the spatial model ($\mathbf{F_1}$+**SM**). (b) **The learned mean-field parameters.** The higher the compatibility value, the stronger the influence between the parts. We can notice small clusters of strong influence, particularly between parts on the same side of the body (left/right), which explains how we mitigate double-counting.

an extra background heatmap in the final modules and changing combinations of weights, but none of these improved the final performance.

3.4.6 Fully-Connected CRF Evaluation

Here we measure the contribution of the fully-connected CRF. The compatibility matrices were initialised as $\boldsymbol{\mu}^{(k)}_{p,p'} = \left[\begin{smallmatrix} 0 & 0 \\ 0 & 0 \end{smallmatrix}\right]$, denoting that the influence between the different body parts is initially null and will be learned during training. The kernel weights are set to $\mathbf{w}^{(k)}_{p,p'} = 1$, once we employ a single Gaussian kernel with spatial features (Eq. 3.4) in the present basic formulation of the model. We adopted the best performing architecture from Tab. 3.1 ($\mathbf{F_1}$) and compared its results with the ones obtained when the spatial model was employed in the second half of the training ($\mathbf{F_1}$+**SM**). Tab. 3.2 shows the improvements obtained over the part detector performance. The current formulation using only spatial information was already beneficial, as shown by the samples of body part heatmaps obtained with and without the use of the spatial model, in Fig. 3.9a. We can observe that in

Method	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	Mean
Chu et al.	98.1	93.7	89.3	86.9	93.4	94.0	92.5	92.6
Wei et al.	97.8	92.5	87.0	83.9	91.5	90.8	89.9	90.5
Our model	97.6	92.6	86.5	83.2	91.7	91.0	89.1	90.2
Insafutdinov et al.	97.4	92.7	87.5	84.4	91.5	89.9	87.2	90.1
Pishchulin et al.	97.0	91.0	83.8	78.1	91.0	86.7	82.0	87.1
Belagiannis et al.	95.2	89.0	81.5	77.0	83.7	87.0	82.8	85.2

Table 3.3: Comparison of PCK@0.2 score on the LSP test set. Only methods trained on MPII and LSP jointly.

Figure 3.10: Sample pose predictions for the LSP dataset.

fact, the heatmaps are refined when prior knowledge about the body is taken into consideration. The learned mean-field parameters are shown in Fig. 3.9b, i.e. the compatibility values, since the kernel weights were kept fixed due to the single spatial kernel employed. The learned compatibility values $\boldsymbol{\mu}_{p,p'}^{(k)}$ of all the pairwise matrices represent how much each body part on the horizontal axis influences all other parts on the vertical axis.

3.4.7 Results on LSP

Tab. 3.3 shows our final results on the LSP test set, after 200 epochs, along with comparable approaches from the literature. We have competitive results in comparison to the state-of-the-art. We show qualitative results in Fig. 3.10 and

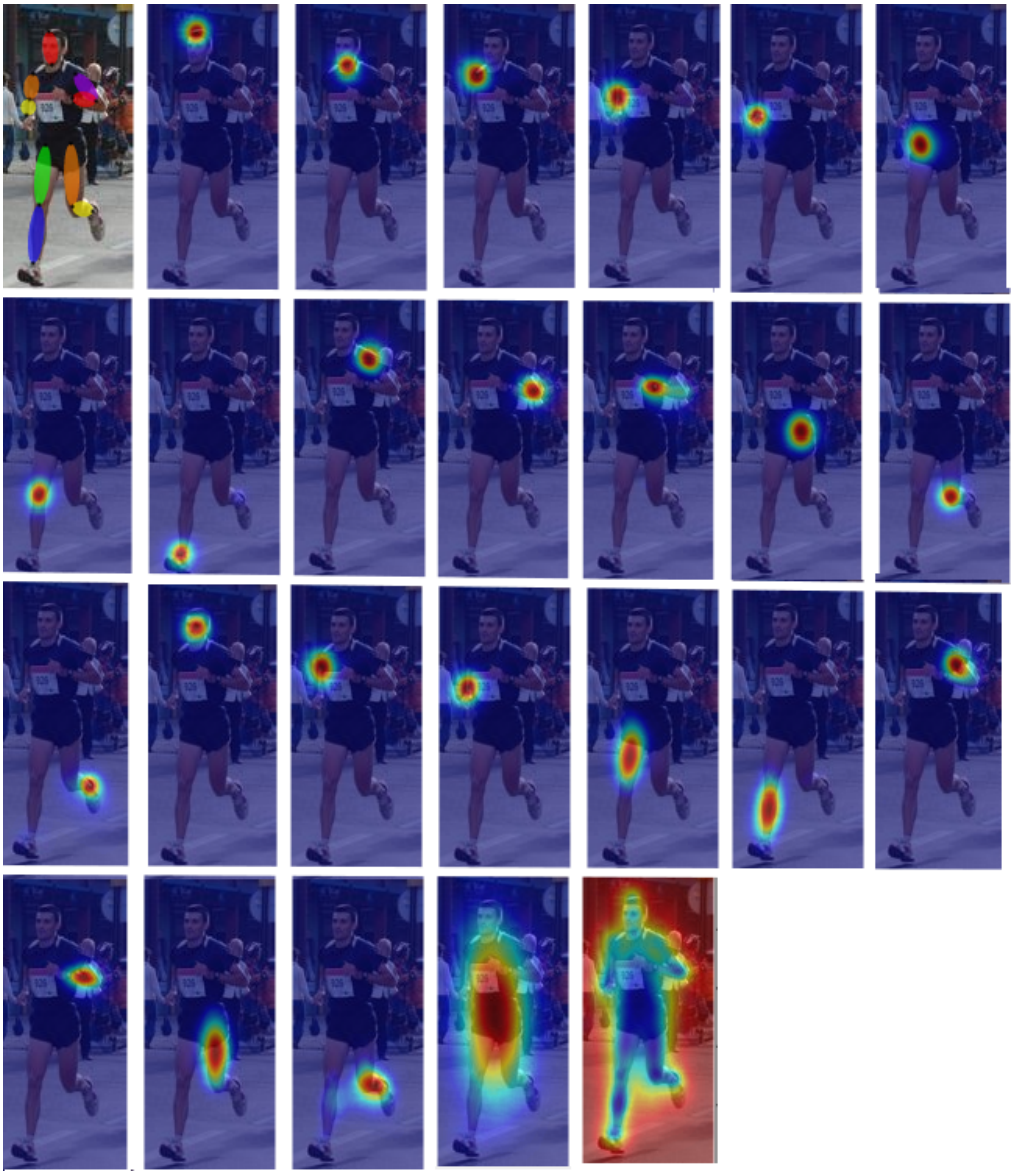


Figure 3.11: Samples of the heatmaps predicted for a person running from the LSP test set. From the top left to the bottom right we have: original images with the limbs superimposed, heat top, neck, right shoulder, right elbow, right wrist, right hip, right knee, right ankle, left shoulder, left elbow, left wrist, left hip, left knee, left ankle, head, right upper arm, right lower arm, right upper leg, right lower leg, left upper arm, left lower arm, left upper leg, left lower leg, whole body. A background heatmap is shown for illustration.

heatmaps samples in Fig. 3.11. Moreover, when the presence of false positives generated by clutter or by unusual poses prejudices the bottom-up methods, we may outperform such approaches, as illustrated in Fig. 3.12. The stronger inductive bias towards the correct positions of the joints, obtained with the use of the rigid

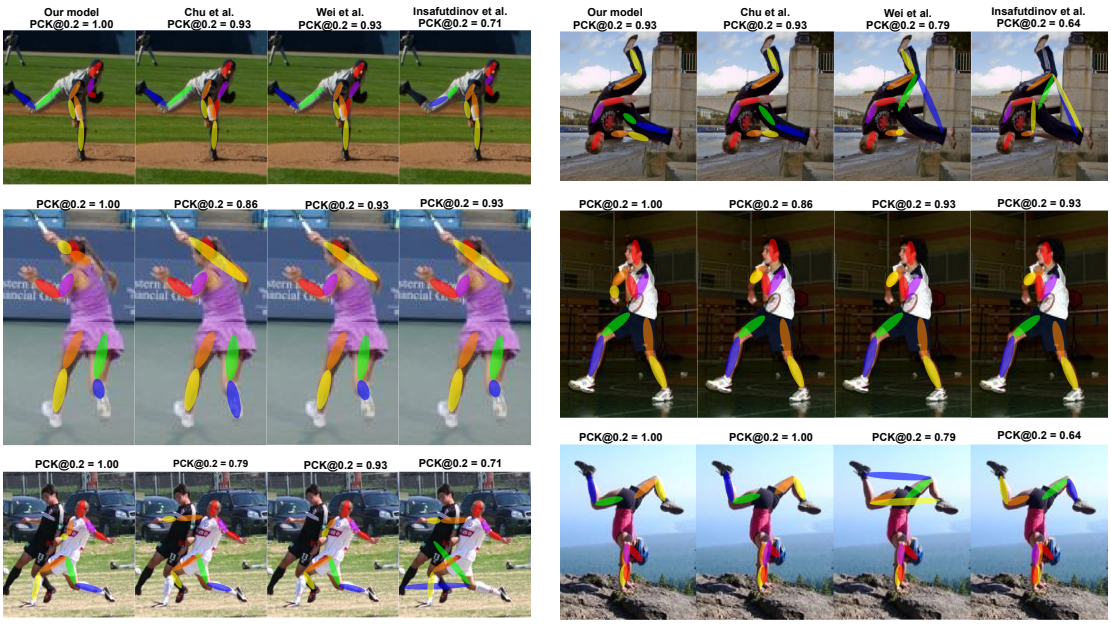


Figure 3.12: Cases from LSP in which we outperformed state-of-the-art bottom-up methods prejudiced by false positives caused by clutter and unusual poses.

parts in our multi-level representation, along with the high level of redundancy of the fully-connected CRF, facilitates the correct estimations in these cases.

3.4.8 Results on MPII

Tab. 3.4 shows results on the MPII test set, in this case employing just the MPII training data for 250 epochs. We have again shown competitive scores, and an ability to perform well in the presence of occlusions and unusual poses as illustrated in Fig. 3.13. Heatmaps samples are shown in Fig. 3.14 and failure cases from both datasets are shown in Fig. 3.15.

3.4.9 Overall Analysis

Our method is able to cope with challenging poses, outperforming state-of-the-art methods in particular cases. From the analysis of our fully-connected CRF model, summarised in Fig. 3.9, we can observe that the spatial model contributes mostly to the distinction between left and right parts of the body, whatever ambiguity is generated by clutter, occlusions or unusual poses, e.g., upside-down postures, as illustrated in Fig. 3.12. Such uncertainties are mitigated by data-driven compatibility matrix values (Fig. 3.9b), which show that parts on the same side of the body

Method	Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	Mean
Ke et al.	98.5	96.8	92.7	88.4	90.6	89.3	86.3	92.1
Chu et al.	98.5	96.3	91.9	88.1	90.6	88.0	85.0	91.5
Newell et al.	98.2	96.3	91.2	87.1	90.1	87.4	83.6	90.9
Bulat et al.	97.9	95.1	89.9	85.3	89.4	85.7	81.7	89.7
Insafutdinov et al.	96.8	95.2	89.3	84.4	88.4	83.4	78.0	88.5
Our model	97.7	95.0	88.1	83.4	87.9	82.1	78.7	88.1
Rafi et al.	97.2	93.9	86.4	81.3	86.8	80.6	73.4	86.3
Gkioxari et al.	96.2	93.1	86.7	82.1	85.2	81.4	74.1	86.1
Lifshitz et al.	97.8	93.3	85.7	80.4	85.3	76.6	70.2	85.0
Hu et al.	95.0	91.6	83.0	76.6	81.9	74.5	69.5	82.4
Tompson et al.	96.1	91.9	83.9	77.8	80.9	72.3	64.8	82.0
Carreira et al.	95.7	91.7	81.7	72.4	82.8	73.2	66.4	81.3
Tompson et al.	95.8	90.3	80.5	74.3	77.6	69.7	62.8	79.6
Pishchulin et al.	74.3	49.0	40.8	34.1	36.5	34.4	35.2	44.1

Table 3.4: Comparison of PCKh@0.5 score on the MPII test set. Only methods trained on MPII dataset.

Figure 3.13: Sample images from the MPII dataset showing contexts in which we successfully handle challenging cases, such as occlusions.

exert a strong influence over each other. High compatibility values between parts that would not be directly connected in simpler tree-structured models, e.g., hips and ankles, also show that the fully-connected model is learning more complex relations between the body parts. Although we have presented a basic and general formulation of our spatial model, based only on the 2D locations of the parts, it has already shown marginal improvement over the part detector alone (Tab. 3.2). Additional features, such as the colours of mirrored body parts, may be naturally incorporated, potentially improving, even more, the performance in relation to the bottom-up part detector. Finally, in our experiments, the CRF model achieved

Figure 3.14: Samples of the heatmaps predicted for a person standing in front of the camera from the MPII test set. From the top left to the bottom right we have: original image with the limbs superimposed, heat top, neck, thorax, pelvis, right shoulder, right elbow, right wrist, right hip, right knee, right ankle, left shoulder, left elbow, left wrist, left hip, left knee, left ankle, head, torso, right upper arm, right lower arm, right upper leg, right lower leg, left upper arm, left lower arm, left upper leg, left lower leg, whole body. A background heatmap is shown for illustration.

convergence in a small number of iterations ($T = 3$) for the recurrent mean-field update, as similarly reported by Zheng et al. (2015). Despite that, training and testing with the fully-connected model are costly. With our best-performing model (Tabs. 3.1 and 3.2), when the CRF model is added to the CNN part-detector, training time per image increases 6 times, from 0.3s to 1.8s, approximately, while testing time per image increases 5 times, from 1.5s to 7.5s, approximately (testing is performed over multiple scales, as previously mentioned in this section). For the final models, the total training time was approximately 7 days. As an alternative, this high cost might be diminished by the elimination of *weak* pairwise relations, i.e. the ones with small values in the compatibility matrix (Fig. 3.9b). In this case, the fully-connected model would function as an intermediate step towards a simpler yet still data-driven model structure.

Figure 3.15: Sample images from the MPII and LSP dataset in which the method could not overcome the ambiguities in the poses.

3.5 Conclusions

In this chapter, we have introduced a novel framework composed of a multi-level appearance representation of the human body, along with a loopy fully-connected part-based model, conveying information about the spatial structure of the body. We proposed a cascade CNN body part detector to obtain our appearance representation, and an RNN to perform mean-field inference on the part-based model, defined as a binary CRF. We evaluate the components of our architecture, showing that the multi-level representation, composed of body parts with different granularities, as well as the spatial model, facilitate the location of the body joints. Our experiments on the MPII and LSP benchmarks have shown competitive results. In particular cases, when the presence of false positives greatly prejudices exclusive bottom-up strategies, we outperformed the state-of-the-art methods. The current deep learning framework may be naturally extended to other problems, such as object detection and per-pixel labelling of objects parts, and also by the addition of extra kernels, based on other features such as colour. As a further improvement, we intend to generalise our spatial model to handle multimodal distributions, a limitation imposed on the current formulation by the use of Gaussian pairwise kernels, which are efficient to compute (Adams et al., 2010).

4

A Conditional Deep Generative Model of People in Natural Images

In the present chapter, we propose a deep generative model of humans in natural images, which keeps 2D pose separated from other latent factors of variation, such as background scene and clothing. In contrast to methods that learn generative models of low-dimensional representations, e.g., segmentation masks and 2D skeletons, our single-stage end-to-end conditional-VAEGAN learns directly on the image space. The flexibility of this approach allows the sampling of people with independent variations of pose and appearance. Moreover, it enables the reconstruction of images conditioned on a given posture, allowing, for instance, pose-transfer from one person to another. We validate our method on the Human3.6M dataset (Ionescu et al., 2014), also performing experiments on the DeepFashion (Liu et al., 2016b) and the ChictopiaPlus (Lassner et al., 2017) benchmarks, achieving state-of-the-art results on the latter. Our model, named Conditional-DGPose, outperforms the closest related work in the literature. It generates more realistic and accurate images regarding both body posture and image quality, learning the underlying factors of pose and appearance variation.

4.1 Introduction

The analysis of visual data containing humans is a central problem in computer vision. In this context, the body posture plays a major role in the process of understanding humans in images and videos (Jhuang et al., 2013; Kleinsmith et al., 2007; Zhang et al., 2013). Modelling and learning how natural images of people in particular postures are generated are important and challenging tasks. It has many applications, such as the generation of synthetic images and the reenactment of body movements in videos (Balakrishnan et al., 2018).

Over the last years, great attention has been given to discriminative models of human pose, especially for 2D pose estimation (Chu et al., 2017; de Bem et al., 2018; Newell et al., 2016; Tompson et al., 2014; Wei et al., 2016). Meanwhile, generative models have attracted considerably less interest. This fact may be partially explained by the relatively higher complexity of the latter, mainly due to a large number of random variables involved in the process, as well as the frequently intractable probability distributions over them.

Lately, however, the general research interest for generative models has increased, greatly driven by deep learning methods, such as Variational Autoencoders (VAEs) (Kingma et al., 2014b; Rezende et al., 2014) and Generative Adversarial Networks (GANs) (Goodfellow et al., 2014). These approaches have introduced strategies to tackle inference and stochastic learning in the context of intractable probabilistic computations and large datasets.

Generative models for human pose defined over low-dimensional *pose-spaces* are well-known from the literature (Fleet, 2011; Pons-Moll et al., 2011; Taylor et al., 2010a). Also in more recent works with deep generative models, in particular with VAEs, low-dimensional spaces are adopted, e.g., pose-spaces (Walker et al., 2017) or segmentation masks (Lassner et al., 2017). Once such generative models are learned, a common approach is to use image-to-image translation (Isola et al., 2017) to map the representations to natural images. These multi-stage training and testing processes often reduce the accuracy and control over image generations w.r.t. appearance and body posture (Lassner et al., 2017). In other cases, as

in (Ma et al., 2017b), it prevents sampling, which is a core characteristic and desirable capability of generative models.

To overcome the aforementioned limitations of related works, we learn a deep generative model of natural images, directly on the high-dimensional image space. We explicitly represent the body pose and other latent factors of variation (or just *visual appearance*) as separated and independent random variables in a unified single-stage end-to-end probabilistic model. In this way, we can generate samples from our generative probability distribution by manipulating either pose or appearance directly and independently. Our conditional-VAEGAN architecture, a principled variational method for approximate Bayesian inference, allows us to have a structured and interpretable latent space, by means of a conditional-VAE framework (Sohn, 2015), associated with a discriminator module (Larsen et al., 2016), which takes advantage of the high-quality image generations from GANs.

To our knowledge, our approach is the first deep generative model capable of generating realistic natural images of people in a unified probabilistic framework, while keeping the body posture and appearance as explicitly separated and interpretable variables. The advantage of that is threefold, as it allows: i) to change the posture of a person in an image, given a conditioning pose (pose-transfer); ii) the sampling from the generative distribution with independent control over pose and visual appearance; iii) the direct and more accurate control over appearance and pose by means of a unified single-stage end-to-end model.

We have validated our method qualitatively on the Human3.6M dataset (Ionescu et al., 2014). Such experiments demonstrate that our model generates realistic images with direct, accurate and independent manipulation of pose and appearance, successfully performing *image reconstruction*, *pose-transfer* and *sampling* with a single network. Moreover, our approach achieves state-of-the-art results on the ChictopiaPlus benchmark, outperforming the closest related work in the literature, the ClothNet-Body network (Lassner et al., 2017), and showing that our model generates more realistic and accurate images w.r.t. both body posture and image quality, while it learns the underlying latent factors of pose and appearance variation.

4.2 Related Work

Deep Generative Models. VAEs (Kingma et al., 2014b; Rezende et al., 2014) and GANs (Goodfellow et al., 2014) have both received great attention in the last years. Both of these deep learning methods are capable of tackling inference and stochastic learning in the context of intractable probabilistic distribution and large datasets. Due to their popularity, several variants of them have been proposed in the literature (Larsen et al., 2016; Mescheder et al., 2017; Wan et al., 2017). In particular, Larsen et al. (2016) propose the association of VAEs' capability of explicitly modelling latent probability distributions, with GANs' high-quality image generations. Another useful variant of VAEs for conditional generative models are the conditional-VAEs (CVAEs), proposed and applied to image segmentation by Sohn (2015). Additionally, conditional-VAEGANs (CVAEGANs) were employed by Bao et al. (2017) for image inpainting and data augmentation.

Recently, other authors (Kingma et al., 2014c; Siddharth et al., 2017) have shown how VAEs can be used to learn structured disentangled representations in the latent space by enforcing partial supervision for a subset of latent variables. Different from previous works, our method is the first to employ a CVAEGAN with a structured, interpretable and disentangled latent space for generating people in natural images.

Deep Generative Models of Humans. Despite the great interest in human pose estimation in the past years, generative models have been far less investigated compared to discriminative approaches. The closest related approach to ours is the one by Lassner et al. (2017), which presents a generative model based on CVAEs for clothes of segmented people conditioned on pose. However, their generative model works on low-dimensional segmentation masks, and an image-to-image translation network (Isola et al., 2017) is used to render the natural images. The segmentation masks (*sketches*) are generated either with a VAE (*ClothNet-full*) or with a CVAE (*ClothNet-body*). The simultaneous, direct, and accurate manipulation of pose and appearance is limited because it is done by two separate networks in a two-stage process. In contrast, we learn the generative model directly on the real images using only pose as a conditioner and without the need of body parts' segmentation.

Another related work is the image-to-image translation model proposed by Ma et al. (2017b). It uses the U-Net-like model from Quan et al. (2016), and no VAE nor CVAE are employed. As in the method of Lassner et al. (2017), the one by Ma et al. (2017b) is trained in two stages, as the authors acknowledge that it is difficult for a complete end-to-end framework to cope with both correct poses and appearances simultaneously. However, training is done using pairs of images from the same person in different poses, views and scales, since the approach is designed strictly for pose-transfer. Differently, the Conditional-DGPose accomplishes pose-transfer as a by-product of our formulation. Moreover, it can be performed between images from different people, since it keeps pose and appearance as disentangled random variables. This relevant difference in our CVAEGAN modelling makes our method more general than the ones aforementioned. For instance, it also allows the direct sampling of natural images conditioned on a given pose. Lassner et al. (2017) can sample segmentation masks (*sketches*) and render them as natural images with the image-to-image model (*portray module*) from Isola et al. (2017). In the image-to-image translation from Ma et al. (2017b), sampling is not possible at all.

Furthermore, Walker et al. (2017) introduce a hybrid VAEGAN architecture for forecasting future poses in a video. Here, a low-dimensional pose representation is learned using a VAE, and once the future poses are predicted, they are mapped to images using a GAN generator. We use a discriminator in our training to improve the quality of the generated images, following Larsen et al. (2016). However, this does not affect our probabilistic generative distribution, and neither does it compromise our capability of sampling from it. Finally, considering GAN based generative models, Tulyakov et al. (2018) present a GAN network that can learn motion and content in two separate latent spaces in an unsupervised manner. However, it does not allow explicit manipulation over the human pose.

4.3 Deep Variational Autoencoders

In this section, we briefly review variational autoencoders and their relevant variations on which our generative model of people in natural images is based.

Variational Autoencoders. VAEs (Kingma et al., 2014b; Rezende et al., 2014) are a class of deep generative models that simultaneously train both a probabilistic *encoder* and a *decoder*, given a training set $\mathcal{D}$ with elements $\mathbf{x} \in \mathcal{D}$. The main idea is that an encoding $\mathbf{z}$ is considered as a latent variable, and the objective is to maximise the likelihood $p(\mathbf{x}) = \int p_\theta(\mathbf{x}|\mathbf{z})\, p(\mathbf{z})\, d\mathbf{z}$. The decoder (referred to as the *generative network*) defines the conditional probability $p_\theta(\mathbf{x}|\mathbf{z})$ and the prior over $\mathbf{z}$ is assumed to be the standard normal distribution.

In a high-dimensional space, finding the decoder parameters θ that maximise the likelihood is intractable. To tackle this, VAEs use a variational method that approximates the posterior distribution $p_\theta(\mathbf{z}|\mathbf{x})$ using an encoder (referred to as the *inference* or the *recognition network*) $q_\phi(\mathbf{z}|\mathbf{x})$. This approximate posterior is assumed to be a Gaussian whose parameters are the output of a neural network parameterised by ϕ. Under these assumptions, the generative and the inference networks are trained jointly by performing stochastic gradient ascent on the Evidence Lower Bound (ELBO),

$$\begin{aligned} \log p_\theta(\mathbf{x}) &\geq \mathcal{L}_{\text{VAE}}(\phi, \theta; \mathbf{x}) \\ &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log \frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})}\right] \\ &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \mathrm{KL}\left[q_\phi(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})\right]. \end{aligned} \tag{4.1}$$

Here, the first term denotes the expectation over the approximate posterior distribution, which measures the decoder accuracy (or reconstruction error), while the KL-divergence term encourages the approximate posterior to be close to the prior $p(\mathbf{z})$. Computing the expectation in Eq. (4.1) involves sampling, which can be circumvented using a re-parameterisation trick (Kingma et al., 2014b). Furthermore, since the approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ and the prior $p(\mathbf{z})$ are both Gaussian distributions, the KL-divergence can be computed in closed form. The structures of the generative and the recognition models in a standard VAE are shown in Fig. 4.1a. At test time, only the generative network (decoder) is retained, and one can easily generate samples by sampling a latent variable $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and passing it through the decoder.

(a) VAE **(b)** CVAE

Figure 4.1: Structure of recognition (top row) and generative (bottom row) models for VAE and CVAE. Here, $\mathbf{z}$ denotes the unobserved latent variables and $\mathbf{y}$ denotes the conditioning variables.

Conditional Variational Autoencoders. CVAEs (Sohn, 2015) are a simple extension of the standard VAEs that allow more flexibility in the generative process. In a CVAE, both the input data $\mathbf{x}$ and the latent variable $\mathbf{z}$ are conditioned on $\mathbf{y}$. It means that both, the encoder and the decoder, are now conditioned on $\mathbf{y}$, i.e. the corresponding distributions can be written as $q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})$ and $p_\theta(\mathbf{x}|\mathbf{z},\mathbf{y})$. In this case, the objective function can be written as

$$\begin{aligned}\log p_\theta(\mathbf{x}|\mathbf{y}) &\geq \mathcal{L}_{\text{CVAE}}(\phi,\theta;\mathbf{x},\mathbf{y}) \\ &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})}\left[\log \frac{p_\theta(\mathbf{x},\mathbf{z},\mathbf{y})}{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})}\right] \\ &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})}\left[\log p_\theta(\mathbf{x}|\mathbf{z},\mathbf{y})\right] - \mathrm{KL}\left[q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})\|p_\theta(\mathbf{z}|\mathbf{y})\right].\end{aligned} \tag{4.2}$$

Note that the KL-divergence is between $q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})$ and $p_\theta(\mathbf{z}|\mathbf{y})$, where both the distributions are now conditioned on $\mathbf{y}$. See Fig. 4.1b for the generative and recognition models of a CVAE. In this work, our CVAE is conditioned on pose, as detailed in the subsequent section.

Conditional-VAEGANs. Note that by the factorisation of the generative model, VAEs necessitate the specification of an explicit likelihood function $p_\theta(\mathbf{x}|\mathbf{z})$, which can often be difficult. GANs, on the other hand, attempt to sidestep this requirement by learning a surrogate to the likelihood function. Here, the generative model $p_\theta(\mathbf{x},\mathbf{z})$, viewed as a mapping $G : \mathbf{z} \mapsto \mathbf{x}$, is setup in a two-player minimax game with a "discriminator" $D : \mathbf{x} \mapsto \{0,1\}$, whose goal is to correctly identify

if a data point $\mathbf{x}$ came from the generative model $p_\theta(\mathbf{x}, \mathbf{z})$ or from the true data distribution $p(\mathbf{x})$. Such an objective is defined as

$$\begin{aligned}\mathcal{L}_{\text{GAN}}(D, G) = &\mathbb{E}_{p(\mathbf{x})}\left[\log D(\mathbf{x})\right] \\ &+ \mathbb{E}_{p_\theta(\mathbf{z})}\left[\log(1 - D(G(\mathbf{z})))\right].\end{aligned} \tag{4.3}$$

Importantly, implicitly learning an approximation to the likelihood function $p_\theta(\mathbf{x}|\mathbf{z})$ can result in a much higher quality of generated data, particularly for the visual domain (Karras et al., 2018). Thus, bringing together these two different approaches, similarly to Larsen et al. (2016) and Bao et al. (2017), our single objective combines both, the CVAE and the GAN objectives, directly as

$$\mathcal{L} = \mathcal{L}_{\text{CVAE}} + \mathcal{L}_{\text{GAN}}. \tag{4.4}$$

4.4 Our Approach

Considering the previous section, in our Conditional-DGPose model, we define $\mathbf{x}$ to be a fully observable random variable correspondent to an RGB image. The random variable $\mathbf{z}$ is unobservable, and it corresponds to all latent factors that affect the generation of an image, except for the body pose of the person in it. We may refer to $\mathbf{z}$ as *appearance*, yet it is an oversimplified definition. As to the body pose, it is represented by the interpretable and fully observable random variable $\mathbf{y}$. Following, we detail our method.

4.4.1 Conditional-DGPose

We have tested several variations of deep architectures, culminating with the one shown in Fig. 4.2. On the latter, all the probability distributions $q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{y})$, $p_\theta(\mathbf{x}|\mathbf{z}, \mathbf{y})$ and $p_\theta(\mathbf{z}|\mathbf{y})$, as well as the discriminator correspond to deep CNN modules. Implementation details are provided in Tabs. 4.6 and 4.5, Sec. 4.A (chapter appendix). In the rest of this section, we describe training and testing phases. However, we start detailing the adopted pose representation, as it sets the basis for understanding the other topics in the section.

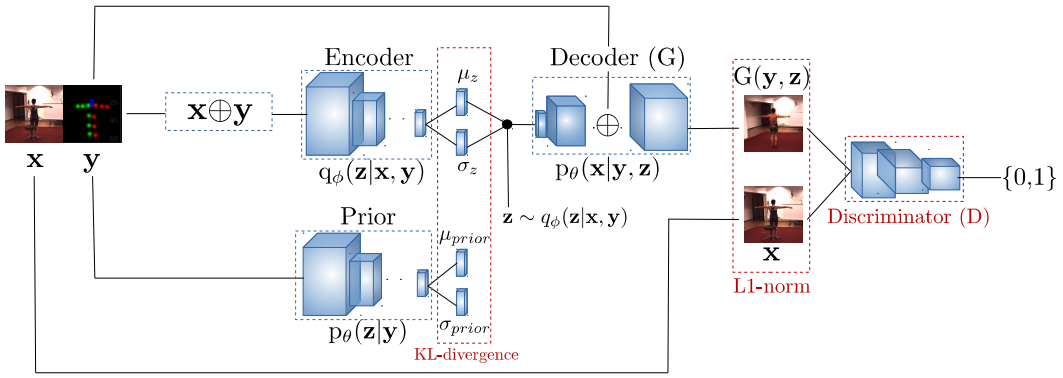
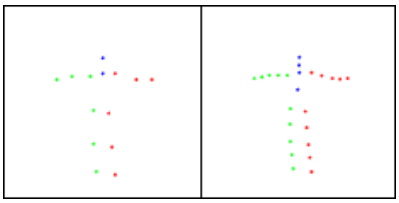


Figure 4.2: Conditional-DGPose architecture. At the training, the Encoder receives $\mathbf{x} \oplus \mathbf{y}$ as input and learns the posterior $q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})$. The Prior module receives $\mathbf{y}$ alone and learns the distribution $p_\theta(\mathbf{z}|\mathbf{y})$. Appearance is sampled $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})$, using the re-parameterisation trick (Kingma et al., 2014b), and passed to the Decoder, as well as the conditioning pose $\mathbf{y}$, which is concatenated to the Decoder feature maps. The Decoder then generates a reconstructed image $G(\mathbf{y},\mathbf{z})$. The loss function (see Eq. 4.4, Sec. 4.3) is composed by the following terms, highlighted in red: the L1-norm $L1(\mathbf{x}, G(\mathbf{y},\mathbf{z}))$ which is computed between the original and the reconstructed image; the KL-divergence $\mathrm{KL}[q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})||p_\theta(\mathbf{z}|\mathbf{y})]$, which is used to regularise the posterior distribution; and the GAN Discriminator cross-entropy loss, used to learn how to discern between real and generated images.

Pose Representation

In our model, the random variable $\mathbf{y}$ corresponds to the body pose; therefore, a suitable representation of the human body must be adopted. As mentioned in our literature review, many methods which define a generative model in the *pose space* would simply encode J joints defining the body as a vector, such that $\mathbf{y} \in \mathcal{R}^{2J}$. Others employ extended versions of it, in which positions of R rigid parts and B whole body are derived from the annotated joints (Yang et al., 2011), such that $\mathbf{y} \in \mathcal{R}^{2(J+R+B)}$. Both cases are illustrated in Fig. 4.3.



(a) (b)

Figure 4.3: Vector representation. (a) $J = 14$ joints which compose a 2D pose vector; (b) extended 2D vector composed by 24 body parts ($J = 14$ annotated joints, $R = 9$ intermediate points between joints and $B = 1$ central point).

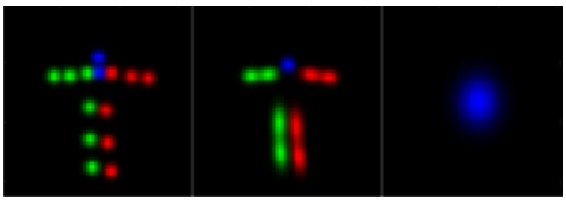
It is not evident whether a sparse 2D vector representation would be capable of conveying the spatial information required for reconstructing realistic natural images, taking into account the human body pose. In fact, related works suggest the opposite (Lassner et al., 2017; Ma et al., 2017b). On the other hand, the mapping of 2D joints positions to heatmaps has shown to be very effective in several pose estimation approaches (Chu et al., 2017; de Bem et al., 2018; Newell et al., 2016; Tompson et al., 2014; Wei et al., 2016). The Gaussian heatmaps represent the underlying probability distribution of body parts' locations.

After an experimental evaluation (see Sec. 4.5.1), we have followed the pose representation introduced in Sec. 3.3.1, such that the heatmap representation consists of P body elements, in a way that $\mathbf{y} \in \mathcal{R}^{P \times H \times W}$, where H and W are the heatmap height and width, respectively. In the simplest case $P = J$, however, as the set of joints is fairly sparse, to cover the entire area of the bodies, joints, rigid parts and the whole body might be used as an extended case, in which $P = J + R + B$, as illustrated in Fig. 4.4. In this way, each body element p is represented using a 2D Gaussian around its centre $\boldsymbol{\mu}_p = (i_p, j_p)$, with diagonal covariance matrix $\Sigma_p = R_p \begin{bmatrix} \sigma_{p,i}^2 & 0 \\ 0 & \sigma_{p,j}^2 \end{bmatrix} R_p^\top$, computed as follows:

Joints. Since joints have a limited spatial extent, we follow previous approaches (Chu et al., 2017; Newell et al., 2016; Tompson et al., 2014; Wei et al., 2016) in modelling them as isotropic Gaussians that are centred at the ground-truth joint location and have a small standard deviation (e.g. $\sigma_{p,i} = \sigma_{p,j} = 1.5$ pixel for a 64×64 heatmap).

Rigid Parts. The centre $\boldsymbol{\mu}_p$ of a rigid part p is defined as the mean point of the centres $\boldsymbol{\mu}_k$ and $\boldsymbol{\mu}_l$ of the joints it connects. We orient the Gaussian representing the rigid part to align its i axis with the line connecting $\boldsymbol{\mu}_k$ and $\boldsymbol{\mu}_l$. We define $\sigma_{p,i}$ to be proportional to $|\boldsymbol{\mu}_k - \boldsymbol{\mu}_l|$, and set $\sigma_{p,j} = \kappa_p \sigma_{p,i}$, where κ_p is a part-specific ratio, inspired by anthropometric measurements (NASA, 1995).

Body. The body centre is defined to be the mean of the annotated joint centres. Principal component analysis (PCA) of the joint centres is used to obtain the orientation of the body in the image plane. We define $\sigma_{p,i}$ and $\sigma_{p,j}$ to be proportional to the distance between the extreme projections of the joint centres onto, respectively, the principal and secondary axes of variation.



(a) (b) (c)

Figure 4.4: Heatmap representation. Total of 24 heatmaps: (a) 14 from the annotated joints, (b) 9 corresponding to rigid parts and (c) 1 corresponding to the whole body. Right, left and central body elements are represented by the colours green, blue and red, respectively, in the person-centric representation. Heatmaps of the same kind are gathered and superimposed on a black background to facilitate visualisation.

Training

The recognition (encoder) and the generative (decoder) networks conditioned on pose $\mathbf{y}$ are defined by $q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})$ and $p_\theta(\mathbf{x}|\mathbf{z},\mathbf{y})$, respectively. In practice, the Gaussian heatmap labels (conditioning pose) are concatenated to the correspondent input image $\mathbf{x}$ at the encoder's 1st layer; and concatenated to $\mathbf{z}$ at the decoder's 7th layer (see Tab. 4.6, Sec. 4.A). In both layers, the feature maps match the heatmaps dimensions. This design option was particularly important for the decoder, since the heatmaps effectively *guided the attention* (Li et al., 2018) of the network towards the position and area of body parts, improving reconstructions. Additionally, the heatmap labels $\mathbf{y}$ alone are the input for the prior module, which learns the distribution $p_\theta(\mathbf{z}|\mathbf{y})$. Finally, the reconstructed image denoted by $G(\mathbf{y},\mathbf{z})$ and the training image $\mathbf{x}$ are used as input of the discriminator module, which learns to distinguish the real from the reconstructed images.

Following Sec. 4.3, the loss function $\mathcal{L} = \mathcal{L}_{\text{CVAE}} + \mathcal{L}_{\text{GAN}}$ (Eq. 4.4) minimised during training is composed of: i) reconstruction loss L1-norm$(\mathbf{x}, G(\mathbf{y},\mathbf{z}))$, between the input image $\mathbf{x}$ and the reconstruction $G(\mathbf{y},\mathbf{z})$; ii) the closed-form KL-

divergence KL $[q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})\|p_\theta(\mathbf{z}|\mathbf{y})]$, between the recognition and the prior Gaussian distributions, respectively, $q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})$ and $p_\theta(\mathbf{z}|\mathbf{y})$; and iii) the discriminator cross-entropy loss $\mathcal{L}_{\text{GAN}}$ (Eq. 4.3).

Testing

Due to the inherent versatility of generative models, our architecture may be employed in different ways, according to the intended task. Thus, the testing stage is divided into *reconstruction*, *pose-transfer* and *sampling*.

Reconstruction. Here, since $\mathbf{x}$ and $\mathbf{y}$ are given, the prior module and the discriminator are not employed. For reconstruction, when an image $\mathbf{x}_1$ and its corresponding pose $\mathbf{y}_1$ are given as input, the reconstructed image $G(\mathbf{y}_1,\mathbf{z}_1)$ is obtained as the decoder output, is illustrated in Fig. 4.5. See sample results in Fig. 4.10.

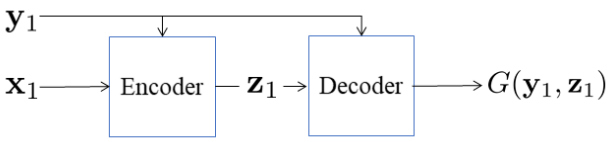


Figure 4.5: Conditional-DGPose reconstruction at test time.

Pose-transfer. Similarly to the reconstruction process, the prior module and the discriminator are not employed for pose-transfer. However, in this case, when $\mathbf{x}_1$ is used as input along with a different target pose $\mathbf{y}_2$, the person in the reconstructed image will keep the appearance of $\mathbf{x}_1$, along with the body pose defined by $\mathbf{y}_2$, as illustrated in Fig. 4.6. See a sample result in Fig. 4.11).

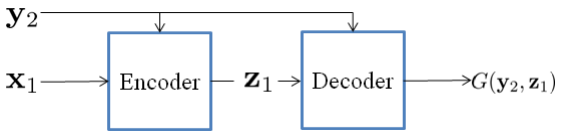


Figure 4.6: Conditional-DGPose pose-transfer and manipulation at test time.

Sampling. In the sampling case, the encoder and the discriminator modules are employed during test. Moreover, as expected, no RGB image is given as input. Instead, only a conditioning pose $\mathbf{y}$ is given as the input to the prior module, which defines $p_\theta(\mathbf{z}|\mathbf{y})$. From this prior distribution, the latent appearance $\mathbf{z}$ is sampled and used as input of the decoder network, as illustrated in Fig. 4.7. In this manner, for a given pose, different appearances can be randomly generated from the learned model, as shown in the sample results in Fig. 4.12.

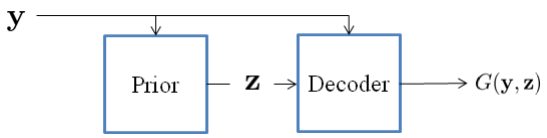


Figure 4.7: Conditional-DGPose sampling at test time.

4.5 Experiments and Discussion

We validate our method on the Human3.6M dataset (Ionescu et al., 2014) and achieve state-of-the-art results on the ChictopiaPlus benchmark (Lassner et al., 2017), outperforming the closest related work in the literature, the ClothNet-Body network (Lassner et al., 2017). In the following sections, we describe in detail the evaluation performed with each one of the datasets, as well as the employed metrics and hyperparameters.

4.5.1 Human3.6M Dataset

The Human3.6M dataset (Ionescu et al., 2014) contains 3.6 million images acquired by recording 5 female and 6 male actors performing a diverse set of motions and poses corresponding to 15 activities, under 4 different viewpoints. We follow the standard protocol and use sequences of two actors as our test set, while the rest of the data is used for training. A subset of 14 (out of 32) body joints represented by their 2D image coordinates is adopted as ground-truth data, while minor body parts are neglected (e.g., fingers). Due to the high frequency of the video acquisition (50Hz), out of images from all 4 cameras, we subsample frames in time, producing

subsets for training and test, with 317,989 and 1,280 images, respectively. The images, with an original resolution of 1000×1000 pixels, are cropped to 64×64 and grouped in mini-batches of 64 samples.

4.5.2 ChictopiaPlus Dataset

The ChictopiaPlus dataset (Lassner et al., 2017) is an extension of the Chictopia dataset (Liang et al., 2015). It augments the original per-pixel annotations for body parts with pose annotation (Insafutdinov et al., 2016), 3D shape (Loper et al., 2015) and facial segmentation. In contrast to the Human3.6M dataset, in which each actor always wears the same outfit, it contains $23,011$ training, $2,913$ validation and $2,873$ test images of segmented people (without background) dressed in a great variety of clothes. All the images have a resolution of 286×286 pixels.

4.5.3 DeepFashion Dataset

The DeepFashion dataset (In-shop Clothes Retrieval Benchmark) (Liu et al., 2016b) consists of 52,712 images of people in a variety of clothing and poses. We follow Ma et al. (2017b), using their joints' annotations obtained with an off-the-shelf pose estimator (Cao et al., 2017), and divide the dataset into training (44,950 images) and test (6,560 images) subsets. Images with corrupted pose estimations were suppressed, with all original images having 256×256 pixels. Note that we aim to learn a complete generative model of people in natural images, which is significantly more complex, compared to models focusing on a particular task, such as pose-transfer. For this reason, we do not restrict our training set to pairs of images of the same person and use individual images, in contrast to Ma et al. (2017b) and Siarohin et al. (2018). Instead, we use the 44,950 training images and the 6,560 test images individually.

4.5.4 Training Hyperparameters

The following parameters were commonly adopted through all experiments: Adam optimiser (Kingma et al., 2014a) with learning rate equal to 10^{-4}; weight decay of 5×10^{-4}; network weights initialised randomly for fully-connected layers and with robust

initialisation (He et al., 2015) for convolutional and transposed-convolutional layers. We crop an image area keeping the person of interest in the central position. Images were normalised to present zero-mean and unit-variance, and no other form of data augmentation or preprocessing was employed. Implementation was done using the Caffe framework (Jia et al., 2014) and the experiments ran on an NVIDIA Titan X.

4.5.5 Metrics

Quantitative evaluation of generative models is inherently difficult (Theis et al., 2015), and usually, a great deal of emphasis is placed on the qualitative evaluation of reconstructed (generated) samples. Since our model explicitly represents *appearance* and *body pose* as separate variables, we evaluate this two aspects independently with appropriate metrics: i) **Image quality** is evaluated using the standard Peak Signal-to-Noise Ratio (PSNR) and the Structural Similarity Index (SSIM) (Wang et al., 2004) metrics. ii) **Accuracy of the reconstructed poses** is evaluated using a protocol introduced by us as follow. To set a common ground for comparing an original test set with a reconstructed one, we start using a well-established (discriminative) human pose estimator (Newell et al., 2016), initially estimating all 2D poses in the original test set. In our protocol, we assume that such estimations are the *ground-truth* poses of the test set. Subsequently, we apply the same discriminative estimator over the reconstructed test images, produced by the trained generative models. Finally, we use the PCK metric (Yang et al., 2011), described in Sec. 3.4.3, which computes the percentage of 2D joints correctly located by the pose estimator. Thus, we assume that any degradation in the PCK metric is caused by imperfections on the reconstructed images, since a PCK score of 100% would correspond to having all the estimated joints, in the original and the reconstructed images, at the same locations, up to the distance threshold. We illustrate this metric in Fig. 4.8. Related works do not evaluate the accuracy of the generated poses directly but only the overall reconstruction quality either by using standard SSIM (Ma et al., 2017b) or the IoU based score, which is specific to the setup of (Lassner et al., 2017), based on the reconstruction of segmentation

masks. In summary, our PCK metric evaluation measures reconstructions accuracy explicitly considering the generated poses.

(a) (b)

Figure 4.8: Accuracy of the reconstructed poses. Samples illustrating best and worst pose reconstructions on the Human3.6M dataset. Each pair of images shows the pose estimation over the original image (left) and the reconstructed image (right). The estimated joints are connected by lines for visualisation purposes. Right limbs, left limbs, and head are shown, respectively, by green, red and blue lines. (a) It illustrates the best possible reconstructed poses, with PCK@0.5 = 1.00. (b) It illustrates the worst possible reconstructed poses, with PCK@0.5 = 0.00. All images are 64×64 pixels.

4.5.6 Results on Human3.6M

The Human3.6M benchmark presents images in a controlled environment. Thus, it is adopted as an initial dataset for qualitative evaluation. We tested different pose representations and architectures on this dataset. Such experiments supported and guided our design options towards the use of heatmaps instead of 2D pose vectors, since reconstructions were better in the former case, as shown in Tab. 4.1 and Fig. 4.9; as well as towards the use of residual blocks in our encoders (Liu et al., 2017), which have improved our reconstructions. On the other hand, we have not observed the benefits of using residual blocks in the decoder.

Pose representation	**L1-Norm**
Vector (14 joints)	14.52
Vector-Extended (28 joints)	13.91
Heatmaps (14 joints)	13.55
Heatmaps-Extended	
⌞ (14 joints + 9 rigid parts + whole body)	**13.41**

Table 4.1: The average reconstruction errors were obtained with our architecture using L1-norm on the Human3.6M test set.

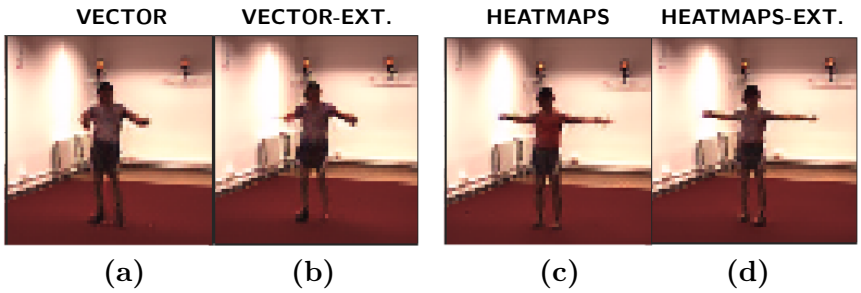


Figure 4.9: Samples of reconstructed images, obtained with: **(a)** 2D vector of joints (Vector); **(b)** 2D vector of joints; rigid parts and body (Vector-Extended); **(c)** heatmap representations of joints (Heatmaps); **(d)** heatmap representations of joints, rigid parts and body (Heatmaps-Extended). We highlight the difficulty for capturing the spatial extent of some body parts, particularly extremities far from the torso, when the vector representations are adopted. In this example, the use of joints' heatmaps is already sufficient to improve the reconstruction. However, the extended version (with rigid parts and body) makes the model more robust to more complex poses.

As part of our initial qualitative evaluations, we have tested our Conditional-DGPose on the three tasks mentioned in Sec. 4.4.1, namely *reconstruction*, *pose-transfer*, and *sampling*. Initially, in Fig. 4.10, we show our heatmap pose representation along with reconstructions, to demonstrate that realistic images with accurate poses can be generated. Furthermore, we illustrate *pose-transfer* in Fig. 4.11 and *sampling* in Fig. 4.12, in which the separation between pose and appearance is made evident by the independent change of each variable.

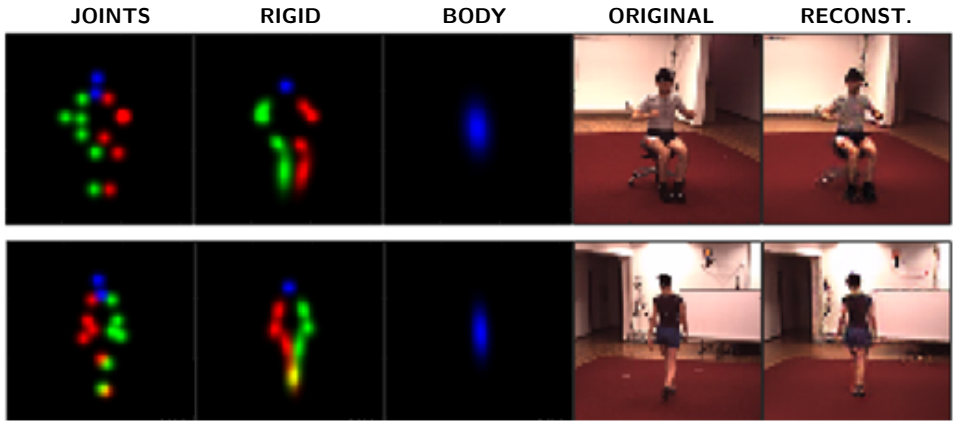


Figure 4.10: Human3.6M reconstructions. From the left to right: the joints, rigid parts and body heatmaps; the original image; and the reconstructed image. See Fig. 4.5 for details on the reconstruction process. In the heatmaps, right parts are shown in green, left parts in red and central parts in blue. Human3.6M images are 64×64 pixels.

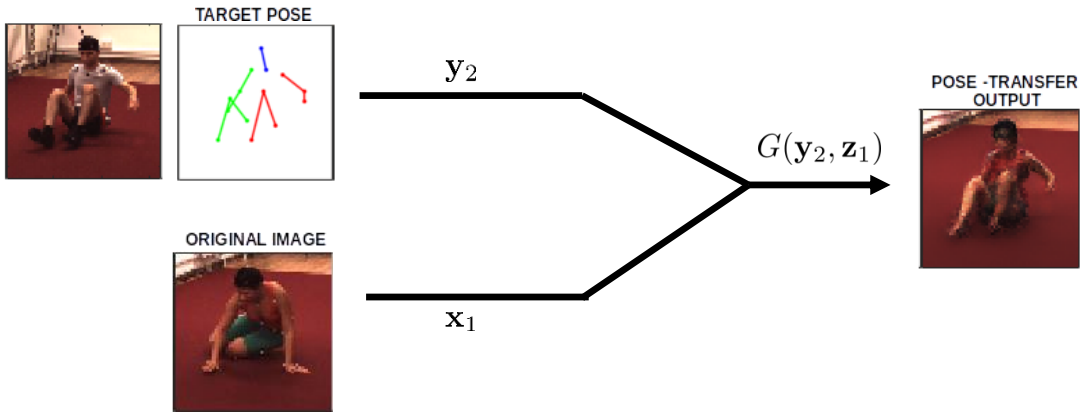


Figure 4.11: Pose-transfer. The original image $\mathbf{x}_1$ (64×64) and the target pose $\mathbf{y}_2$ are combined into the reconstructed image $G(\mathbf{y}_2, \mathbf{z}_1)$, in which the target pose has been transferred yet keeping the appearance of the original image. The target pose is used as a conditioner in the generation of the reconstructed image. The pose is illustrated with a skeleton to facilitate visualisation, even though we use the heatmap representation. We refer to Fig. 4.6 for details on the pose-transfer process.

Next, we stress the pose-transfer and compositionality capabilities of the model, pushing it beyond what is usually done in related methods. Regarding *pose-transfer*, we demonstrate the capability of our model to learn pose and appearance as separate variables, which allows direct control over the two at test time. To this end, we generate images in which we maintain the appearance of the input image; however, the generated person is "moved" into the required target pose. The target pose may be composed manually, extracted from another image with an off-the-shelf pose estimator, or provided interactively by a user. This is illustrated in Fig. 4.13, where we employed target poses from LSP dataset (Johnson et al., 2010), that have completely different poses in a drastically different environment compared to our training set. Our generative model could disentangle pose and appearance and generate images with poses that do not exist in the training data.

Concerning manipulation, we show in Fig. 4.14 how our model can be used to "compose" images that have never been seen in the training data. For instance, we can generate images with multiple people in the same (replicated) pose simply by conditioning on a respective heatmap. In fact, we can go one step further and generate an image where all people are in the same pose, but, e.g. one of them is *shorter* and another *thinner*, as shown in Fig. 4.15a. In an extreme case, we can

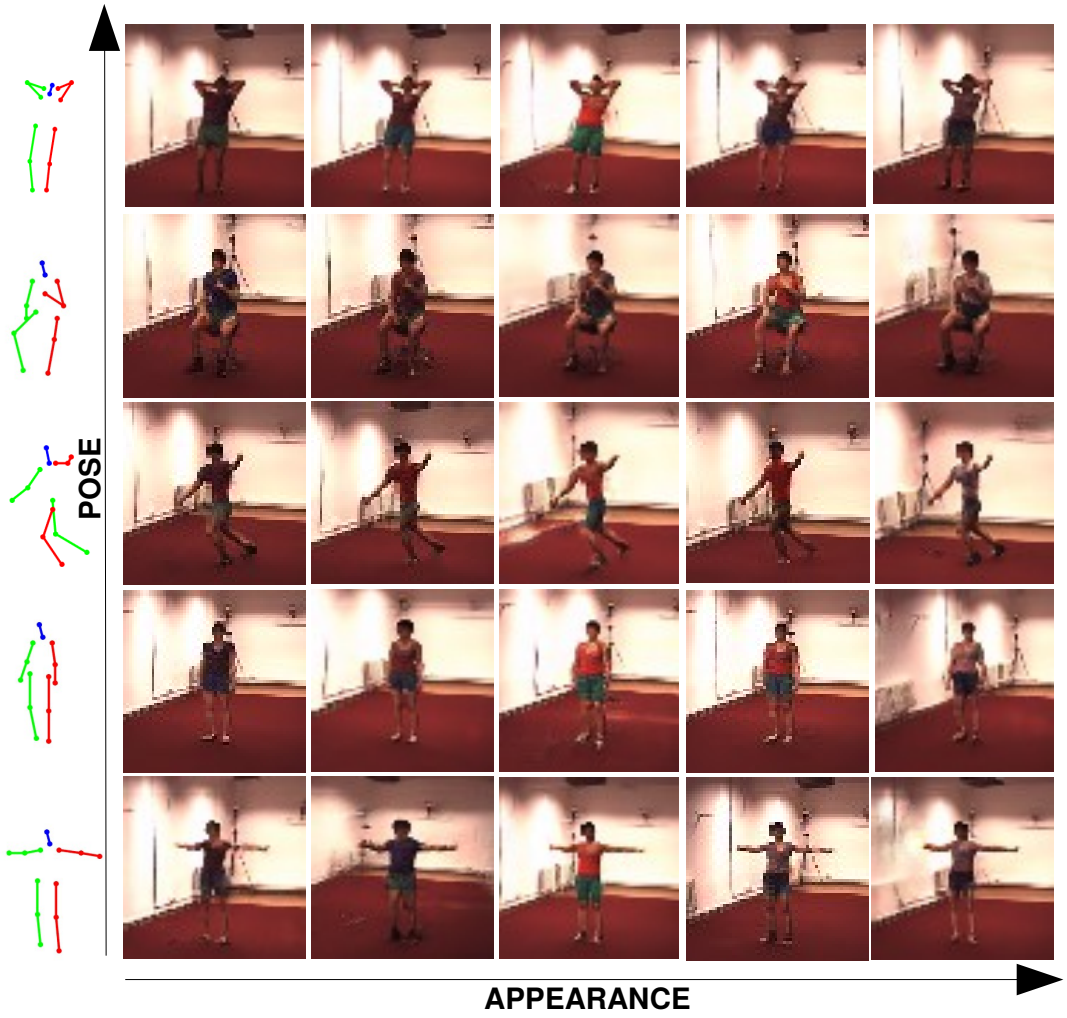


Figure 4.12: Sampling. Random samples of people in given poses. As the two axes show, pose and appearance can be independently manipulated. The pose is illustrated with a skeleton to facilitate visualisation, even though we use the heatmap representation. Images have 64×64 pixels.

even generate "unreal" images containing only certain body parts (e.g. heads) or disconnecting them from the rest of the body, as in Figs. 4.15b and 4.15c, respectively. Note that the training dataset is composed of only single person images; thus, the model has never seen an image with multiple people or only some separate body parts. This suggests that the learned latent space of our model is indeed disentangled.

Figure 4.13: Cross-domain pose-transfer. Here we illustrate the pose-transfer capability of our Conditional-DGPose. On the leftmost column, we show test images from the LSP dataset (Johnson et al., 2010), along with their corresponding ground-truth 2D pose annotations, composed by 14 joints (see Sec. 3.4.1 for details on the LSP dataset). These are conditioners (target-poses) for the generation of the reconstructions, shown from the third to the rightmost column. As can be observed, the target-poses are transferred to the validation images, while the latter maintain their original appearances. We highlight the fact that neither the LSP images nor their poses were part of the training set.

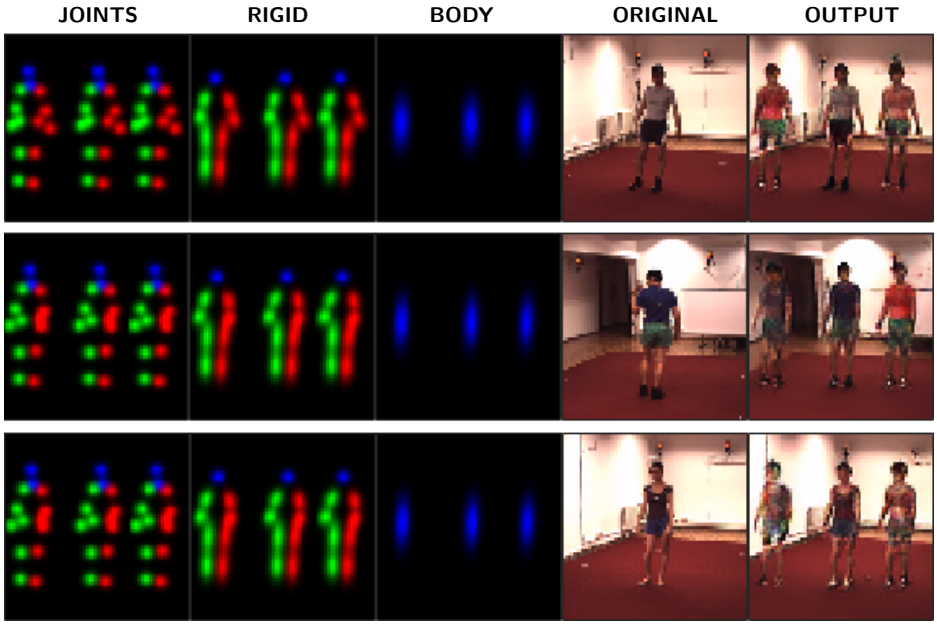


Figure 4.14: Hallucinating multiple people. The Conditional-DGPose model was trained with images containing only one person. The original image in each set is the input, and the output is generated conditioned on the heatmap pose representation.

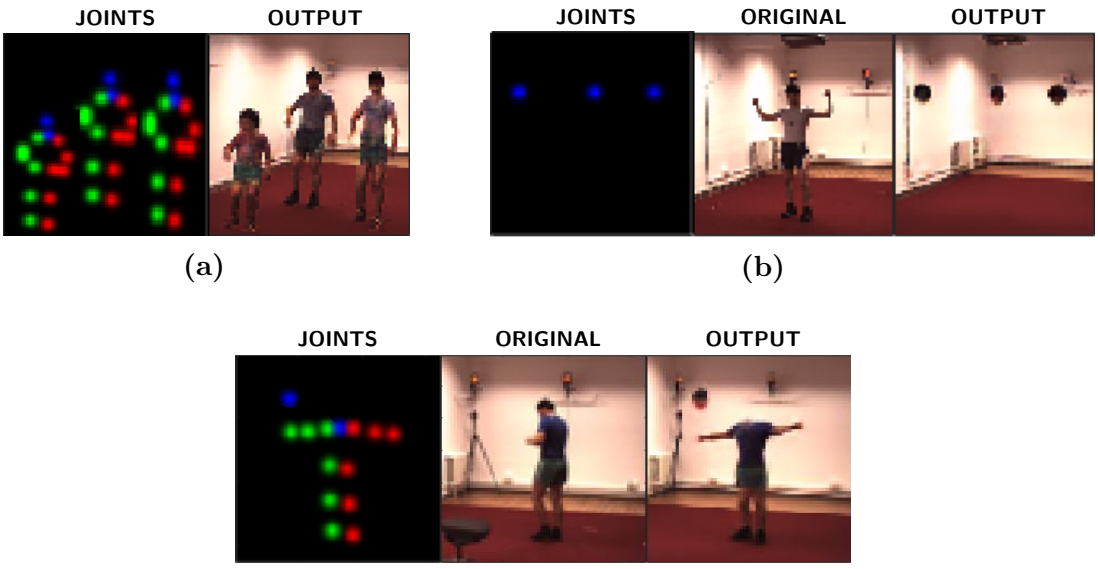


Figure 4.15: Generating "unreal" images. We illustrate the versatility of the model extrapolating the generation of images to unseen scenes. (**a**) A sampled image in which the pose representation in the centre was manually translated and scaled, producing two new bodies: one shorter and chunkier (*left*), and one taller and thinner (*right*). (**b**) A reconstructed image in which all the body parts were suppressed, except the head. (**c**) Pose-transfer in which the position of the head was manually changed, disconnecting it from the rest of the body. Heatmaps of rigid parts and whole body are not shown for simplicity.

4.5.7 Results on ChictopiaPlus

We employ this benchmark to compare our approach with the closest related work in the literature, the ClothNet-body by Lassner et al. (2017). In order to do so, we use the trained models made publicly available by the ClothNet-body authors. We perform quantitative and qualitative comparisons, detailed below, showing that we outperform Lassner et al. (2017) w.r.t. image quality and body pose reconstructions.

Quantitative Results. Regarding the PCK metric, our model reports 95.14% of accuracy, with PCK score at 0.5, and outperforms Lassner et al. (2017) by a large margin, which reports 70.89%, as shown in Fig. 4.16. Moreover, our approach also outperforms Lassner et al. (2017) w.r.t. the image quality, as can be seen in Tab. 4.2, which reports the PSNR and the SSIM scores for both methods. The results demonstrate good quality of reconstructions w.r.t. the human pose, suggesting that

our Conditional-DGPose model benefits from the single-stage end-to-end approach, in contrast to the multiple stages of training and testing from Lassner et al. (2017).

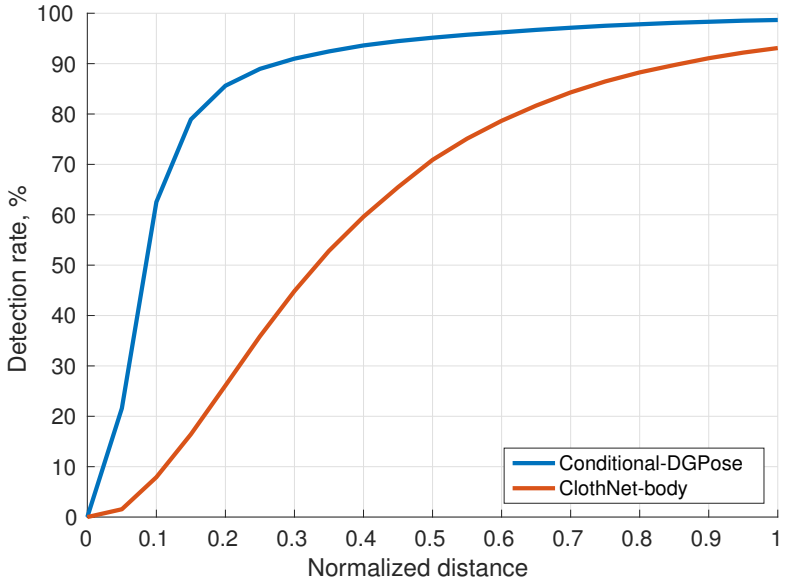


Figure 4.16: ChictopiaPlus. The PCK scores over reconstructed images of our Conditional-DGPose (*blue*) significantly outperforms the ClothNet-body (Lassner et al., 2017) (*red*). Detection rate represents the percentage of joints corrected relocated in the reconstructed images.

	PSNR	SSIM
Conditional-DGPose (ours)	**21.33**	**0.88**
ClothNet-Body	16.89	0.82

Table 4.2: Quantitative evaluation w.r.t. image quality. Our method outperforms the ClothNet-Body (Lassner et al., 2017) considering both metrics, the PSNR and the SSIM.

Qualitative results. In Fig. 4.17, we compare images generated by the Conditional-DGPose to the ones by the ClothNet-body (Lassner et al., 2017). We note that, while both approaches are capable of generating people in the required poses, our approach performs better in terms of appearances, which are much closer to the original in our case. Even though both methods were able to generate visually good images and poses, the Conditional-DGPose was more accurate in capturing the locations of the body parts, particularly regarding limbs' extremities. This shows that even without the image-to-image translation network, our method was able to generate realistic images. In addition to that, in Fig. 4.18, we present qualitative sampling results from our model to demonstrate that it generates realistic images with accurate poses.

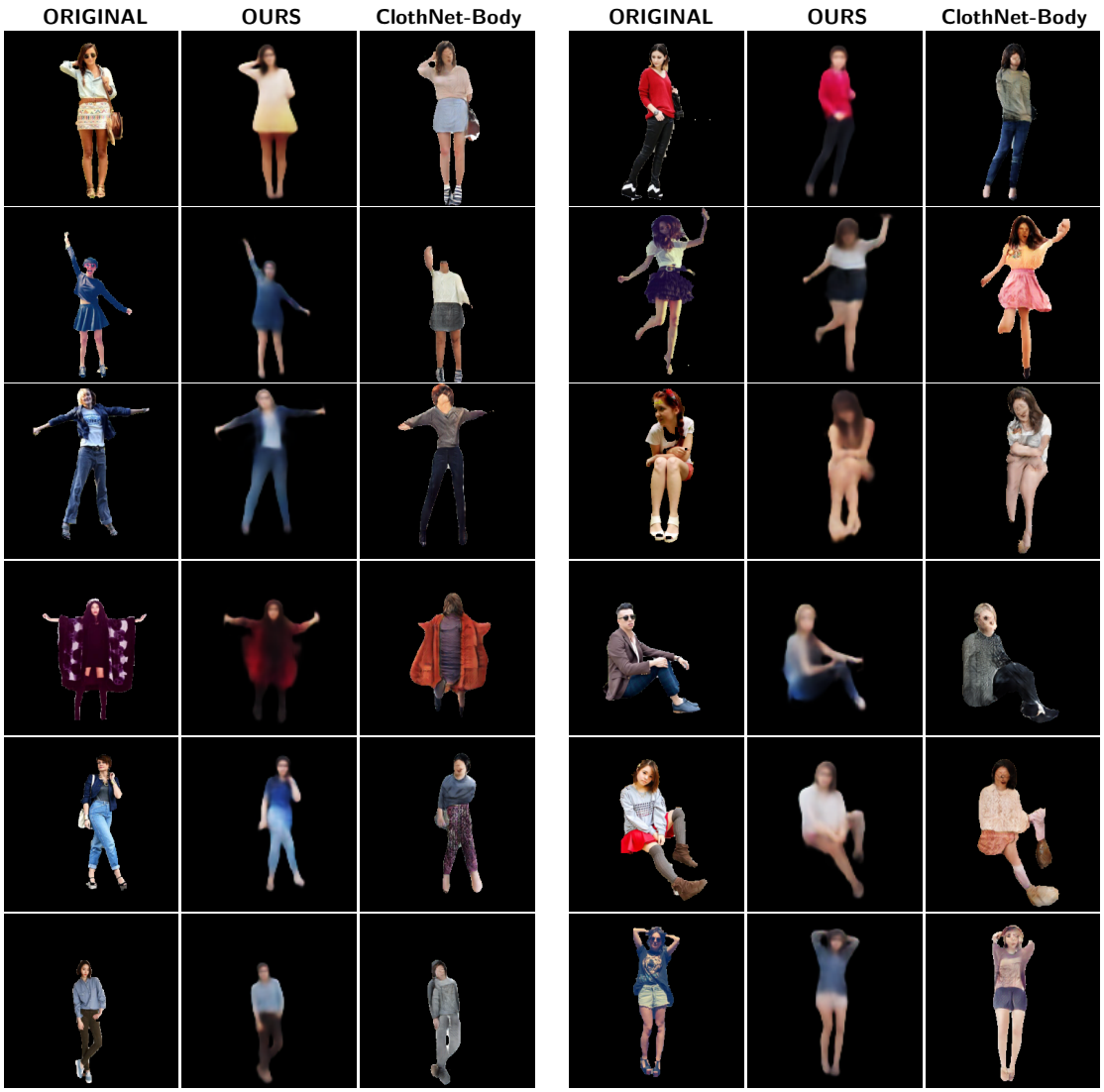


Figure 4.17: Reconstructions. In each row, respectively: original image (256×256), Conditional-DGPose, and ClothNet-body (Lassner et al., 2017) reconstructions. The images generated by our model are much closer to the originals in terms of appearance (colours). Moreover, in general, the Conditional-DGPose captures the body parts' locations more accurately, which results in better quantitative results w.r.t. the pose reconstruction, shown in Fig. 4.16. Limbs' extremities are frequently lost in the ClothNet-body (Lassner et al., 2017) reconstructions. Best viewed if zoomed in digital version.

4.5.8 Results on DeepFashion

Here we show qualitative and quantitative experiments on the DeepFashion dataset (Liu et al., 2016b). The baseline on this dataset is the image-to-image pose guided generation (PG^2) by Ma et al. (2017b). We use their same training and test sets. However, as our model is not an image-to-image translation architecture, we do not use pairs of images for training.

(a) **(b)**

Figure 4.18: (a) Random samples from the Conditional-DGPose model for a fixed pose (leftmost image). (b) Random samples from the ClothNet-body (Lassner et al., 2017) for a fixed pose (leftmost image), which are rendered as natural images by an image-to-image translation network (Isola et al., 2017) over the segmentation masks sampling.

Quantitative results. Again, we employ the PSNR and the SSIM metrics to evaluate image quality, and the PCK metric to provide a quantitative evaluation of pose reconstructions, as described previously (see Sec. 4.5.2). In Table 4.3, we initially show that even not being trained on images pairs and tackling the significantly more complex task of learning a generative model, instead of executing image-to-image translation, our method achieves scores only slightly below the ones by the PG^2 network on image reconstruction. A similar observation can be done regarding pose reconstruction, since our model reports 74.94% of accuracy, with PCK score at 0.5, against 78.27% from Ma et al. (2017b). The overall PCK curve is shown in Fig. 4.19.

	PSNR	SSIM
Conditional-DGPose (ours)	18.38	0.79
PG^2	**18.96**	**0.83**

Table 4.3: DeepFashion. Quantitative evaluation w.r.t. image quality, showing that our method presents a performance only slightly below the PG^2 baseline (Ma et al., 2017b), considering both metrics, the PSNR and the SSIM, despite the fact it tackles a significantly more complex task than image-to-image translation.

Qualitative results. Concretely, the learning of a full generative model, instead of image-to-image translation, allows for the execution of tasks, such as sampling from the learned latent space, which are just not feasible with architectures purely trained on image pairs. To illustrate this, in Fig. 4.20 we traverse the appearance

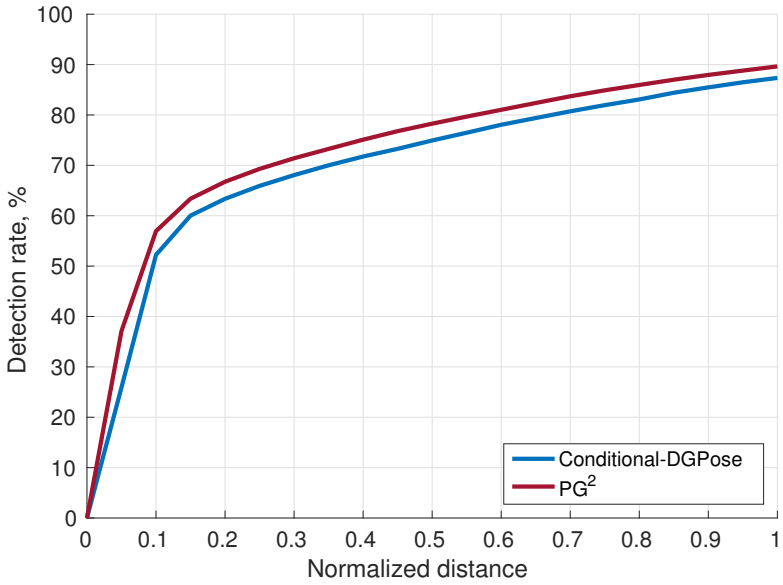


Figure 4.19: DeepFashion. The PCK scores over reconstructed images of our Conditional-DGPose (*blue*) performs only slightly below the PG^2 network (Ma et al., 2017b) the (*red*), despite the fact it is tackling a significantly more complex problem than image-to-image translation. Detection rate represents the percentage of joints corrected relocated in the reconstructions.

manifold learned on the DeepFashion dataset. Using only our heatmap pose representation as input, for a given pose, we smoothly vary the values of the latent appearance representation, generating samples with different visual aspect for the same body posture. Such kind of direct sampling is not feasible with the PG^2 architecture (Ma et al., 2017b). Finally, we show qualitative comparisons in Fig. 4.21. As observed, the quality of our reconstructions is clearly comparable with the ones from (Ma et al., 2017b).

Figure 4.20: Conditional-DGPose Appearance Manifold. Illustration of the appearance manifold learned on the DeepFashion dataset. We smoothly traverse the manifold for a given pose, causing changes in the visual appearance of the person in the image. No image is used as input, only our heatmap pose representation, evidencing that a truly generative model for natural images was learned, in which pose and appearance are disentangled. Best viewed if zoomed in digital version.

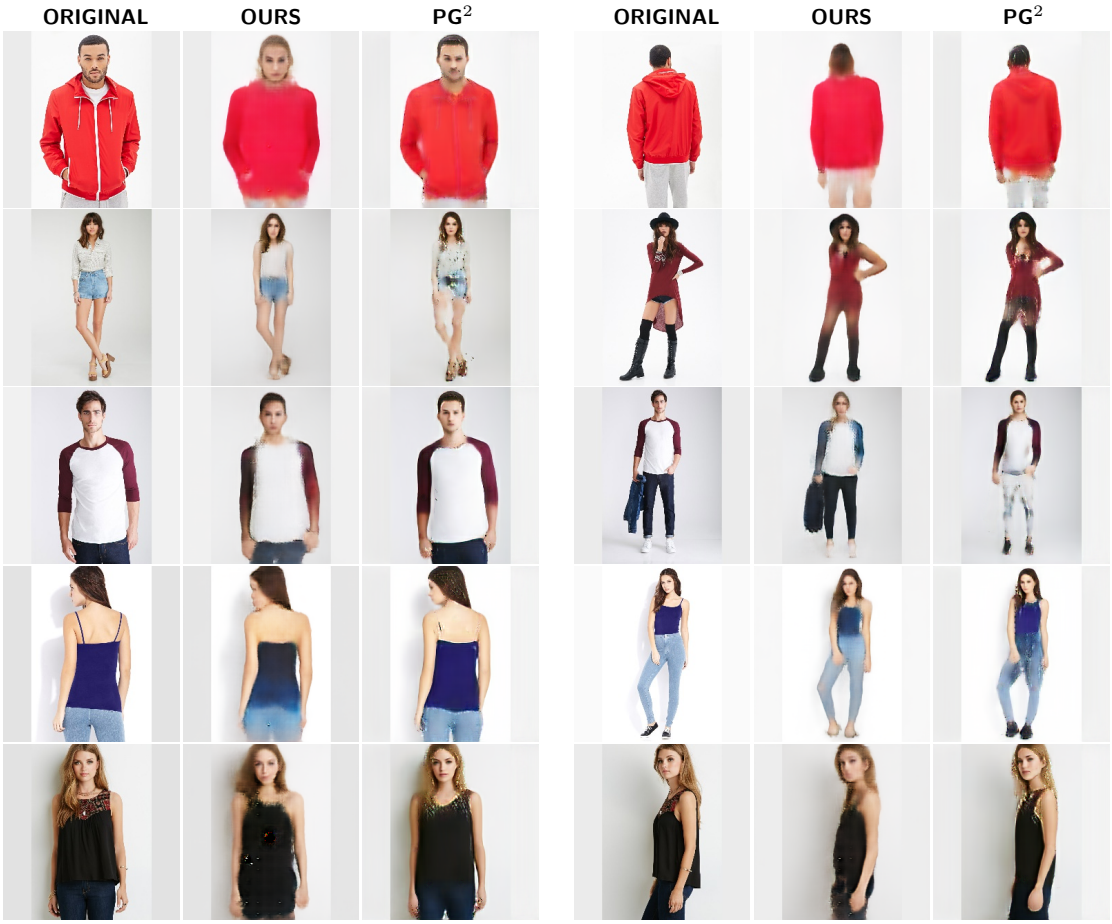


Figure 4.21: DeepFashion Reconstructions. In each trio of images, we have, respectively: original image, Conditional-DGPose, and PG^2 (Ma et al., 2017b) reconstructions. All images have 256×256 pixels. Although tackling a more complex task than Ma et al. (2017b), our results are still comparable. Best viewed if zoomed in digital version.

4.6 Conclusions

In this chapter, we have introduced the Conditional-DGPose, a conditional-VAEGAN deep generative model of people in natural images. Our model is conditioned on 2D human pose, allowing the disentangled representation of body posture and other factors of variation in the images. In contrast to other approaches in the literature, we model the problem in the high-dimensional image space. This allows us to generate image samples conditioned on human pose, in opposition to other methods which can only sample in a low-dimensional space (e.g. pose vectors or segmentation masks), relying on image-to-image translations for mappings to the image space. We have evaluated several design options and

performed experiments specifically in the context of human pose. In the adopted benchmarks, by the generation of realistic images, our methodology has shown the capability of learning the underlying factors that jointly contribute to the generation of a human body in RGB images. We have successfully validated our model on the Human3.6M dataset and obtained state-of-the-art results in the ChictopiaPlus benchmark, outperforming the closest related method in the literature, the ClothNet-body architecture (Lassner et al., 2017).

4.A Appendix

Here, we provide implementation details of the Conditional-DGPose architecture in Tabs. 4.5 and 4.6.

RESIDUAL Layer	
Input: *previous_layer_output*	
Layer	**Definition**
1	CONV-(N512, K3, S1, P1), BN, ReLU
2	CONV-(N512, K3, S2, P1), BN
3	SUM(*conv2_output*, *previous_layer_output*)

Table 4.5: Architecture of the residual block employed in the Conditional-DGPose encoder.

Conditional-DGPose Architecture	
Encoder	
Input: *images*(batch_size=64, channels=3, height=64, width=64) *labels*(batch_size=64, channels=24, height=64, width=64);	
Layer	**Definition**
1	CONCAT(*image*, *labels*)
2	CONV-(N64, K7, S2, P1), LeakyReLU(0.01)
3	CONV-(N128, K3, S2, P1), BN, ReLU
4	CONV-(N256, K3, S2, P1), BN, ReLU
5	CONV-(N512, K3, S2, P1), BN, ReLU
6	CONV-(N512, K3, S2, P1), BN, ReLU
7	CONV-(N512, K3, S2, P1), BN, ReLU
8	RESIDUAL-(N512, K3, S1, P1)
9	RESIDUAL-(N512, K3, S1, P1)
10	RESIDUAL-(N512, K3, S1, P1)
11	RESIDUAL-(N512, K3, S1, P1), SIGMOID
μ	FC-(N100)
σ	FC-(N100)
Prior	
Input: *labels*(batch_size=64, channels=24, height=64, width=64)	
Layer	**Definition**
1	CONV-(N128, K4, S2, P1), LeakyReLU(0.2)
2	CONV-(N256, K4, S2, P1), BN, LeakyReLU(0.2)
3	CONV-(N512, K4, S2, P1), BN, LeakyReLU(0.2)
4	CONV-(N1024, K4, S2, P1), BN, LeakyReLU(0.2)
5	CONV-(N100, K4, S1, P0), SIGMOID
μ_{prior}	FC-(N100)
σ_{prior}	FC-(N100)

Decoder	
Input:	*sample*(batch_size=64, channels=100);
Layer	**Definition**
1	RESHAPE(batch_size=64, channels=100, height=1, width=1)
2	DECONV-(N512, K4, S1, P0), BN, LeakyReLU(0.2)
3	DECONV-(N256, K4, S2, P1), BN, LeakyReLU(0.2)
4	DECONV-(N128, K4, S2, P1), BN, LeakyReLU(0.2)
5	DECONV-(N64, K4, S2, P1), BN, LeakyReLU(0.2)
6	DECONV-(N128, K4, S2, P1), BN, LeakyReLU(0.2)
7	CONCAT(*deconv6_output*, *labels*)
8	CONV-(N512, K5, S1, P2), BN, LeakyReLU(0.2)
9	CONV-(N256, K5, S1, P2), BN, LeakyReLU(0.2)
10	CONV-(N128, K5, S1, P2), BN, LeakyReLU(0.2)
11	CONV-(N128, K5, S1, P2), BN, LeakyReLU(0.2)
$G(\mathbf{y}, \mathbf{z})$	CONV-(N3, K5, S1, P2), TANH
Discriminator	
Input:	*decoder_output*(batch_size=64, channels=3, height=64, width=64); *images*(batch_size=64, channels=3, height=64, width=64)
Layer	**Definition**
1	CONV-(N64, K4, S2, P1), LeakyReLU(0.2)
2	CONV-(N128, K4, S2, P1), BN, LeakyReLU(0.2)
3	CONV-(N256, K4, S2, P1), BN, LeakyReLU(0.2)
4	CONV-(N512, K4, S2, P1), BN, LeakyReLU(0.2)
5	CONV-(N1, K4, S1, P0), SIGMOID

Table 4.6: Conditional-DGPose architecture for 64×64 input images. We use the following abbreviations: N for the number of kernels/neurons, K for kernel size, S for stride and P for zero padding. Concerning the layers, CONCAT means concatenation layer, CONV means convolutional layer, BN means batch normalisation layer with running average coefficient $\beta = 0.9$ and learnable affine transformation, DECONV means transpose convolutional layer, FC means fully connected layer, SUM corresponds to element-wise sum layer, and RESIDUAL denotes a residual block, detailed at Tab. 4.5. The additional layers can be clearly understood. Finally, particular parameters for specific layers are defined between parenthesis after the layers' names.

5

A Semi-supervised Deep Generative Model for Human Body Analysis

Deep generative modelling for human body analysis is an emerging problem with many interesting applications. However, the latent space learned by such approaches is typically not interpretable, resulting in less flexibility. In this chapter, we present a deep generative model for human body analysis where the body pose and the visual appearance are disentangled. Such a disentanglement allows independent manipulation of pose and appearance, and hence enables applications such as pose-transfer without explicitly training for such a task. Our Semi-DGPose is a structured semi-supervised approach which allows for pose estimation to be performed by the model itself and relaxes the need for labelled data. Therefore, the Semi-DGPose aims for the joint *understanding* and *generation* of people in natural images, since it is not only capable of mapping images to interpretable latent representations, but it is also capable of mapping these representations back to the image-space. We compare our model with relevant baselines demonstrating its merits on the Human3.6M and DeepFashion benchmarks.

5.1 Introduction

Human body analysis has been a long-standing goal in computer vision, with many applications in human-machine interaction, health-care, shopping, sports, entertainment and gaming (Achilles et al., 2016; Moeslund et al., 2011; Seemann et al., 2004; Shotton et al., 2011; Marcard et al., 2017). Popular approaches to this problem have focused on supervised learning of discriminative models (Bulat et al., 2016; Cao et al., 2017; Chu et al., 2017; Wei et al., 2016), which map visual inputs (images or videos) to suitable abstract representations (e.g., human body pose). While these approaches do exceptionally well on their prescribed task, as evidenced by their performance on pose estimation benchmarks (Andriluka et al., 2014; Ionescu et al., 2014; Johnson et al., 2010), they fall short due to: a) reliance on fully-labelled data, and b) the inability to generate novel data from the abstractions.

The former is a fairly onerous shortcoming, particularly when one is dealing with real-world visual data, as it requires a substantial amount of human time and effort to annotate. Thus, being able to relax the reliance on labelled data is a highly desirable goal. The latter states a rather significant limitation, the incapacity to manipulate abstractions directly with the aim of generating new visual data. For instance, changes in the pose of an arm cannot be used for the generation of images or videos in which that arm is correspondingly displaced.

Generative models, in contrast to discriminative ones, enable *analysis-by-synthesis* of the human body, where ideally one could generate images of humans in diverse combinations of body poses and appearances, i.e. clothing, skin colours, hair styles, and scenarios. This has many potential applications. For instance, it can be used for performance capture and reenactment of RGB videos, as already showcased for faces (Thies et al., 2016), and still incipient for human bodies (Balakrishnan et al., 2018; Chan et al., 2018). It can also be used to generate images in user specified poses, to enhance and augment datasets with minimal annotation effort.

Recently, such approaches have been commonly formulated as deep generative models (DGMs) (Goodfellow et al., 2014; Kingma et al., 2014b; Rezende et al., 2014) – an extension of standard generative models that incorporate neural networks

as flexible function approximators. These models are particularly effective in complex perceptual domains such as computer vision (Kulkarni et al., 2015), language (Massiceti et al., 2018), and robotics (Wang et al., 2017), effectively delegating bottom-up feature learning to neural networks, while simultaneously incorporating top-down probabilistic semantics into the model. They solve both the deficiencies of discriminative methods discussed above by a) employing unsupervised learning, thereby removing the need for labels, and b) embracing a fully generative modelling.

However, DGMs introduce a new problem – the learnt abstractions, or latent variables, are not human-interpretable. This lack of interpretability is a by-product of the unsupervised learning of representations from data. The learnt latent variables, usually represented as a smooth high-dimensional manifold, do not have a consistent semantic meaning as different sub-spaces in this manifold can encode arbitrary variations in the data. This is particularly unsuitable for our purposes as we would like to view and manipulate the latent variables, e.g., the body pose.

In order to ameliorate the aforementioned issue, while still eschewing reliance on fully-labelled data, we rely on a structured semi-supervised variational autoencoder (VAE) framework (Kingma et al., 2014c; Siddharth et al., 2017). Here, the model structure is assumed to be partially specified, with consistent semantics imposed on some interpretable subset of the latent variables (e.g., pose), and the rest is left to be non-interpretable, although referred by us here as appearance. Weak (semi) supervision acts as a means to constrain the pose latent variables to actually encode the pose. This gives us the full complement of desirable features, allowing a) semi-supervised learning, relaxing the need for labelled data, b) generative modelling through stochastic computation graphs (Schulman et al., 2015), and c) a interpretable subset of latent variables defined through the model structure.

Thus, in this chapter, we present a structured semi-supervised VAEGAN architecture, the Semi-DGPose, in which we further extend structured semi-supervised models (Kingma et al., 2014c; Siddharth et al., 2017) with a discriminator-based loss function from generative adversarial networks (GANs) (Goodfellow et al., 2014; Larsen et al., 2016), formulating it as a principled and unified probabilistic

framework. To our knowledge, it is the first structured semi-supervised deep generative model of people in natural images, directly learned in the image space. In contrast to our Conditional-DGPose (Chapter 4) and previous work (Lassner et al., 2017; Ma et al., 2017b; Ma et al., 2017a; Siarohin et al., 2018; Walker et al., 2017), it directly enables: i) *semi-supervised pose estimation*; and ii) *indirect pose-transfer* without explicit training for such a task. Both of which are tested and verified by experimental evidence.

In the following sections, we present theoretical and technical details, evaluation and discussion which also enable us to shed light on differences and similarities between conditional-VAEGANs, such as our Conditional-DGPose, and structured semi-supervised VAEGANs regarding their theoretical and practical aspects. In summary, our main contributions are:

i) a comprehensive framework for the joint *understanding* and *generation* of people in natural images, not only capable of mapping images to interpretable latent representations but also capable of mapping these representations back to the image-space;

ii) a real-world application of structured deep generative models of natural images, disentangling pose from the appearance in the analysis of the human body;

iii) a thorough quantitative and qualitative evaluation of the capabilities of our model; and

iv) a demonstration of its main utilities by performing semi-supervised pose estimation, pose-transfer and pose manipulation.

5.2 Related Work

To our knowledge, the Semi-DGPose approach contrasts with other methods in the literature since it aims for joint *understanding* and *generation* of people directly in natural images. Particularly, we have not identified any method that gathers the capabilities of pose estimation, image generation and semi-supervised learning.

The novelty in the Semi-DGPose largely relies on how the body pose is handled in our method and how it differs from related works. Distinctively from visual

appearance, which is normally let to be non-interpretable and unsupervised, as it is also the case in our method, the body posture representation may be modelled and learned in more diverse ways.

Generally, in *classical DGMs*, such as standard VAEs and GANs, pose representation is non-interpretable and unsupervised, entangled with the visual appearance in the latent space. This is similarly employed by some *image-to-image translation networks*, however, in contrast to the relatively low-dimensional manifolds learned by the DGMs, in the latter case high-dimensional abstractions are learned and used strictly for direct mapping from and to the image space. On the other hand, *conditional DGMs* usually define part of the abstract data representation, i.e. body pose, to be an interpretable and observable random variable, while the rest of the representation (visual appearance) is kept non-interpretable and latent, still subjected to unsupervised learning. Finally, in *structured DGMs* approaches, as the Semi-DGPose, the latent space can be simultaneously composed by interpretable and non-interpretable random variables. In the former case, the variables may be fully or semi-supervised, while in the latter group they are still maintained unsupervised. Following, we describe related literature gathering the methods according to their adopted type of approach.

Image-to-image networks. Ma et al. (2017b) introduce the Pose Guided Person Generation Network (PG^2), a two-stage image-to-image translation model which is trained on pairs of images of the same person in different poses, scales and points of view. The difficulty of generating poses and detailed appearance simultaneously, in an end-to-end fashion, is admitted by the authors. Their model, which is conditioned on images rather than pose, does not allow sampling, thus in its essence, it is not a generative model, which is again in contrast to our single-stage approaches. In a second approach, Ma et al. (2017a) propose a GAN-based model for learning image embeddings of foreground, background and pose variables encoded as interpretable variables. The method is still limited to training and testing with cross-pose/scale pairs for pose-transfer; however, it allows sampling, differently from the PG^2. In

contrast to our Semi-DGPose model, this method is not capable of performing either pose estimation or semi-supervised learning, relying on off-the-shelf pose estimators to perform pose-transfer. Recently, Esser et al. (2018) present a conditional image-to-image translation network based on the U-Net (Ronneberger et al., 2015). The model is conditioned on an appearance encoding obtained using a VAE architecture. The model is more versatile than the ones by Ma et al. (2017b) and Ma et al. (2017a), although still not capable of producing either an interpretable encoding of pose (pose estimation) or performing semi-supervised learning. Similarly, Balakrishnan et al. (2018) also propose an U-Net-based approach (Ronneberger et al., 2015). In this case, the authors make use of three U-Nets which tackle foreground segmentation and synthesis, as well as background synthesis. The model is trained with video sequences of the same person performing a limited set of activities. Therefore it is limited to translating images of the same person to different poses. Other very recent approaches by Chan et al. (2018) and Neverova et al. (2018) have to be explicitly trained for pose transfer, i.e. using images pairs, and do not have the capability of predicting pose. This is in sharp contrast to our Semi-DGPose approach, in which we learn pose *estimation*, while pose transfer is achieved as a by-product. In the method by Trumble et al. (2018), pose is estimated from multiple views, although it does not allow semi-supervised learning.

Classical DGMs. Lassner et al. (2017) have proposed the ClothNet-full model, in which a VAE model is used to learn a latent representation of segmentation masks of people in given poses. The reconstructed masks are mapped back to the image space by an image-to-image translation module based on the approach by Isola et al. (2017). In contrast, we learn our generative models directly on the raw image data without the need for body parts segmentation. Moreover, the pose is interpretable in our method. Siarohin et al. (2018) propose a GAN model with skip connection in the generator and a discriminator conditioned on pose. Similarly to Ma et al. (2017b), the model is restricted to pose transfer on pairs of images of the same person. The body pose is always given to the model and non-interpretable in the learned latent

encoding. Apart from this, Walker et al. (2017) proposed a hybrid architecture, associating a VAE and a GAN for forecasting future poses in a video. Here, a low-dimensional pose representation is learned using a VAE, and once the future poses are predicted, they are mapped to images using a GAN generator. Considering GAN based generative models, Tulyakov et al. (2018) present a GAN network that learns motion and content in two separate latent spaces in an unsupervised manner. However, it does not allow explicit manipulation over the human pose.

Conditional DGMs. Lassner et al. (2017) present a second model, named ClothNet-Body, which is now a CVAE conditioned on human pose. This model is closely related to our Conditional-DGPose (Chapter 4); however, it also uses low-dimensional segmentation masks and an auxiliary "image-to-image" transfer network (Isola et al., 2017) to generate realistic images. Pumarola et al. (2018) propose an unsupervised image synthesis based on a conditional GAN method; however, it is also not capable of performing pose prediction.

Finally, although there are methods in the literature closely related to our Conditional-DGPose model, to our knowledge, no other method gathers the capabilities of our Semi-DGPose approach. Particularly regarding pose estimation, indirect pose transfer and semi-supervised learning, aiming for joint *understanding* and *generation* of people in natural images. Following Larsen et al. (2016), we use a discriminator in our training to improve the quality of the generated images. However, in contrast to their work, the latent space of our approach is interpretable, which enables us to sample different poses and appearance.

5.3 Preliminaries

Deep generative models (DGMs) come in two broad flavours – Variational Autoencoders (VAEs) (Kingma et al., 2014b; Rezende et al., 2014), and Generative Adversarial Networks (GANs) (Goodfellow et al., 2014). In both cases, the goal is to learn a generative model $p_\theta(\mathbf{x}, \mathbf{z})$ over data $\mathbf{x}$ and latent variables $\mathbf{z}$, with

parameters θ. Typically the model parameters θ are represented in the form of a neural network.

VAEs express an objective to learn the parameters θ that maximise the marginal likelihood (or evidence) of the model denoted as $p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z})dz$. They introduce a conditional probability density $q_\phi(\mathbf{z}|\mathbf{x})$ as an approximation to the unknown and intractable model posterior $p_\theta(\mathbf{z}|\mathbf{x})$, employing the variational principle in order to optimise a surrogate objective $\mathcal{L}(\phi, \theta; \mathbf{x})$, called the evidence lower bound (ELBO), as defined in Eq. 4.1 (Sec. 4.3). The conditional density $q_\phi(\mathbf{z}|\mathbf{x})$ is called the recognition or inference distribution, with parameters ϕ also represented in the form of a neural network.

As already mentioned in Sec. 4.3, learning a surrogate to the likelihood function $p_\theta(\mathbf{x}|\mathbf{z})$ results in a much higher quality of generated data, particularly in the visual domain. GANs achieve that by viewing the generative model $p_\theta(\mathbf{x}, \mathbf{z})$ as a mapping $G : \mathbf{z} \mapsto \mathbf{x}$ in a two-player minimax game. In this setup the "discriminator" $D : \mathbf{x} \mapsto \{0, 1\}$ attempts to correctly identify if a data point $\mathbf{x}$ came from the generative model $p_\theta(\mathbf{x}, \mathbf{z})$ or from the true data distribution $p(\mathbf{x})$. Thus, this family of DGMs, the VAEGANs (Larsen et al., 2016), bring together these two different approaches into a single objective that combines both the VAE and GAN objectives directly as

$$\mathcal{L} = \mathcal{L}_{\text{VAE}} + \mathcal{L}_{\text{GAN}}. \tag{5.1}$$

On the other hand, in the context of structured semi-supervised learning, one can factor the latent variables into unstructured or non-interpretable variables $\mathbf{z}$ and structured or interpretable variables $\mathbf{y}$ without loss of generality (Kingma et al., 2014c; Siddharth et al., 2017). For learning in this framework, the objective can be expressed as the combination of supervised and unsupervised objectives. Let $\mathcal{D}_u$ and $\mathcal{D}_s$ denote the unlabelled and labelled subset of the dataset $\mathcal{D}$, and let the joint recognition network factorise as $q_\phi(\mathbf{y}, \mathbf{z}|\mathbf{x}) = q_\phi(\mathbf{y}|\mathbf{x})q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{y})$. Then, the combined

objective summed over the entire dataset corresponds to

$$\mathcal{L}_{\text{SS}}(\theta, \phi; \mathcal{D}) = \sum_{\mathbf{x}_u \in \mathcal{D}_u} \mathcal{L}_u(\theta, \phi; \mathbf{x}_u) + \gamma \sum_{(\mathbf{x}_s, \mathbf{y}_s) \in \mathcal{D}_s} \mathcal{L}_s(\theta, \phi; \mathbf{x}_s, \mathbf{y}_s), \tag{5.2}$$

where $\mathcal{L}_u$ and $\mathcal{L}_s$ are defined respectively as

$$\mathcal{L}_u(\theta, \phi; \mathbf{x}_u) = \mathcal{L}_{\text{VAE}}(\theta, \phi; \mathbf{x}_u), \tag{5.3}$$

$$\mathcal{L}_s(\theta, \phi; \mathbf{x}_s, \mathbf{y}_s) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_s, \mathbf{y}_s)} \left[\log \frac{p_\theta(\mathbf{x}_s, \mathbf{z}|\mathbf{y}_s)}{q_\phi(\mathbf{z}|\mathbf{x}_s, \mathbf{y}_s)} \right] + \alpha \log q_\phi(\mathbf{y}_s|\mathbf{x}_s). \tag{5.4}$$

Here, the hyper-parameter γ (Eq. 5.2) controls the relative weight between the supervised and unsupervised dataset sizes, and α (Eq. 5.4) controls the relative weight between generative and discriminative learning.

5.4 Our Approach

Following the preliminaries (Sec. 5.3), we use the VAEGAN framework as the basis for our generative models (Larsen et al., 2016). Note that, in incorporating semi-supervised learning, the semi-supervised VAEGAN includes two distinct tasks. First, it involves learning a recognition network that can estimate pose $\mathbf{y}$ (interpretable) and *appearance* $\mathbf{z}$ (non-interpretable) for any given RGB image $\mathbf{x}$. Second, it consists of learning a generative network that combines a given pose with an appearance to generate visual data (RGB image) corresponding to those variables.

From discriminative modelling, we know that the first task, i.e. predicting pose, is eminently plausible up to learning an appearance model. However, learning the full generative model is something that can be fraught with difficulties. For one, pose and appearance can exhibit a large degree of information imbalance – pose can be distilled into a set of (x, y) coordinates, whereas appearance can encode a vast swathe of information (i.e. texture, colour, and shapes) about the given input.

Given a generative model that takes both appearance $\mathbf{z}$ and pose $\mathbf{y}$ as inputs to produce an RGB image $\mathbf{x}$, a reasonable first step could be to just evaluate the

performance of a conditional generative model, where the conditioning variable is taken to be the interpretable pose $\mathbf{y}$. This is, in fact, a conditional-VAEGAN model which turns out to be our Conditional-DGPose setup (Chapter 4). Its lower bound is given by Eq. 4.2, and its final objective function is defined as by Eq. 4.4 (Sec. 4.3), differently from the standard VAEGAN objective (Eq. 5.1). In the Conditional-DGPose, all data is "labelled" with pose, but the goals were: i) primarily, to verify qualitatively if a low-dimensional conditioning variable would affect the conditional generative model; ii) secondly, to evaluate the accuracy of the reconstructed images quantitatively w.r.t. the human body poses and the image quality.

Once verified through experiments that the conditional approach works, we could then proceed towards our structured semi-supervised VAEGAN, referred to as Semi-DGPose. The main difference from the previous setup is that the encoding distribution is no longer conditioned on the pose, but instead predicts it as per Eq. 5.2–5.4. In contrast to the standard VAEGAN objective (Eq. 5.1), the structured semi-supervised VAEGAN final objective function is given by,

$$\mathcal{L} = \mathcal{L}_{\text{SS}} + \mathcal{L}_{\text{GAN}}. \tag{5.5}$$

We describe the details and implementations of our Semi-DGPose model in the rest of this section.

5.4.1 Semi-DGPose

We have tested several variations of deep CNN architectures for implementing our model, culminating in our best performing one, which is described here. All its modules are deep CNNs, and full implementation definitions are given in Tabs. 5.5 and 5.4, Sec. 5.A (chapter appendix), and properly referred to in the text. An overview of our model is shown in Fig. 5.1. Due to the generality of generative models, the architectures may be employed in different ways according to the aimed tasks. Thus, we describe separately training and test phases, dividing the latter into *reconstruction*, *pose-transfer*, *sampling* and *pose-estimation*.

Our structured semi-supervised VAEGAN model learns the parameters of three deep CNN networks simultaneously: i) a recognition network (Encoder), which estimates appearance $\mathbf{z}$ and pose $\mathbf{y}$ from a given RGB image $\mathbf{x}$; ii) a generative network (Decoder), which combines appearance $\mathbf{z}$ and pose $\mathbf{y}$, to generate corresponding RGB images $G(\mathbf{y}, \mathbf{z})$; and iii) a Discriminator network, which differentiates between real images $\mathbf{x}$ and generated images $G(\mathbf{y}, \mathbf{z})$. Learning is pursued by the minimisation of the loss function $\mathcal{L} = \mathcal{L}_{\text{SS}} + \mathcal{L}_{\text{GAN}}$ (Eq. 5.5, Sec. 5.4), composed by the structured semi-supervised VAE evidence lower bound (ELBO) $\mathcal{L}_{\text{SS}}$ and by the GAN cross-entropy discriminator loss $\mathcal{L}_{\text{GAN}}$. A fourth module, called Mapper, is introduced by us to overcome a peculiarity caused by the inclusion of pose in the latent space. Such a module, trained separately, is described next.

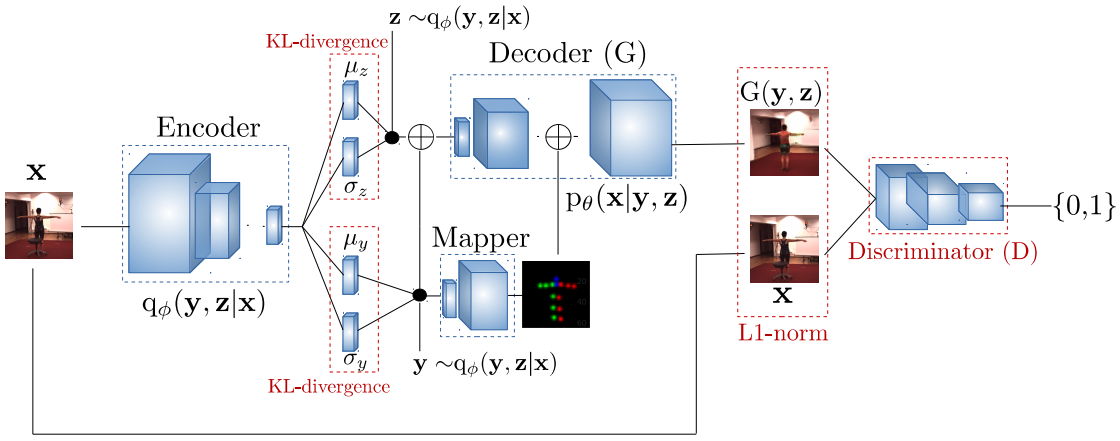


Figure 5.1: Semi-DGPose architecture. At the training, the Encoder receives $\mathbf{x}$ as input and learns the posterior distribution $q_\phi(\mathbf{y}, \mathbf{z}|\mathbf{x})$. In the *unsupervised* routine, samples of appearance $\mathbf{z}$ and pose $\mathbf{y}$ are obtained using the reparameterisation trick (Kingma et al., 2014b). These samples are passed to the Decoder, which generates a reconstructed image $G(\mathbf{y}, \mathbf{z})$). The unsupervised loss function is composed by the following terms, highlighted in red: the L1-norm $L1(\mathbf{x}, G(\mathbf{y}, \mathbf{z}))$ between the original and the reconstructed images; the KL-divergence losses between the posterior distribution $q_\phi(\mathbf{y}, \mathbf{z}|\mathbf{x})$ and the weak priors $p(\mathbf{y})$ and $p(\mathbf{z})$, which work as regularisers (see Eq. 5.3); and the cross-entropy Discriminator loss (Eq. 4.3, Sec. 5.3). In the *supervised* routine (not shown above for simplicity), the only difference is that a regression loss between the estimated pose and the pose ground-truth label substitutes the KL-divergence over the pose posterior distribution (see Eq. 5.4). In both, supervised and unsupervised training routines, the low-dimensional pose vector $\mathbf{y}$ is mapped to a heatmap representation by the Mapper module and concatenated to the Decoder. Eq. 5.2 (Sec. 5.4) shows the overall loss function.

Pose Representation and the Mapper Module

We use the same heatmap pose representation detailed in Sec. 4.4.1. However, differently than the Conditional-DGPose (Chapter 4), in which only the heatmap representation is employed, in the Semi-DGPose model, we additionally employ the vector-based form, as a way of maintaining a low-dimensional latent representation of pose, as described following.

Our experiments with the Conditional-DGPose (Chapter 4) showed that heatmaps led to better quality generation results, in contrast to the vector-based representation. On the other hand, a low-dimensional representation is more suitable and desirable as a latent variable, since human pose lies in a low-dimensional manifold embedded in the high-dimensional image space (Elgammal et al., 2004; Goodfellow et al., 2016). To cope with this mismatch, we introduce the Mapper module, which maps 2D pose-vectors to heatmaps. Ground-truth heatmaps are constructed from manually annotated ground-truth 2D joints labels, by means of a simple weak annotation strategy (Sec. 3.3.1, Chapter 3). The Mapper module is then trained to map 2D joints to heatmaps, minimising the L2-norm between predicted and ground-truth heatmaps. This module is trained separately with the same training hyper-parameters used for our full architecture, described later in Sec. 5.5.2. In the training of the full Semi-DGPose architecture, the Mapper module is integrated to it with its weights kept fixed, since the mapping function has been learned already. The Mapper allows us to keep a low-dimensional representation in the latent space, at the same time that a dense high-dimensional "spatial" heatmap representation facilitates the generation of accurate images by the Decoder. As it is fully differentiable, the module allows the gradients to be backpropagated normally from the Decoder to the Encoder, when it is required during the training of the full architecture.

Training

The terms of Eq. 5.2 correspond to two training routines which are alternately employed, according to the presence of ground-truth labels.

In the *unsupervised case*, when no label is available, it is similar to the standard VAE (see Eq. 5.3). Specifically, given the image $\mathbf{x}$, the Encoder estimates the posterior distribution $q_\phi(\mathbf{y}, \mathbf{z}|\mathbf{x})$, where both appearance $\mathbf{z}$ and pose $\mathbf{y}$ are assumed to be independent given the image $\mathbf{x}$. Then, pose $\mathbf{y}$ and appearance $\mathbf{z}$ are sampled from the posterior, using the reparameterisation trick (Kingma et al., 2014b), and passed to the Decoder to generate a reconstructed image. Finally, the unsupervised loss function minimised during training is composed of the L1-norm reconstruction loss $L1(\mathbf{x}, G(\mathbf{y}, \mathbf{z}))$; the KL-divergences, which act as regularisers, between the posterior and the prior distributions, $\mathrm{KL}[q_\phi(\mathbf{y}|\mathbf{x})|p(\mathbf{y})]$ and $\mathrm{KL}[q_\phi(\mathbf{z}|\mathbf{x})|p(\mathbf{z})]$; and the GAN cross-entropy Discriminator loss (Eq. 4.3, Sec. 4.3).

In the *supervised case*, when the pose label is available, the KL-divergence between the posterior pose distribution and the pose prior, $\mathrm{KL}[q_\phi(\mathbf{y}|\mathbf{x})|p(\mathbf{y})]$, is replaced with a regression loss between the estimated pose and the given label (see Eq. 5.4). Now, only the appearance $\mathbf{z}$ is sampled from the posterior distribution and passed to the Decoder, along with the ground-truth pose label. Finally, the supervised loss function minimised during training is composed of the L1-norm reconstruction loss, the KL-divergence over the appearance distribution, the regression loss over the pose vector and the GAN cross-entropy Discriminator loss. In this case, gradients are not backpropagated from the Decoder to the Encoder, through the pose posterior distribution, since pose was not estimated.

In both *unsupervised* and *supervised* cases, the Mapper module, which is trained *offline*, is used to map the 2D pose-vector in the latent space to a dense heatmap representation, as illustrated in Fig. 5.1.

Testing

Due to the inherent versatility of generative models, our architecture may be employed in different ways, according to the intended task. Thus, the testing stage is divided into *reconstruction*, *indirect pose-transfer*, *sampling* and *pose estimation*.

Reconstruction. At test time, only an image $\mathbf{x}$ is given as input, and the reconstructed image $G(\mathbf{y}, \mathbf{z})$ is obtained from the Decoder. In the reconstruction process, *direct manipulation* of the pose representation $\mathbf{y}$ allows image generations with varying body pose and size while the appearance is kept the same.

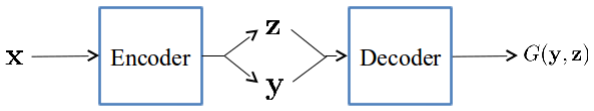


Figure 5.2: Semi-DGPose reconstruction at test time.

Indirect Pose-transfer. Our method allows us to do *indirect* pose-transfer without explicit training for such a task. In this case, i) an image $\mathbf{x}_1$ is first passed through the Encoder network, from which the target pose $\mathbf{y}_1$ is kept. ii) In the second step, another image $\mathbf{x}_2$ is propagated through the Encoder, from which the appearance encoding $\mathbf{z}_2$ is kept. iii) Finally, $\mathbf{z}_2$ and $\mathbf{y}_1$ are jointly propagated through the Decoder, and an image $\mathbf{x}_3$ is reconstructed, containing a person in the pose $\mathbf{y}_1$ estimated from the first image, but with the appearance $\mathbf{z}_2$ defined by the second image. This is a novel application that our approach enables; in contrast to the prior art, our network neither relies on any external pose estimator nor on conditioning labels to perform pose-transfer.

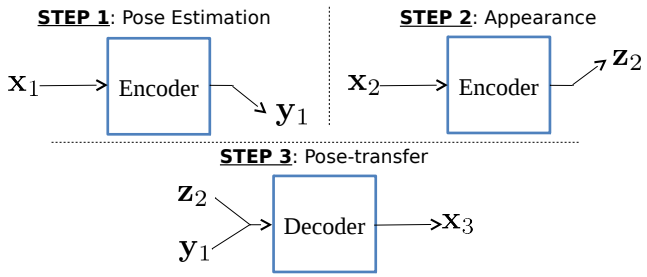


Figure 5.3: Semi-DGPose indirect pose-transfer at test time.

Sampling. When no image is given as input, we can jointly or separately sample pose $\mathbf{y}$ and appearance $\mathbf{z}$ from the posterior distribution. They may be sampled at the same time, or one may be kept fixed while the other distribution is sampled. In all cases, the encodings are passed through the Decoder network to generate a corresponding RGB image.

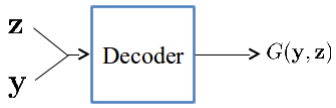


Figure 5.4: Semi-DGPose sampling at test time.

Pose Estimation. One of the main differences between our approach and the prior art is the ability of our model to estimate human-body pose as well. In our model, given an input image $\mathbf{x}$, it is possible to perform pose estimation by regressing to the pose representation vector $\mathbf{y}$. In this case, the appearance encoding $\mathbf{z}$ is disregarded, and the Decoder, Mapper, and Discriminator networks are not used.

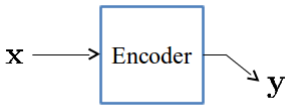


Figure 5.5: Semi-DGPose pose estimation at test time.

5.5 Experiments and Discussion

We have performed several experiments to evaluated our model on the Human3.6M (Ionescu et al., 2014) and on the DeepFashion (Liu et al., 2016b) datasets, described respectively in Secs. 4.5.1 and 4.5.3 of Chapter 4. The Human3.6M is more suitable than both, the ChictopiaPlus (Lassner et al., 2017) and the DeepFashion, for pose estimation evaluations, since the former has joints' annotations obtained by means of an accurate motion capture system. While the two other datasets are augmented with 2D pose labels obtained by means of an *off-the-shelf* pose estimator, consequently resulting in more errors in the

ground-truth annotations. We show quantitative and qualitative results, focusing particularly on the pose estimation and the *indirect* pose-transfer capabilities described later in this section. Our experiments and results show the effectiveness of the Semi-DGPose method on the Human3.6M.

To show the generality of the model, we present additional results on the DeepFashion dataset. We now use our Conditional-DGPose architecture and the image-to-image translation network PG^2 (Ma et al., 2017b) as baselines, despite their relevant differences with the Semi-DGPose. However, to our knowledge, there are no closer related methods in the literature, i.e. that simultaneously pursue the *understanding* and the *generation* of people directly in natural images. Since our Conditional-DGPose method outperforms the ClothNet-body (Lassner et al., 2017) architecture, we do not carry out a direct comparison with the latter.

Following, we present the evaluation metrics, the training hyperparameters, and the quantitative and qualitative results on Human3.6M and DeepFashion, showing the effectiveness and novelty of our Semi-DGPose architecture.

5.5.1 Metrics

We evaluate the performance of the Semi-DGPose w.r.t. three different aspects: i) **Image quality** and ii) **Accuracy of the reconstructed poses**, both already used for the Conditional-DGPose evaluation and described in Sec. 4.5.5, Chapter 4. Additionally, we assess iii) **Accuracy of pose estimation**, obtained by the Semi-DGPose model, using the PCK metric (Yang et al., 2011) with *real* 2D annotated labels as ground-truths.

5.5.2 Training Hyperparameters

The model is trained with mini-batches consisting of 64 images. We used the Adam optimiser (Kingma et al., 2014a) with an initial learning rate set to 10^{-4}. The weight decay regulariser was set to 5×10^{-4}. Network weights were initialised randomly for fully-connected layers and with robust initialisation (He et al., 2015) for convolutional and transposed-convolutional layers. Except when stated differently,

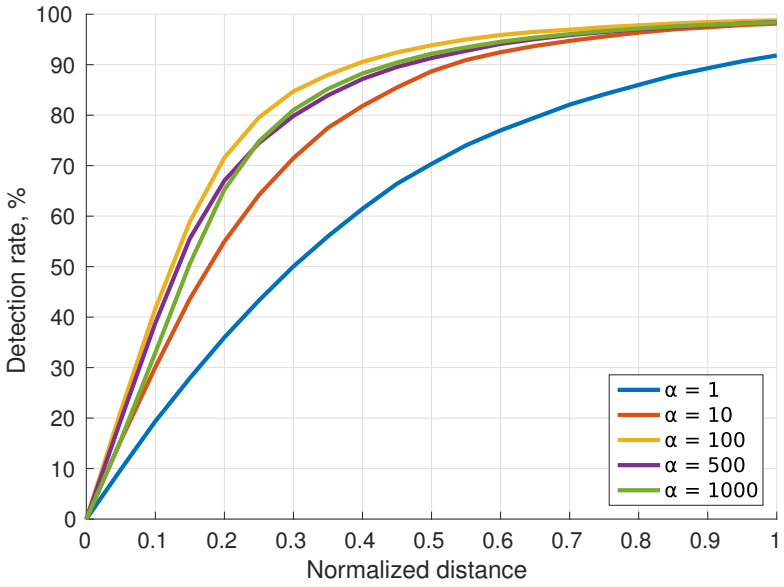
for all images and all models, we used a 64×64 pixel crop, centring the person of interest. We did not use any form of data augmentation or preprocessing except for image normalisation to zero mean and unit variance. Implementation is done in Caffe (Jia et al., 2014) and all experiments ran on an NVIDIA Titan X GPU.

5.5.3 Results on Human3.6M

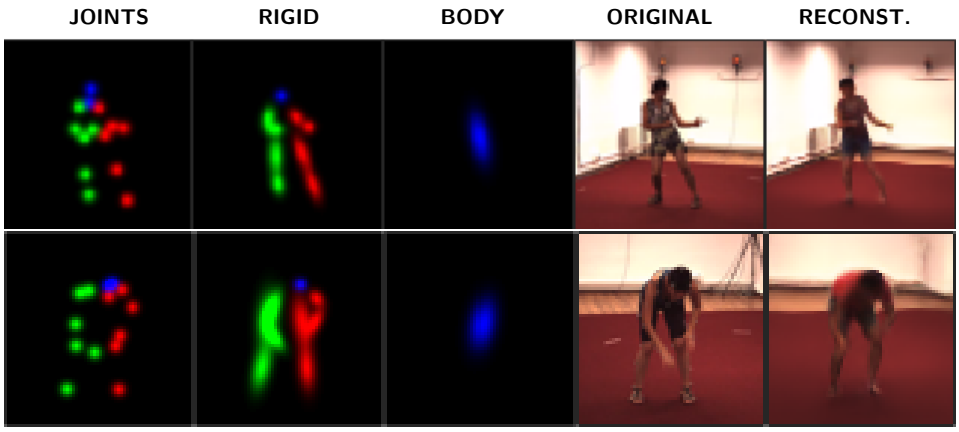
To evaluate the efficacy of our model, we perform a "relative" comparison. In other words, we first train our model with full supervision (i.e. all data points are labelled) to evaluate performance in an ideal case and then we train the model with other setups, using labels only for 75%, 50% and 25% data points. Such an evaluation allows us to decouple the efficacy of the model itself and the semi-supervision to see how the gradual decrease in the level of supervision affects the final performance of the method on the same validation set.

With full supervision, we first cross-validated the hyperparameter α which weights the regression loss (see Eq. 5.4, in Sec. 5.3) and found that $\alpha = 100$ yields the best results, as shown in Fig. 5.6a. Following (Siddharth et al., 2017), we keep $\gamma = 1$ in all experiments (see Eq. 5.2, in Sec. 5.3). In Fig. 5.6b, we show reconstructed images along with the heatmap pose representation, which are realistic and comparable with the ones obtained with the Conditional-DGPose. *Direct manipulation*, when pose representation is changed during the reconstruction process while appearance is kept the same, is illustrated in Fig. 5.7. Still with full supervision, we show the pose estimation accuracy for different samples in Fig. 5.8. The Semi-DGPose achieves 93.85% PCK score, normalised at 0.5, in the fully-supervised setup (see Fig. 5.10). This pose estimation accuracy is on par with the state-of-the-art pose estimators on unconstrained images (Yang et al., 2017). However, since the Human3.6M was captured in a controlled environment, a standard (discriminative) pose estimator can be expected to perform better.

We evaluate it across different levels of supervision, with the PSNR and SSIM metrics (see Sec. 4.5.5) and show results in Tab. 5.1. In Fig. 5.9, we show reconstructed images obtained with such different levels. It allows us to



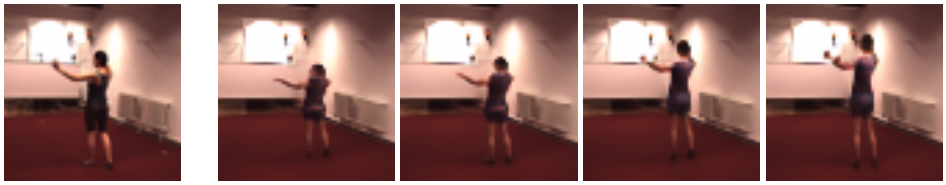
(a)



(b)

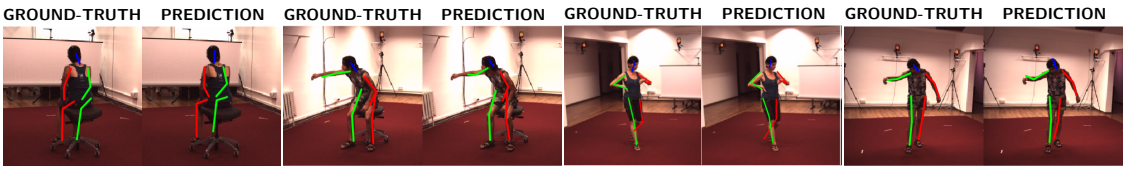
Figure 5.6: (a) PCK scores for the cross-validation adjustment of the regression loss weight α. (b) Qualitative reconstructions with full supervision.

observe how image quality is affected when we gradually reduce the availability of labels. Furthermore, we also evaluated the pose estimation accuracy with semi-supervision. The overall PCK curves corresponding to each percentage of supervision in the training set is shown in Fig. 5.10. Note that, even with only 25% of labels available, our model still obtains 88.35% PCK score, normalised at 0.5, showing the effectiveness of the semi-supervised approach. Qualitative samples are shown in Figure 5.11. Again, aiming to illustrate how the gradual decrease of supervision in the training set affects the quality of pose estimation on the test images.



(a) **(b)** **(c)** **(d)** **(e)**

Figure 5.7: Direct manipulation. Original image (**a**), followed by reconstructions in which the person's height was changed to a percentage of the original, as (**b**) 80%, (**c**) 95%, (**d**) 105%, and (**e**) 120%. The same procedure may be applied to produce different changes in the body size and aspect ratio.



PCK=92.9% PCK=100% PCK=96.4% PCK=100%

Figure 5.8: PCK scores with 100% of supervision, normalised at 0.5. Pose ground-truth (left) and prediction (right) pairs, superimposed on the original images. Each pair correspond to one of the 4 cameras from the Human3.6M dataset. Best viewed if zoomed in digital version.

Concerning *indirect pose-transfer*, as both latent variables corresponding to pose and appearance can be inferred by the model's Encoder (recognition network) at test time, latent variables extracted from different images can be combined in a subsequent step, and employed together as inputs for the Decoder (generative network). The result of that is a generated image combining appearance and body pose, extracted from two different images. The process is done in three phases, as illustrated in Fig. 5.12: i) the latent pose representation $\mathbf{y}_1$ is estimated from the first input image through the Encoder; ii) the latent *appearance* representation $\mathbf{z}_2$ is estimated from a second image, also through the Encoder; iii) $\mathbf{y}_1$ and $\mathbf{z}_2$ are propagated through the Decoder, and a new image is generated, combining body pose and appearance, respectively, from the first and second *encoded* images. We evaluate the effects of semi-supervision over the indirect pose-transfer qualitatively in Fig. 5.13.

Level of supervision	PSNR	SSIM
100%	22.27	0.89
75%	21.49	0.87
50%	21.36	0.86
25%	20.06	0.83

Table 5.1: Quantitative evaluations of Semi-DGPose on Human3.6M. PSNR and SSIM measures for different levels of supervision.

(a) **(b)** **(c)** **(d)** **(e)** **(f)** **(g)**

Figure 5.9: Semi-DGPose reconstructions. (**a**) original images, and (**b**) heatmap pose representation (rigid parts and body suppressed in the illustration for simplicity), followed by reconstructions with different levels of supervision: (**c**) 100%, (**d**) 75%, (**e**) 50%, (**f**) 25%, and (**g**) Conditional-DGPose.

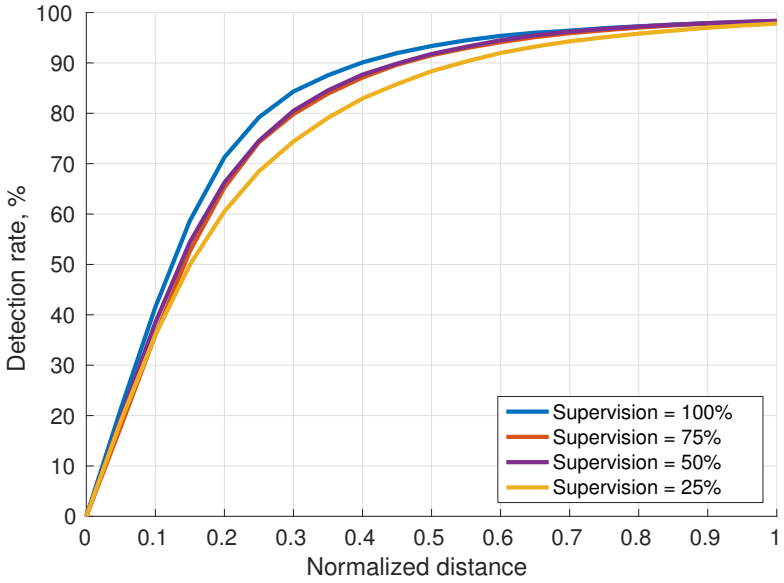
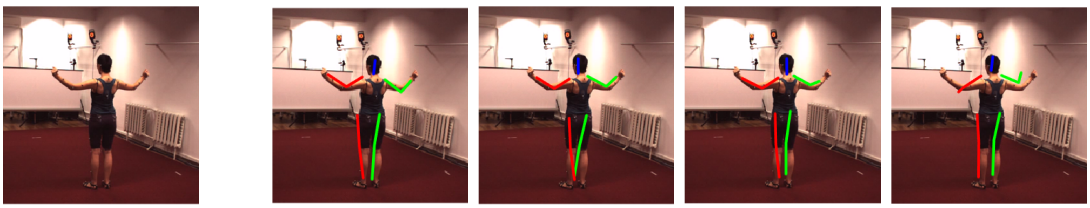


Figure 5.10: Quantitative evaluations of Semi-DGPose on Human3.6M. PCK scores for different levels of supervision. Note that, even with 25% supervision, our Semi-DGPose obtains 88.35% PCK score, normalised at 0.5.



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

Figure 5.11: Pose estimation. Original image (**a**), followed by estimations, over the original image, with: (**b**) 100%, (**c**) 75%, (**d**) 50% and (**e**) 25% of supervision.

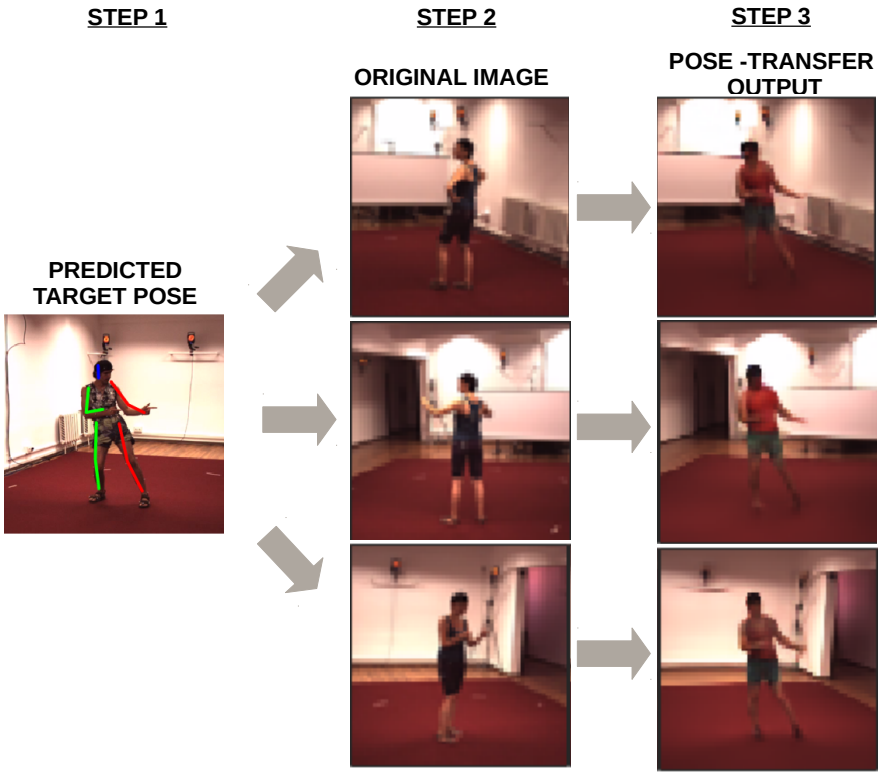


Figure 5.12: *Indirect* pose transfer. Step 1: the latent target pose representation $\mathbf{y}_1$ is estimated (Encoder). **Step 2**: the image from which the latent *appearance* $\mathbf{z}_2$ is estimated (Encoder). **Step 3**: the output image generated as a combination of $\mathbf{y}_1$ and $\mathbf{z}_2$ (Decoder). The people's outfits in the output images are approximated to the ones in the original images, however, restricted by the low diversity of outfits observed in Human3.6M training data. Note that, to highlight the separation of appearance and pose, we chose the image on Step 1 to be from camera 2, while the original images are from cameras, 1, 3 and 4, respectively. As can be seen, the background scene is totally defined by the original images.

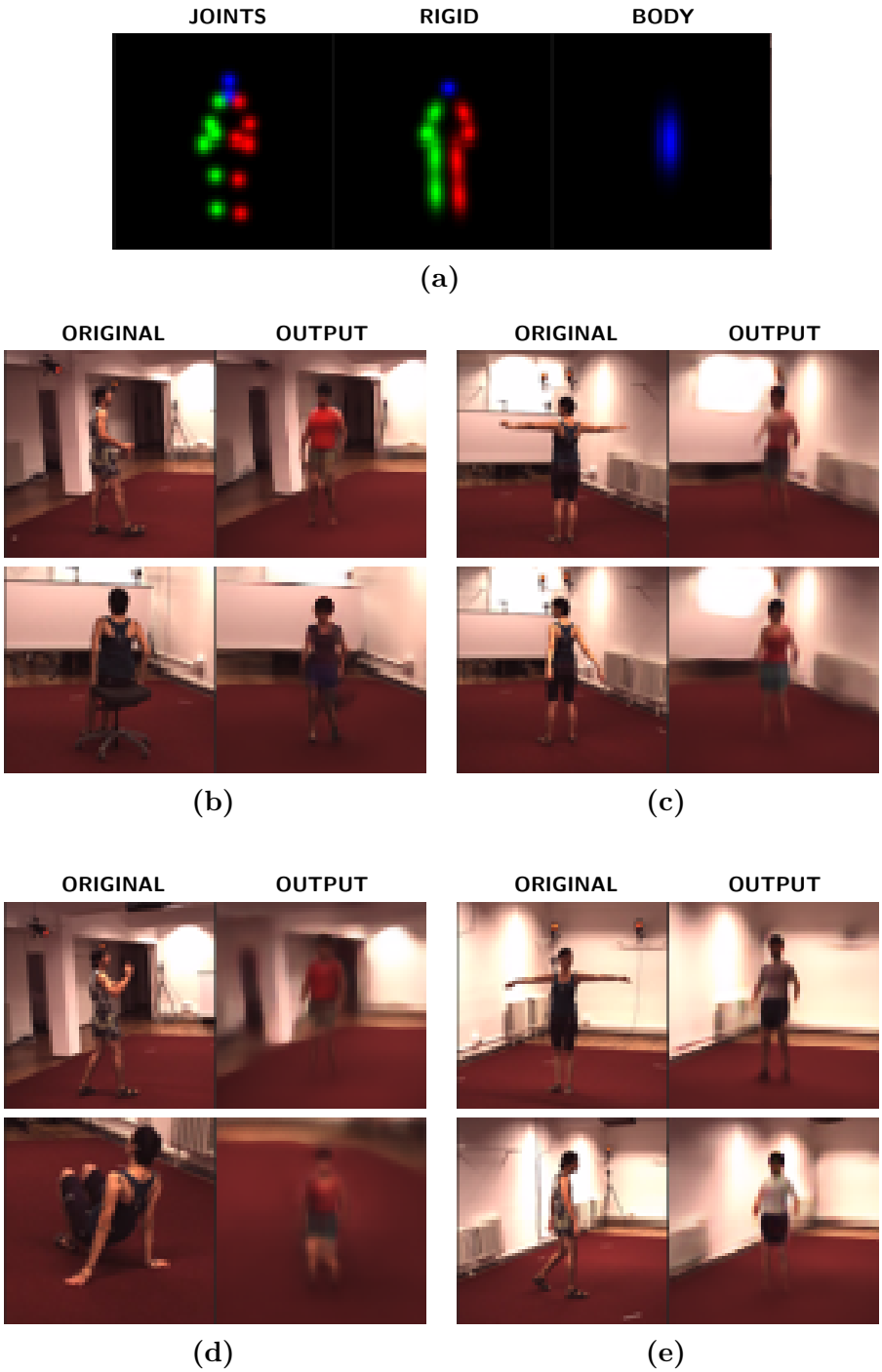


Figure 5.13: Qualitative indirect pose-transfer with different level of supervision. (a) Target-pose after outputted by the Mapper module. Results for the following levels of supervision: (b) 100%, (c) 75%, (d) 50%, and (e) 25%.

5.5.4 Results on DeepFashion

To show the generality of the Semi-DGPose model, we present additional results on the DeepFashion dataset, using our Conditional-DGPose architecture and the image-to-image translation network PG^2 (Ma et al., 2017b) as baselines. The same hyperparameters reported previously were used in training. In Tab. 5.2, we compare the image quality of reconstructions, while in Fig. 5.14, we show the comparison with respect to the quality of pose reconstructions. Although the Semi-DGPose presents less accurate results, it is important to highlight that it is also tackling the pose estimation task, which is not performed by either one of the other two methods, i.e. the Conditional-DGPose and the PG^2. To pursue, simultaneously, the *understanding*, i.e. estimation of pose and appearance in the latent space, and the *generation* of people directly in natural images shows to be indeed a significantly more complex task. Nevertheless, the justification for seeking such a challenging goal, as mentioned before, mainly lie on its important capability of allowing for semi-supervised learning, that is not present in the comparable methods.

	PSNR	SSIM
Semi-DGPose	16.84	0.76
Conditional-DGPose	18.38	0.79
PG^2	**18.96**	**0.83**

Table 5.2: Quantitative evaluations of Semi-DGPose on DeepFashion. PSNR and SSIM measures are comparing the image quality of reconstructions. The Semi-DGPose shows less accurate results; however, in contrast to the other methods, it performs a significantly more complex task, simultaneously executing pose estimation, and also allowing for semi-supervised learning.

In Fig. 5.15, we show comparisons between input and reconstructed images. In some of the samples, we can observe small differences between the original and the reconstructed body postures, mainly regarding the positions of the limbs. This illustrates the higher complexity involved in simultaneously estimating pose and appearance in our latent space. For instance, inaccurate predictions of pose, performed by the Encoder, may have effects into the final reconstructed appearance, and vice-versa, when the latent representations are mapped back to the image

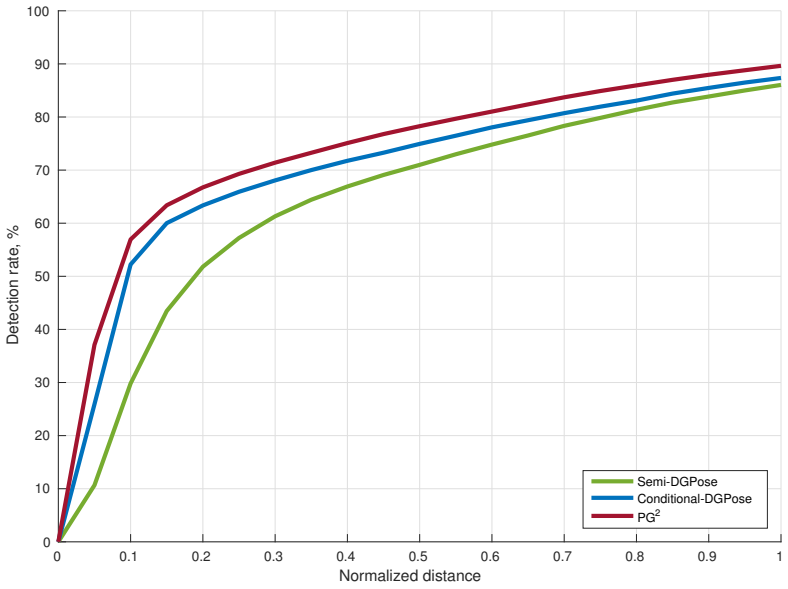


Figure 5.14: Quantitative evaluations of Semi-DGPose on DeepFashion. PCK scores over reconstructed poses. The Semi-DGPose (*green*) shows less accurate results, however, in contrast to the Conditional-DGPose (*blue*) and the PG^2 network (Ma et al., 2017b) the (*red*), it performs a significantly more complex task, simultaneously executing pose estimation, and also allowing for semi-supervised learning. Detection rate represents the percentage of joints corrected relocated in the reconstructions.

space, by the Decoder. Such interdependence does not exist when the pose is a given observable variable, as in the case of the conditional models or image-to-image translation networks.

Finally, we highlight *indirect pose transfer* in the DeepFashion dataset, which is a distinctive capability of the Semi-DGPose, in comparison to related methods. In Fig. 5.16, we compare the indirect pose transfer results, from our single-stage Semi-DGPose structured generative model, with the results from the image-to-image translation baseline, the PG^2 network (Ma et al., 2017b). It is important to notice that our Semi-DGPose model was not explicitly trained for pose-transfer, i.e. it was not trained on pairs of images. On the other hand, the PG^2 architecture is trained on pairs of images of the same person, in different poses, scales or point of views (first two images of each set in Fig. 5.16). Moreover, in the Semi-DGPose, the body pose is estimated by the Decoder network (illustrated in every second image of each set in Fig. 5.16), along with the appearance. While in the PG^2, pose is given as an observable variable to the model. Despite such important competitive

disadvantages, we can observe that the Semi-DGPose produce results comparable to the ones from PG^2. Lastly, it is important to call attention for the capabilities of our Semi-DGPose approach such as, interpretability of the latent space, pose estimation, sampling and semi-supervised learning, which are not jointly present in the PG^2 neither in related work from the literature. These features justify our approach for learning a deep generative model of people in natural images and, to our knowledge, significantly differentiate the Semi-DGPose model from the prior art.

Figure 5.15: Semi-DGPose DeepFashion reconstructions. The only input of the Semi-DGPose is the original image. At test time, as pose is estimated in the latent space, discrepancies between the original and reconstructed poses are more frequently observed, in comparison with the Conditional-DGPose. Best viewed if zoomed in digital version.

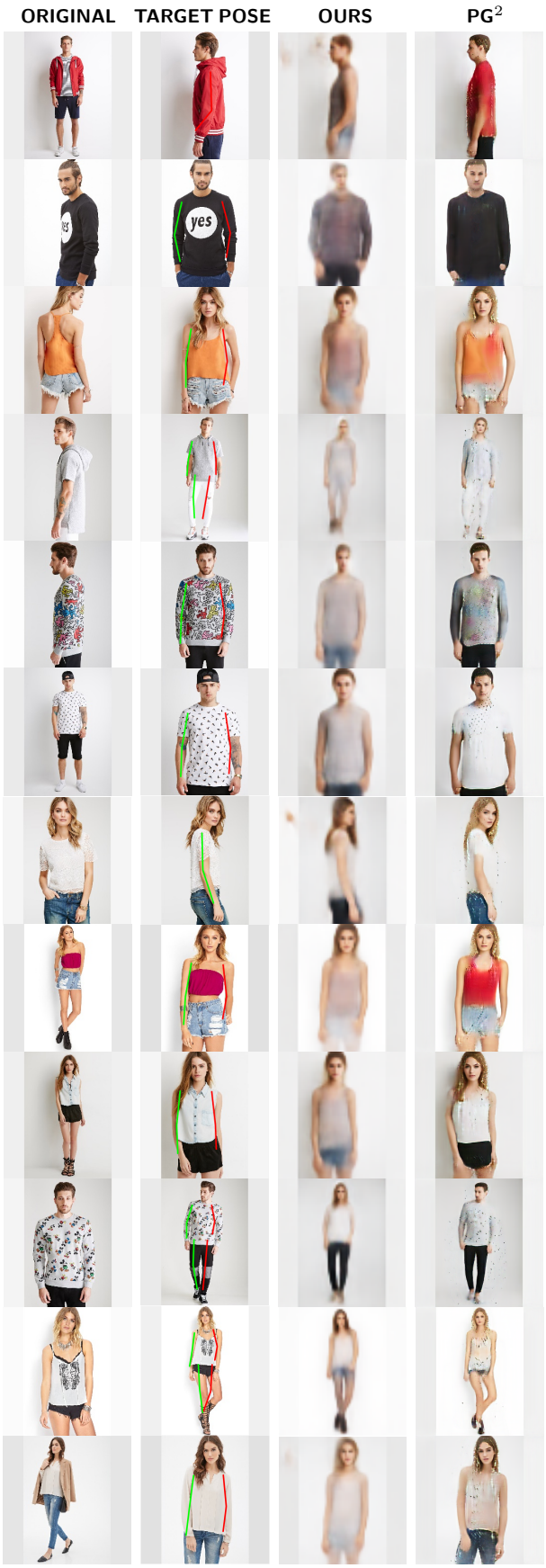


Figure 5.16: Indirect pose transfer in DeepFashion dataset. In each set of images, we have, respectively: the original image, the target image with the superimposed target pose predicted by the Semi-DGPose, the pose transfer output from the Semi-DGPose and the pose transfer output from PG^2 (Ma et al., 2017b). Although tackling a more complex task than Ma et al. (2017b), which includes the prediction of pose, our results are still comparable.

5.6 Conclusions

In this chapter, we have presented a comprehensive deep generative model framework for human pose analysis in natural images. Our model is based on a principled VAEGAN approach and allow the disentanglement of body posture and visual appearance, aiming for the independent manipulation of such factors. With a structured semi-supervised VAEGAN architecture, the Semi-DGPose, we pursued the joint *understanding* and *generation* of people in natural images, not only mapping images to partially interpretable latent representations but also mapping these representations back to the image-space. Importantly, such an approach simultaneously allows for reconstruction, direct manipulation, sampling, pose estimation, indirect pose transfer and semi-supervised learning. These joint capabilities differentiate the Semi-DGPose from other methods in the literature and demonstrate a real-world application of structured deep generative models with the highly important potential of being less dependable of fully-labelled data. We have systematically evaluated our methods on well-known benchmarks, the Human3.6M, and the DeepFashion datasets, comparing our results with the closest related baseline method in the literature (Ma et al., 2017b). Such results and comparisons highlight the novelty and effectiveness of our approaches and its capabilities, despite the great challenge posed by our aimed goal. We believe that we have shown and reinforced the relevance of employing an interpretable and structured latent space, which allows for semi-supervised learning, as well as the importance of tackling the problem with end-to-end architectures.

5.A Appendix

Here, we provide implementation details of the Semi-DGPose architecture in Tabs. 5.4 and 5.5.

RESIDUAL Layer	
Input: *previous_layer_output*	
Layer	**Definition**
1	CONV-(N512, K3, S1, P1), BN, ReLU
2	CONV-(N512, K3, S2, P1), BN
3	SUM(*conv2_output*, *previous_layer_output*)

Table 5.4: Architecture of the residual block employed in the Semi-DGPose encoder.

Semi-DGPose Architecture	
Encoder	
Input: *images*(batch_size=64, channels=3, height=64, width=64)	
Layer	**Definition**
1	CONV-(N64, K7, S2, P1), LeakyReLU(0.01)
2	CONV-(N128, K3, S2, P1), BN, ReLU
3	CONV-(N256, K3, S2, P1), BN, ReLU
4	CONV-(N512, K3, S2, P1), BN, ReLU
5	CONV-(N512, K3, S2, P1), BN, ReLU
6	CONV-(N512, K3, S2, P1), BN, ReLU
7	RESIDUAL-(N512, K3, S1, P1)
8	RESIDUAL-(N512, K3, S1, P1)
9	RESIDUAL-(N512, K3, S1, P1)
10	RESIDUAL-(N512, K3, S1, P1), SIGMOID
μ	FC-(N100)
σ	FC-(N100)
μ_y	FC-(N48)
σ_y	FC-(N48)
Mapper	
Input: *pose_vector*(batch_size=64, channels=48)	
Layer	**Definition**
1	RESHAPE(batch_size=64, channels=48, height=1, width=1)
2	DECONV-(N512, K4, S1, P0), BN, LeakyReLU(0.2)
3	DECONV-(N256, K4, S2, P1), BN, LeakyReLU(0.2)
4	DECONV-(N128, K4, S2, P1), BN, LeakyReLU(0.2)
5	DECONV-(N64, K4, S2, P1), BN, LeakyReLU(0.2)
heatmaps	DECONV-(N24, K4, S2, P1), SIGMOID
Decoder	
Input: *sample*(batch_size=64, channels=100); *heatmaps*(batch_size=64, channels=24, height=64, width=64);	
Layer	**Definition**
1	RESHAPE(batch_size=64, channels=100, height=1, width=1)
2	DECONV-(N512, K4, S1, P0), BN, LeakyReLU(0.2)
3	DECONV-(N256, K4, S2, P1), BN, LeakyReLU(0.2)
4	DECONV-(N128, K4, S2, P1), BN, LeakyReLU(0.2)
5	DECONV-(N64, K4, S2, P1), BN, LeakyReLU(0.2)
6	DECONV-(N128, K4, S2, P1), BN, LeakyReLU(0.2)
7	CONCAT(*deconv6_output*, *heatmaps*)
8	CONV-(N512, K5, S1, P2), BN, LeakyReLU(0.2)
9	CONV-(N256, K5, S1, P2), BN, LeakyReLU(0.2)
10	CONV-(N128, K5, S1, P2), BN, LeakyReLU(0.2)
11	CONV-(N128, K5, S1, P2), BN, LeakyReLU(0.2)
$G(\mathbf{y}, \mathbf{z})$	CONV-(N3, K5, S1, P2), TANH

Discriminator	
Input: *decoder_output*(batch_size=64, channels=3, height=64, width=64); *images*(batch_size=64, channels=3, height=64, width=64)	
Layer	**Definition**
1	CONV-(N64, K4, S2, P1), LeakyReLU(0.2)
2	CONV-(N128, K4, S2, P1), BN, LeakyReLU(0.2)
3	CONV-(N256, K4, S2, P1), BN, LeakyReLU(0.2)
4	CONV-(N512, K4, S2, P1), BN, LeakyReLU(0.2)
5	CONV-(N1, K4, S1, P0), SIGMOID

Table 5.5: Semi-DGPose architecture for 64×64 input images. We use the following abbreviations: N for the number of kernels/neurons, K for kernel size, S for stride and P for zero padding. Concerning the layers, CONCAT means concatenation layer, CONV means convolutional layer, BN means batch normalization layer with running average coefficient $\beta = 0.9$ and learnable affine transformation, DECONV means transpose convolutional layer, FC means fully connected layer, SUM corresponds to element-wise sum layer and RESIDUAL denotes a residual block, detailed at Table 4.5. The additional layers can be clearly understood. Finally, particular parameters for specific layers are defined between parenthesis after the layers' names.

6 Conclusion

In this thesis, we investigate the *understanding* and *generation* of humans in visual data under the light of deep learning methods. In a vast spectrum of topics related to understanding, we have focused on human pose estimation. It is one important and fundamental building block for more semantically complex tasks, such as action recognition and behaviour analysis (Pantic et al., 2006; Poppe, 2010; Jhuang et al., 2013). Regarding generation, here it corresponds to the process of synthesising images of humans taking into account visual appearance attributes, e.g., colour, illumination, surrounding environment; and body pose. Again for generating, the body posture is an essential feature since suitable postural configurations are crucial to grant fidelity to generated "virtual" humans (Akhter et al., 2015). Therefore, the body pose as an attribute of humans in images has centrality in the present work.

In this context, we have explored deep discriminative and generative methods in the form of convolutional neural networks (CNNs), recurrent neural network (RNNs), variational autoencoders (VAEs) and generative adversarial networks (GANs). In **Chapter 3**, we built upon CNNs and RNNs to introduce a novel framework for performing deep discriminative learning and inference in part-based models applied to human pose estimation. To our knowledge, the presented method is the first to allow mean-field approximate inference over a loopy, fully-connected, part-based model using a deep end-to-end architecture. Pairwise spatial relations are

established according to the structure of the conditional random field (CRF) part-based model (e.g., fully-connected), differently from the kind of pairwise relations used in CRF models for image segmentation (Zheng et al., 2015; Krähenbühl et al., 2011). Additionally, we propose a multi-level Gaussian representation for the human body and a novel, inexpensive, and straightforward weakly-supervised methodology for generating corresponding ground-truth annotations. Finally, competitive results on two well-established benchmarks for 2D human pose estimation, the MPII (Andriluka et al., 2014), and the LSP (Johnson et al., 2010), outperforming state-of-the-art methods in particular cases, e.g., unusual poses.

In **Chapter 4**, we propose a conditional deep generative model of people in natural images. The Conditional-DGPose aims to overcome typical limitations of related methods in the literature, such as entangled and non-interpretable factors of variations, dependency on image-to-image translation networks, inaccuracies due to multiple stages of training and testing, and restriction to specific tasks, e.g., pose-transfer. Our conditional-VAEGAN architecture, a principled variational method for approximate Bayesian inference (Sohn, 2015), allows us to pursue the generation of people in images from interpretable and disentangled factors of variation, namely, body pose and visual appearance, by the association with a discriminator module, which takes advantage of the high-quality image generations from GANs (Larsen et al., 2016). To our knowledge, our approach is the first deep generative model learned directly on the high-dimensional image space and capable of generating realistic natural images of people in a unified probabilistic framework, while keeping the body posture and appearance as explicitly separated and interpretable variables. The advantage of that is threefold, as it allows: i) to change the posture of a person in an image, given a conditioning pose (pose-transfer); ii) the sampling from the generative distribution with independent control over pose and visual appearance; and iii) the direct and more accurate control over the visual appearance and pose employing a unified single-stage end-to-end model. We validated our method qualitatively on the Human3.6M dataset (Ionescu et al., 2014). Moreover, our approach achieves state-of-the-art results on the ChictopiaPlus

benchmark, outperforming the closest related work in the literature, the ClothNet-Body network (Lassner et al., 2017). We show that our model generates more realistic and accurate images concerning both body posture and image quality, while it learns the underlying latent factors of pose and appearance variation.

Lastly, in **Chapter 5**, we move towards a unified framework aiming for the simultaneous *understanding* and *generating* of people in images. We present a structured semi-supervised VAEGAN architecture, the Semi-DGPose, in which we further extend structured semi-supervised models (Kingma et al., 2014c; Siddharth et al., 2017) with a discriminator-based loss function from GANs, formulating it as a principled and unified probabilistic framework. This framework is the gathering of a deep discriminative model (i.e., the recognition network), capable of performing pose estimation, and a generative model, capable of generating images from interpretable and disentangled factors of variation. To our knowledge, it is the first structured semi-supervised deep generative model of people in natural images, directly learned in the image space. In summary, contrasting to previous work, our main contributions are: i) a comprehensive framework for jointly *understanding* and *generating* people in natural images, not only capable of mapping images to an interpretable subset of latent variables, but also capable of mapping these variables (representations) back to the image-space; ii) a real-world application of structured deep generative models of natural images, disentangling pose from the visual appearance in the analysis of the human body allowing generative modelling through stochastic computation graphs (Schulman et al., 2015); iii) a thorough quantitative and qualitative evaluation of the capabilities of our models; and iv) a demonstration of its main utilities by performing semi-supervised pose estimation, pose-transfer and pose manipulation on the Human3.6M (Ionescu et al., 2014) and DeepFashion (Liu et al., 2016b) benchmarks.

In Sec. 6.1, we follow examining potential specific improvements to be sought in our proposals. In Sec. 6.2, along with our final remarks, we discuss possible general directions for future research which may follow our current contributions.

6.1 Potential Improvements

In this section, we examine specific characteristics of our methods that could be improved to overcome current limitations. We approach each one of them below, detailing the particular points that could be addressed in further investigations.

A Deep Fully-Connected Part-Based Model for Human Pose Estimation

i) Handling of multimodal spatial distributions – the use of Gaussian pairwise kernels to model the spatial arrangement of body parts allows for an efficient message passing process (Adams et al., 2010); however, it is a limitation of our current approach. The investigation and evaluation of multimodal distributions, such as mixtures of Gaussians, to model the spatial displacement of parts is desirable since it might be beneficial for the performance of the method.

ii) Multiple image features – the use of different images features such as colour, texture, and depth is supported by our method. However, our basic formulation was limited to the use of spatial information regarding the structure of the human body. The addition of extra features to the model might be explored.

iii) Part-based model structure – to showcase the versatility of our method for mean-field inference in part-based models and increase redundancy in the location of body parts we have adopted a fully-connected CRF. However, as it is costly, different part-based models' structures may also be evaluated, e.g., tree-structured, star-structured, aiming for a better balance between cost and performance.

iv) Higher-order potentials – the exploration of higher-order potentials can also be considered. For instance, the joint distributions of triplets of body parts may improve the robustness of the method, although such extensions are usually computationally costly. The evaluation of such high-order potentials might be balanced with the adoption of "lighter" part-based structures instead of the fully-connected one, as mentioned above.

v) Extension to 3D pose – we have applied our model to the 2D pose estimation problem. However, there are no fundamental restrictions for its extension to 3D

pose estimation. The existent ambiguities related to the 2D-3D mapping might turn the explicit use of spatial reasoning about the 3D pose consistency even more relevant. Moreover, as the model naturally supports the use of multiple features, as mentioned above, the 3D locations of body parts may be used jointly with the 2D positions and other features, like colour and depth.

vi) Multi-level pose representation for weak-supervised tasks – the Gaussian heatmaps for joints, rigid parts, and body, from our multi-level representation of humans, has shown to be an effective way of densely cover the area of people's bodies across different datasets. This multi-level representation, derived entirely from the 2D joint's manual annotations, might be explored for tackling tasks such as weakly-supervised segmentation (Kolesnikov et al., 2016) and salience detection (Itti et al., 1998).

vii) Distinct applications – finally, our framework may also be applied in the investigation of other problems, other than human pose estimation, in which part-based models are employed, such as object co-detection (Hayder et al., 2014) and multi-person detection (Liu et al., 2016a).

A Conditional Deep Generative Model of People in Natural Images

i) Conditioning on 3D pose – in the current model, when two body parts are located in a similar position, there is no explicit information about which one appears in the image and which is partially or entirely occluded, turning the correct reconstruction more challenging. Thus, the use of a 3D pose representation for conditioning our VAEGAN model might be helpful in such situations of self-occlusion since it might mitigate ambiguities regarding the correct location of body parts.

ii) Enhanced body models – recently, significant advances were made towards simple and efficient body models. Parametric methods like SMPL (Loper et al., 2015), integrate pose and shape to represent the human body and may easily allow the incorporation of additional features in our model, i.e., segmentation masks. Lassner et al. (2017) use the SMPL model *offline* to produce extra training

data labels. However, we understand that this could be done directly during the end-to-end training of our single-stage Conditional-DGPose model.

iii) Additional interpretable variables – the addition of extra random variables in the model is desirable. Likely, pose and appearance are not sufficient for allowing the application of the model over unconstrained data. Two immediate extensions could be explored: a) the split of the current *pose* and *appearance* variables into more specific ones, e.g., 2D and 3D poses, and background and foreground appearances; b) the addition of extra and even higher-level interpretable variables in the generative model, such as activity.

iv) Generating videos – extending our conditional generative model from still images to videos is an interesting direction to be followed. The use of recurrent neural networks (RNNs) or long short-term memory (LSTMs) for learning generative models in the pose-space is already known (Walker et al., 2017). To our knowledge, learning such models directly in the image space was not yet explored. Moreover, learning video sequences conditioned on higher-level interpretable variables may allow a semantically controlled sampling of videos. For instance, considering activity, one could sample activity-specific videos in which lower-level variables, such as pose and appearance, would be dependent on activity.

A Semi-supervised Deep Generative Model for Human Body Analysis

i) Conditional-DGPose improvements – in theory, all the improvements mentioned earlier, for the Conditional-DGPose model, could be applied to the Semi-DGPose model.

ii) Task-specific recognition networks – the use of specialised recognition networks, e.g., for segmentation, detection, pose estimation, and activity recognition, would be capable of providing more diverse and accurate abstract interpretable representations (latent variables) from images. For that, such representations would need to be correctly modelled as probability distributions and interrelated in the latent space, as mentioned below.

iii) Complex structured relations in the latent space – the exploration of richer generative models with more complex structured relations between the latent variables, modelled through stochastic computation graphs, is desirable. This major advantage of our method may be explored further with refined abstract data representations and interrelations. These improvements would allow explicit control over more interpretable variables and conditional dependencies in the model.

iv) Leverage of semi-supervision – in a model where multiple abstract representations present interrelations and dependencies among each other, it is straightforward to take advantage of semi-supervision over them. It is very common to find works in the literature that extend existing datasets with extra annotations (Lin et al., 2014), (Lassner et al., 2017), (Gong et al., 2017), (Romero et al., 2017). Even when the desirable new labels are closely related to the existing ones, their generation usually requires human annotators to ensure good quality, which makes the method very costly. Semi-supervision could facilitate such a process, possibly requiring fewer data to be "re-annotated". During training, supervision may be used or not, according to the availability of labels.

v) Images in the wild – learning generative models of images in the wild is a difficult task. For that, deep generative methods need to be able to tackle multimodal data distributions to handle more diverse datasets, such as MPII (Andriluka et al., 2014) or COCO (Lin et al., 2014) regarding images of humans. Evaluation of strategies to enhance our models with such capabilities is highly desirable for advancing toward unconstrained images of humans.

6.2 Future Directions and Final Remarks

In the title of this thesis, we mention – "... *Towards* Understanding and Generating ..." – trying to convey our comprehension that still there is a long way to go in these two main directions regarding the analysis of people in images. Despite all the impressive breakthroughs and vertiginous advances led by the use of deep learning techniques in the last few years, it is fair to say that a substantial part of them are still related to low-level tasks in the case of deep discriminative models (*understanding*), or

Figure 6.1: Former United States president, Barack Obama, playing a trick on one of his officials (Obama White House Flickr, 2010).

limited to constrained scenarios concerning deep generative approaches (*generating*). A glimpse that summarises the challenges ahead involving these endeavours may be obtained by the observation of two pictures described following.

The first, shown in Fig. 6.1, relates to *understanding* and depicts the former United States president, Barack Obama, playing a trick on one of his officials. Someone familiarised with the current state-of-the-art deep discriminative methods would expect from such approaches reasonably good results while performing over this image tasks such as object detection, 2D and 3D body pose and shape estimation, activity and face recognition, instance and body-parts segmentation, and captioning. However, could we really say that all this information together allow the "understanding", by these methods, of what is going on in the image? The answer is no. The current automated techniques do not present the common sense knowledge, deeply embedded in our visual and cognitive systems, that allows us to readily say that it is a joke and explain why it is funny. Such a complex awareness, probably a mixture of inherited skills along with the product of a lifelong learning process, is not easily reproducible algorithmically. Andrej Karpathy (2012)[1] presents an interesting informal description of what would be necessary to "know"

[1]The choice for this illustrative picture was inspired by his article.

Figure 6.2: Computer graphics *painting*, "The Portrait of Erica", by the character artist Ian Spriggs (2018).

for correctly comprehend all the subtleties of what is happening on Obama's picture, such as how a scale works and why people worry about their weight. These capabilities are indeed not yet present on the state-of-the-art deep learning methods for understanding humans in images.

Regarding *generating* people in visual data, relevant results were obtained lately with deep generative models. However, the level of fidelity and specialisation obtained by human experts at computer graphics, as can be seen in our second illustrative picture, shown in Fig. 6.2, has not yet been achieved by machine learning methods. For such accuracy, artists need to meticulously adjust variables like pose, shape, colour, illumination, and shading of different elements with diverse visual and physical characteristics, i.e., body, hands, face, hair, clothes, and the surrounding environment. To automatically learn these parameters, instead of handcrafting them, is a clear application for deep learning methods. However, even the most

prominent state-of-the-art generative methods are still either limited to specific body parts, e.g., faces or restricted to studios or predefined scenes, falling short when exposed to unconstrained conditions.

In this challenging context, we see *analysis-by-synthesis* frameworks like the one introduced by us in the form of the Semi-DGPose model (Chapter 5), as promising approaches to be pursued. They simultaneously associate three crucial components that functioning together, have the potential to allow for the construction of machine learning *engines* for automatic understanding and generation of synthetic RGB images of people. The association of powerful and specialised **deep discriminative networks** with **deep generative models**, through a **structured and interpretable latent space**, has interesting and desirable characteristics. It potentially enables to leverage the capabilities of discriminative methods for inferring abstract representations from visual data by interrelating them through principled and semantically meaningful probabilistic graphical models. On top of such structured and interpretable abstract representations (probabilistic distributions), the generative models can map samples back to the image space completing the end-to-end analysis-by-synthesis. Finally, a significant by-product of this kind of framework is the possibility of performing semi-supervised learning. It relaxes the need for labelled data and may facilitate transfer-learning since it diminishes the necessity for new labels to related yet different tasks, what is a costly process.

Although some incipient and very recent efforts have been done towards machine learning-based game engines (James Vincent, 2018; Wang et al., 2018), some critical advances rest to be accomplished. They might involve, for instance, the adequate modelling of abstract representations of people in images as probability distributions, the use of probabilistic programming to facilitate the training and testing of such engines, and better performance of deep generative models when dealing with multi-modal data distributions. Despite all the related obstacles, significant progress along these lines of research has the potential to produce not only a drastic change in the current paradigm of *understanding* and *generating* humans in visual data, but in the computer vision area as a whole.

Appendices

A

Critical Percolation as a Framework to Analyze the Training of Deep Networks

CRITICAL PERCOLATION AS A FRAMEWORK TO ANALYZE THE TRAINING OF DEEP NETWORKS

Zohar Ringel
Racah Institute of Physics
The Hebrew University of Jerusalem
zohar.ringel@mail.huji.ac.il.

Rodrigo de Bem*
Department of Engineering Science
University of Oxford
rodrigo@robots.ox.ac.uk

ABSTRACT

In this paper we approach two relevant deep learning topics: i) tackling of graph structured input data and ii) a better understanding and analysis of deep networks and related learning algorithms. With this in mind we focus on the topological classification of reachability in a particular subset of planar graphs (Mazes). Doing so, we are able to model the topology of data while staying in Euclidean space, thus allowing its processing with standard CNN architectures. We suggest a suitable architecture for this problem and show that it can express a perfect solution to the classification task. The shape of the cost function around this solution is also derived and, remarkably, does not depend on the size of the maze in the large maze limit. Responsible for this behavior are rare events in the dataset which strongly regulate the shape of the cost function near this global minimum. We further identify an obstacle to learning in the form of poorly performing local minima in which the network chooses to ignore some of the inputs. We further support our claims with training experiments and numerical analysis of the cost function on networks with up to 128 layers.

1 INTRODUCTION

Deep convolutional networks have achieved great success in the last years by presenting human and super-human performance on many machine learning problems such as image classification, speech recognition and natural language processing (LeCun et al. (2015)). Importantly, the data in these common tasks presents particular statistical properties and it normally rests on regular lattices (e.g. images) in Euclidean space (Bronstein et al. (2016)). Recently, more attention has been given to other highly relevant problems in which the input data belongs to non-Euclidean spaces. Such kind of data may present a graph structure when it represents, for instance, social networks, knowledge bases, brain activity, protein-interaction, 3D shapes and human body poses. Although some works found in the literature propose methods and network architectures specifically tailored to tackle graph-like input data (Bronstein et al. (2016); Bruna et al. (2013); Henaff et al. (2015); Li et al. (2015); Masci et al. (2015a;b)), in comparison with other topics in the field this one is still not vastly investigated.

Another recent focus of interest of the machine learning community is in the detailed analysis of the functioning of deep networks and related algorithms (Daniely et al. (2016); Ghahramani (2015)). The minimization of high dimensional non-convex loss function by means of stochastic gradient descent techniques is theoretically unlikely, however the successful practical achievements suggest the contrary. The hypothesis that very deep neural nets do not suffer from local minima (Dauphin et al. (2014)) is not completely proven (Swirszcz et al. (2016)). The already classical adversarial examples (Nguyen et al. (2015)), as well as new doubts about supposedly well understood questions, such as generalization (Zhang et al. (2016)), bring even more relevance to a better understanding of the methods.

In the present work we aim to advance simultaneously in the two directions described above. To accomplish this goal we focus on the topological classification of graphs (Perozzi et al. (2014);

*Rodrigo de Bem is also Assistant Professor at the Federal University of Rio Grande, Rio Grande, Brazil (rodrigobem@furg.br).

Scarselli et al. (2009)). However, we restrict our attention to a particular subset of planar graphs constrained by a regular lattice. The reason for that is threefold: i) doing so we still touch upon the issue of real world graph structured data, such as the 2D pose of a human body (Andriluka et al. (2014); Jain et al. (2016)) or road networks (Masucci et al. (2009); Viana et al. (2013)); ii) we maintain the data in Euclidean space, allowing its processing with standard CNN architectures; iii) this particular class of graphs has various non-trivial statistical properties derived from percolation theory and conformal field theories (Cardy (2001); Langlands et al. (1994); Smirnov & Werner (2001)), allowing us to analytically compute various properties of a deep CNN proposed by the authors to tackle the problem.

Specifically, we introduce Maze-testing, a specialized version of the reachability problem in graphs (Yu & Cheng (2010)). In Maze-testing, random mazes, defined as L by L binary images, are classified as solvable or unsolvable according to the existence of a path between given starting and ending points in the maze (vertices in the planar graph). Other recent works approach maze problems without framing them as graphs (Tamar et al. (2016); Oh et al. (2017); Silver et al. (2017)). However, to do so with mazes (and maps) is a common practice in graph theory (Biggs et al. (1976); Schrijver (2012)) and in applied areas, such as robotics (Elfes (1989); Choset & Nagatani (2001)). Our Maze-testing problem enjoys a high degree of analytical tractability, thereby allowing us to gain important theoretical insights regarding the learning process. We propose a deep network to tackle the problem that consists of $O(L^2)$ layers, alternating convolutional, sigmoid, and skip operations, followed at the end by a logistic regression function. We prove that such a network can express an exact solution to this problem which we call the optimal-BFS (breadth-first search) minimum. We derive the shape of the cost function around this minimum. Quite surprisingly, we find that gradients around the minimum do not scale with L. This peculiar effect is attributed to rare events in the data.

In addition, we shed light on a type of sub-optimal local minima in the cost function which we dub "neglect minima". Such minima occur when the network discards some important features of the data samples, and instead develops a sub-optimal strategy based on the remaining features. Minima similar in nature to the above optimal-BFS and neglect minima are shown to occur in numerical training and dominate the training dynamics. Despite the fact the Maze-testing is a toy problem, we believe that its fundamental properties can be observed in real problems, as is frequently the case in natural phenomena (Schmidt & Lipson (2009)), making the presented analytical analysis of broader relevance.

Additionally important, our framework also relates to neural network architectures with augmented memory, such as Neural Turing Machines (Graves et al. (2014)) and memory networks (Weston et al. (2014); Sukhbaatar et al. (2015)). The hot-spot images (Fig. 7), used to track the state of our graph search algorithm, may be seen as an external memory. Therefore, to observe how activations spread from the starting to the ending point in the hot-spot images, and to analyze errors and the landscape of the cost function (Sec. 5), is analogous to analyze how errors occur in the memory of the aforementioned architectures. This connection gets even stronger when such memory architectures are employed over graph structured data, to perform task such as natural language reasoning and graph search (Weston et al. (2015); Johnson (2017); Graves et al. (2016)). In these cases, it can be considered that their memories in fact encode graphs, as it happens in our framework. Thus, the present analysis may eventually help towards a better understanding of the cost functions of memory architectures, potentially leading to improvements of their weight initialization and optimization algorithms thereby facilitating training (Mishkin & Matas (2015)).

The paper is organized as follows: Sec. 2 describes in detail the Maze-testing problem. In Sec. 3 we suggest an appropriate architecture for the problem. In Sec. 4 we describe an optimal set of weights for the proposed architecture and prove that it solves the problem exactly. In Sec. 5 we report on training experiments and describe the observed training phenomena. In Sec. 6 we provide an analytical understanding of the observed training phenomena. Finally, we conclude with a discussion and an outlook.

2 MAZE-TESTING

Let us introduce the Maze-testing classification problem. Mazes are constructed as a random two dimensional, $L \times L$, black and white array of cells (I) where the probability (ρ) of having a black cell is given by $\rho_c = 0.59274(6)$, while the other cells are white. An additional image (H_0), called

the initial hot-spot image, is provided. It defines the starting point by being zero (*Off*) everywhere except on a 2×2 square of cells having the value 1 (*On*) chosen at a random position (see Fig.7). A Maze-testing sample (i.e. a maze and a hot-spot image) is labelled *Solvable* if the ending point, defined as a 2×2 square at the center of the maze, is reachable from the starting point (defined by the hot-spot image) by moving horizontally or vertically along black cells. The sample is labelled *Unsolvable* otherwise.

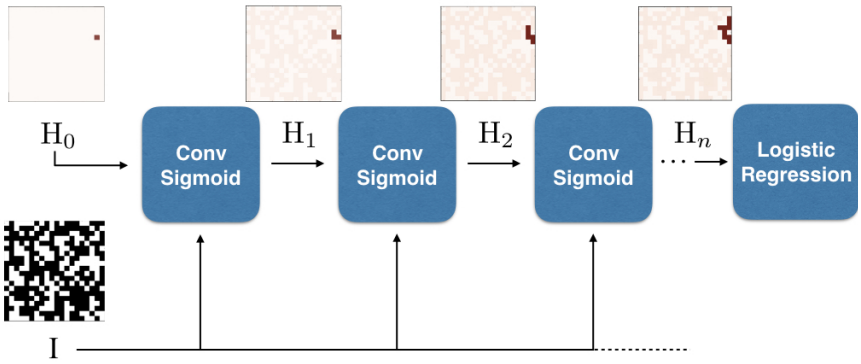


Figure 1: Dataset, Architecture, and the Breadth-First Search optimum. A maze-testing sample consists of a maze (I) and an initial hot-spot image (H_0). The proposed architecture processes H_0 by generating a series of hot-spot images ($\mathrm{H}_{i>0}$) which are of the same dimension as H_0 however their pixels are not binary but rather take on values between 0 (*Off*, pale-orange) and 1 (*On*, red). This architecture can represent an optimal solution, wherein the red region in H_0 spreads on the black cluster in I to which it belongs. Once the spreading has exhausted, the Solvable/Unsolvable label is determined by the values of H_n at center (ending point) of the maze. In the above example, the maze in question is Unsolvable, therefore the *On* cells do not reach the ending point at the center of H_n.

A maze in a Maze-testing sample has various non-trivial statistical properties which can be derived analytically based on results from percolation theory and conformal field theory (Cardy (2001); Langlands et al. (1994); Smirnov & Werner (2001)). Throughout this work we directly employ such statistical properties, however we refer the reader to the aforementioned references for further details and mathematical derivations.

At the particular value chosen for ρ, the problem is at the percolation-threshold which marks the phase transition between the two different connectivity properties of the maze: below ρ_c the chance of having a solvable maze decays exponentially with r (the geometrical distance between the ending and starting points). Above ρ_c it tends to a constant at large r. Exactly at ρ_c the chance of having a solvable maze decays as a power law ($1/r^\eta, \eta = 5/24$). We note in passing that although Maze-testing can be defined for any ρ, only the choice $\rho = \rho_c$ leads to a computational problem whose typical complexity increases with L.

Maze-testing datasets can be produced very easily by generating random arrays and then analyzing their connectivity properties using breadth-first search (BFS), whose worse case complexity is $O(L^2)$. Notably, as the system size grows larger, the chance of producing solvable mazes decay as $1/L^\eta$, and so, for very large L, the labels will be biased towards unsolvable. There are several ways to de-bias the dataset. One is to select an unbiased subset of it. Alternatively, one can gradually increase the size of the starting-point to a starting-square whose length scales as L^η. Unless stated otherwise, we simply leave the dataset biased but define a normalized test error (err), which is proportional to the average mislabeling rate of the dataset divided by the average probability of being solvable.

3 THE ARCHITECTURE

Here we introduce an image classification architecture to tackle the Maze-testing problem. We frame maze samples as a subclass of planar graphs, defined as regular lattices in the Euclidean space, which can be handle by regular CNNs. Our architecture can express an exact solution to the problem and, at least for small Mazes ($L \leq 16$), it can find quite good solutions during training. Although applicable to general cases, graph oriented architectures find it difficult to handle large sparse graphs due to

regularization issues (Henaff et al. (2015); Li et al. (2015)), whereas we show that our architecture can perform reasonably well in the planar subclass.

Our network, shown in Fig. (7), is a deep feedforward network with skip layers, followed by a logistic regression module. The deep part of the network consists of n alternating convolutional and sigmoid layers. Each such layer (i) receives two $L \times L$ images, one corresponding to the original maze (I) and the other is the output of the previous layer (H_{i-1}). It performs the operation $\mathrm{H}_i = \sigma(K_{hot} * \mathrm{H}_{i-1} + K * \mathrm{I} + b)$, where $*$ denotes a $2D$ convolution, the K convolutional kernel is 1×1, the K_{hot} kernel is 3×3, b is a bias, and $\sigma(x) = (1+e^{-x})^{-1}$ is the sigmoid function. The logistic regression layer consists of two perceptrons ($j = 0, 1$), acting on H_n as $[p_0, p_1]^T = W_j\vec{\mathrm{H}}_n + \vec{b}_{reg}$, where $\vec{\mathrm{H}}_n$ is the rasterized/flattened version of H_n, W_j is a $2 \times L^2$ matrix, and $\vec{b}_{reg}$ is a vector of dimension 2. The logistic regression module outputs the label *Solvable* if $p_1 \geq p_0$ and *Unsolvable* otherwise. The cost function we used during training was the standard negative log-likelihood.

4 An optimal solution: The breadth-first search minimum

As we next show, the architecture above can provide an exact solution to the problem by effectively forming a cellular automaton executing a breadth-first search (BFS). A choice of parameters which achieves this is $\lambda \geq \lambda_c = 9.727\pm0.001$, $K_{hot} = [[0, \lambda, 0], [\lambda, \lambda, \lambda], [0, \lambda, 0]]$, $K = 5\lambda_c$, $b = -5.5\lambda_c$, $[W]_{jq} = (-1)^j\lambda\delta_{q_{center}q}$ and $\vec{b}_{reg} = [0.5\lambda, -0.5\lambda]^T$, where q_{center} is the index of $\vec{\mathrm{H}}_n$ which corresponds to the center of the maze.

Let us explain how the above neural network processes the image (see also Fig. 7). Initially H_0 is *On* only at the starting-point. Passing through the first convolutional-sigmoid layer it outputs H_1 which will be *On* (i.e. have values close to one) on all black cells which neighbor the *On* cells as well as on the original starting point. Thus *On* regions spread on the black cluster which contains the original starting-point, while white clusters and black clusters which do not contain the starting-point remain *Off* (close to zero in H_i). The final logistic regression layer simply checks whether one of the 2×2 cells at the center of the maze are *On* and outputs the labels accordingly.

To formalize the above we start by defining two activation thresholds, v_l and v_h, and refer to activations which are below v_l as being *Off* and to those above v_h as being *On*. The quantity v_l is defined as the smallest of the three real solutions of the equation $v_l = \sigma(5v_l - 0.5\lambda)$. Notably we previously chose $\lambda > \lambda_c$ as this is the critical value above which three real solutions to v_l (rather than one) exist. For v_h we choose 0.9.

Next, we go case by case and show that the action of the convolutional-sigmoid layer switches activations between *Off* and *On* just as a BFS would. This amounts to bounding the expression $\sigma(K_{hot} * \mathrm{H}_{i-1} + K * \mathrm{I} + b)$ for all possibly 3×3 sub-arrays of H_{i-1} and 1×1 sub-arrays of I. There are thus 2^{10} possibilities to be examined.

Figure 2 shows the desired action of the layer on three important cases (A-C). Each case depicts the maze shape around some arbitrary point x, the hot-spot image around x before the action of the layer (H_{i-1}), and the desired action of the layer (H_i). **Case A.** Having a white cell at x implies $\mathrm{I}[x] = 0$ and therefore the argument of the above sigmoid is smaller than $-0.5\lambda_c$ this regardless of H_{i-1} at and around x. Thus $\mathrm{H}_i[x] < v_l$ and so it is *Off*. As the 9 activations of H_{i-1} played no role, case A covers in fact 2^9 different cases. **Case B.** Consider a black cell at x, with H_{i-1} in its vicinity all being *Off* (vicinity here refers to x and its 4 neighbors). Here the argument is smaller or equal to $5v_l - 0.5\lambda_c$, and so the activation remains *Off* as desired. Case B covers 2^4 cases as the values of H_{i-1} on the 4 corners were irrelevant. **Case C.** Consider a black cell at x with one or more *On* activations of H_{i-1} in its vicinity. Here the argument is larger than $v_h\lambda_c - 0.5\lambda_c = 0.4\lambda_c$. The sigmoid is then larger than 0.97 implying it is *On*. Case C covers $2^4(2^5 - 1)$ different cases. Since $2^9 + 2^4 + 2^4(2^5 - 1) = 2^{10}$ we exhausted all possible cases. Lastly it can be easily verified that given an *On* (*Off*) activation at the center of the full maze the logistic regression layer will output the label *Solvable* (*Unsolvable*).

Let us now determine the required depth for this specific architecture. The previous analysis tells us that at depth d unsolvable mazes would always be labelled correctly however solvable mazes would be label correctly only if the shortest-path between the starting-point and the center is d or less. The worse case scenario thus occurs when the center of the maze and the starting-point are

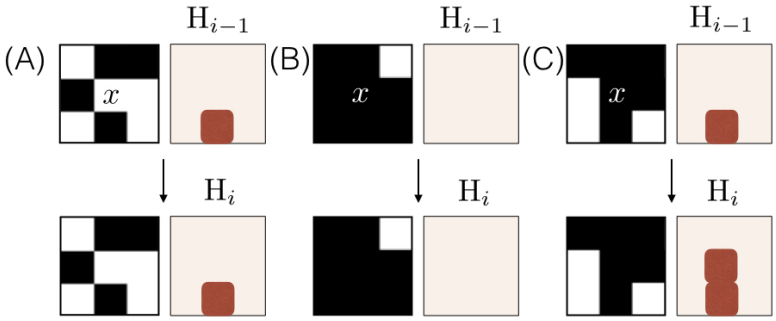


Figure 2: The three cases for the action of the convolutional-sigmoid layers. These cases are representative of the three sets corresponding to all possible states of the binary maze images (I) and the hot-spot images (H_{i-1} and H_i), with values between 0 (*Off*, pale-orange) and 1 (*On*, red).

connected by a one dimensional curve twisting its way along $O(L^2)$ sites. Therefore, for perfect performance the network depth would have to scale as the number of sites namely $n = O(L^2)$. A tighter but probabilistic bound on the minimal depth can be established by borrowing various results from percolation theory. It is known, from Zhou et al. (2012), that the typical length of the shortest path (l) for critical percolation scales as $r^{d_{min}}$, where r is the geometrical distance and $d_{min} = 1.1(3)$. Moreover, it is known that the probability distribution $P(l|r)$ has a tail which falls as l^{-2} for $l \approx> r^{d_{min}}$ (Dokholyan et al. (1999)). Consequently, the chance that at distance r the shortest path is longer than $r^{d_{min}} r^a$, where a is some small positive number, decays to zero and so, d should scale as L with a power slightly larger than d_{min} (say $n = L^{1.2}$).

5 Training Experiments

We have performed several training experiments with our architecture on $L = 16$ and $L = 32$ mazes with depth $n = 16$ and $n = 32$ respectively, datasets of sizes $M = 1000$, $M = 10000$, and $M = 50000$. Unless stated otherwise we used a batch size of 20 and a learning rate of 0.02. In the following, we split the experiments into two different groups corresponding to the related phenomena observed during training, which will the analyzed in detail in the next section.

Optimal-BFS like minima. For $L = 16$, $M = 10000$ mazes and a positive random initialization for K_{hot} and K in $[0, \sqrt{6/8}]$ the network found a solution with $\approx 9\%$ normalized test error performance in 3 out of the 7 different initializations (baseline test error was 50%). In all three successful cases the minima was a variant of the Optimal-BFS minima which we refer to as the checkerboard-BFS minima. It is similar to the optimal-BFS but spreads the *On* activations from the starting-point using a checkerboard pattern rather than a uniform one, as shown in Fig. 3. The fact that it reaches $\approx 9\%$ test error rather than zero is attributed to this checkerboard behavior which can occasionally miss out the exit point from the maze.

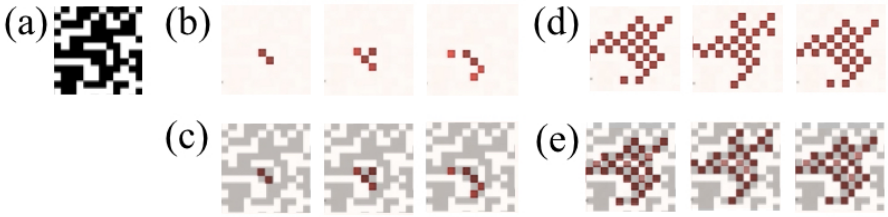


Figure 3: Dynamics of activations for the checkerboard BFS minima obtained in training. The activations in $H_{1..3}$ and $H_{11..13}$ are shown in (b) and (d), respectively, along with the corresponding maze (a). Superposition of the maze on top of $H_{1..3}$ and $H_{11..13}$ are shown in (c) and (e), respectively. See also a short movie with the checkerboard activations at `https://youtu.be/t-_TDkt3ER4`.

Neglect minima. Again for $L = 16$ but allowing for negative entries in K and K_{hot} test error following 14 attempts and 500 epochs did not improve below 44%. Analyzing the weights of the

network, the 6% improvement over the baseline error (50%) came solely from identifying the inverse correlation between many white cells near the center of the maze and the chance of being solvable. Notably, this heuristic approach completely neglects information regarding the starting-point of the maze. For $L = 32$ mazes, despite trying several random initialization strategies including positive entries, dataset sizes, and learning rates, the network always settled into such a partial neglect minimum. In an unsuccessful attempt to push the weights away from such partial neglect behavior, we performed further training experiments with a biased dataset in which the maze itself was uncorrelated with the label. More accurately, marginalizing over the starting-point there is an equal chance for both labels given any particular maze. To achieve this, a maze shape was chosen randomly and then many random locations were tried-out for the starting-point using that same maze. From these, we picked 5 that resulted in a Solvable label and 5 that resulted in an Unsolvable label. Maze shapes which were always Unsolvable were discarded. Both the $L = 16$ and $L = 32$ mazes trained on this biased dataset performed poorly and yielded 50% test error. Interestingly they improved their cost function by settling into weights in which $b \approx -10$ is large compared to $[K_{hot}]_{ij} <\approx 1$ while W and $\vec{b}$ were close to zero (order of 0.01). We have verified that such weights imply that activations in the last layer have a negligible dependence on the starting-point and a weak dependence on the maze shape. We thus refer to this minimum as a "total neglect minimum".

6 Cost function landscape and the observed training phenomena

Here we seek an analytical understanding of the aforementioned training phenomena through the analysis of the cost function around solutions similar or equal to those the network settled into during training. Specifically we shall first study the cost function landscape around the optimal-BFS minimum. As would become clearer at the end of that analysis, the optimal BFS shares many similarities with the checkerboard-BFS minimum obtained during training and one thus expects a similar cost function landscape around both of these. The second phenomena analyzed below is the total neglect minimum obtained during training on the biased dataset. The total neglect minimum can be thought of as an extreme version of the partial neglect minima found for $L = 32$ in the original dataset.

6.1 The shape of the cost function near the optimal-BFS minimum

Our analysis of the cost function near the optimal-BFS minimum will be based on two separate models capturing the short and long scale behavior of the network near this miminum. In the first model we approximate the network by linearizing its action around weak activations. This model would enable us to identify the density of what we call "bugs" in the network. In the second model we discretize the activation levels of the neural network into binary variables and study how the resulting cellular automaton behaves when such bugs are introduced. Figure 4 shows a numerical example of the dynamics we wish to analyze.

Figure 4: Activation dynamics for weights close to the optimal solution ($\lambda = \lambda_c - 0.227$). Up to layer 19 ($H_{19}$) the *On* activations spread according to BFS however at H_{20} a very faint localized unwanted *On* activation begins to develop (a bug) and quickly saturates (H_{23}). Past this point BFS dynamics continues normally but spreads both the original and the unwanted *On* activations. While not shown explicitly, *On* activations still appear only on black maze cells. Notably the bug developed in rather large black region as can be deduced from the large red region in its origin. See also a short movie showing the occurrence of this bug at `https://youtu.be/2I436BVAVdM` and more bugs at `https://youtu.be/kh-AfOo4TkU`. At `https://youtu.be/t-_TDkt3ER4` a similar behavior is shown for the checkerboard-BFS.

6.1.1 LINEARIZATION AROUND THE OPTIMAL-BFS MINIMUM AND THE EMERGENCE OF BUGS

Unlike an algorithm, a neural network is an analog entity and so a-priori there are no sharp distinctions between a functioning and a dis-functioning neural network. An algorithm can be debugged and the bug can be identified as happening at a particular time step. However it is unclear if one can generally pin-point a layer and a region within where a deep neural network clearly malfunctioned. Interestingly we show that in our toy problem such pin-pointing of errors can be done in a sharp fashion by identifying fast and local processes which cause an unwanted switching been *Off* and *On* activations in H_i (see Fig. 4). We call these events bugs, as they are local, harmful, and have a sharp meaning in the algorithmic context.

Below we obtain asymptotic expressions for the chance of generating such bugs as the network weights are perturbed away from the optimal-BFS minimum. The main result of this subsection, derived below, is that the density of bugs (or chance of bug per cell) scales as

$$\rho_{bug} \propto e^{\frac{C'}{(\lambda-\lambda_c)}}, \tag{1}$$

for $(\lambda - \lambda_c) <\approx 0$ and zero for $\lambda - \lambda_c >= 0$ where $C' \approx 1.7$. Following the analysis below, we expect the same dependence to hold for generic small perturbations only with different C' and λ_c. We have tested this claim numerically on several other types of perturbations (including ones that break the $\pi/2$ rotational symmetry of the weights) and found that it holds.

To derive Eq. (1), we first recall the analysis in Sec. 4, initially as it is decreased λ has no effect but to shift v_l (the *Off* activation threshold) to a higher value. However, at the critical value ($\lambda = \lambda_c, v_l = v_{l,c}$) the solution corresponding to v_l vanishes (becomes complex) and the correspondence with the BFS algorithm no longer holds in general. This must not mean that all *Off* activations are no longer stable. Indeed, recall that in Sec. 4 the argument that a black *Off* cell in the vicinity of *Off* cells remains *Off* (Fig. 2, Case B) assumed a worse case scenario in which all the cells in its vicinity where both *Off*, black, and had the maximal *Off* activation allowed (v_l). However, if some cells in its vicinity are white, their *Off* activations levels are mainly determined by the absence of the large K term in the sigmoid argument and orders of magnitude smaller than v_l. We come to the conclusion that black *Off* cells in the vicinity of many white cells are less prone to be spontaneously turned *On* than black *Off* cells which are part of a large cluster of black cells (see also the bug in Fig. 4). In fact using the same arguments one can show that infinitesimally below λ_c only uniform black mazes will cause the network to malfunction.

To further quantify this, consider a maze of size $l \times l$ where the hot-spot image is initially all zero and thus *Off*. Intuitively this hot-spot image should be thought of as a sub-area of a larger maze located far away from the starting-point. In this case a functioning network must leave all activation levels below v_l. To assess the chance of bugs we thus study the probability that the output of the final convolutional-sigmoid layer will have one or more *On* cells.

To this end, we find it useful to linearize the system around low activation yielding (see the Appendix for a complete derivation)

$$\psi_n(r_b) = \tilde{\lambda} \left(\psi_{n-1}(r_b) + \sum_{\langle r'_b, r_b \rangle} \psi_{n-1}(r'_b) \right) + O(\tilde{\lambda}\psi_n^2), \tag{2}$$

where r_b denotes black cells ($\mathrm{I}(r_b) = 1$), the sum is over the nearest neighboring black cells to r_b, $\psi_n(r) = \mathrm{H}_n(r) - v_{l,c}$, and $\tilde{\lambda} = \lambda \frac{d\sigma}{dx}(\sigma^{-1}(v_{l,c}))$.

For a given maze (I), Eq. (2), defines a linear Hermitian operator (L_{I}) with random off-diagonal matrix elements dictated by I via the restriction of the off-diagonal terms to black cells. Stability of *Off* activations is ensured if this linear operator is contracting or equivalently if all its eigenvalues are smaller than 1 in magnitude.

Hermitian operators with local noisy entries have been studied extensively in physics, in the context of disordered systems and Anderson localization (Kramer & MacKinnon (1993)). Let us describe the main relevant results. For almost all I's the spectrum of L consists of localized eigenfunctions (ϕ_m). Any such function is centered around a random site (x_m) and decays exponentially away from that site with a decay length of χ which in our case would be a several cells long. Thus given ϕ_m with an eigenvalue $|E_m| > 1$, t repeated actions of the convolutional-sigmoid layer will make $\psi_n[x]$ in

a χ vicinity of x_m grow in size as $e^{E_m t}$. Thus $(|E_m| - 1)^{-1}$ gives the characteristic time it takes these localized eigenvalue to grow into an unwanted localized region with an *On* activation which we define as a bug.

Our original question of determining the chance of bugs now translates into a linear algebra task: finding, $N_{\tilde{\lambda}}$, the number of eigenvalues in L_{I} which are larger than 1 in magnitude, averaged over I, for a given λ. Since $\tilde{\lambda}$ simply scales all eigenvalues one finds that $N_{\tilde{\lambda}}$ is the number of eigenvalues larger than $\tilde{\lambda}^{-1}$ in L_{I} with $\tilde{\lambda} = 1$. Analyzing this latter operator, it is easy to show that the maximal eigenvalues occurs when $\phi_n(r)$ has a uniform pattern on a large uniform region where the I is black. Indeed if I contains a black uniform true box of dimension $l_u \times l_u$, the maximal eigenvalue is easily shown to be $E_{l_u} = 5 - 2\pi^2/(l_u)^2$. However the chance that such a uniform region exists goes as $(l/l_u)^2 e^{\log(\rho_c) l_u^2}$ and so $P(\Delta E) \propto l^2 e^{\frac{\log(\rho_c) 2\pi^2}{(\Delta E)}}$, where $\Delta E = 5 - E$. This reasoning is rigorous as far as lower bounds on $N_{\tilde{\lambda}}$ are concerned, however it turns out to capture the functional behavior of $P(\Delta E)$ near $\Delta E = 0$ accurately (Johri & Bhatt (2012)) which is given by $P(\Delta E \to 0_+) = l^2 e^{-\frac{C}{(\Delta E)}}$, where the unknown constant C captures the dependence on various microscopic details. In the Appendix we find numerically that $C \approx 0.7$. Following this we find $N_{\tilde{\lambda}} \propto l^2 \int_0^{\Delta E_\lambda} dx P(x)$ where $\Delta E_\lambda = 5 - \tilde{\lambda}^{-1} \geq 0$. The range of integration is chosen to includes all eigenvalues which, following a multiplication by $\tilde{\lambda}$, would be larger than 1.

To conclude we found the number of isolated unwanted *On* activations which develop on $l \times l$ *Off* regions. Dividing this number by l^2 we obtain the density of bugs (ρ_{bug}) near $\lambda \approx \lambda_c$. The last technical step is thus to express ρ_{bug} in terms of λ. Focusing on the small ρ_{bug} region or $\Delta E \to 0_+$, we find that $\Delta E = 0$ occurs when $\frac{d\sigma}{dx}(\sigma^{-1}(\eta_\infty(\lambda))) = 1/(5\lambda)$, $\tilde{\lambda} = 1/5$, and $\lambda = \lambda_c = 9.72(7)$. Expanding around $\lambda = \lambda_c$ we find $\Delta E_\lambda = \frac{49-\lambda_c}{10\lambda_c}(\lambda_c - \lambda) + O((\lambda_c - \lambda)^2)$. Approximating the integral over $P(x)$ and taking the leading scale dependence, we arrive at Eq. (1) with $C' = C\frac{10\lambda_c}{49-\lambda_c}$.

6.1.2 Effects of bugs on breadth-first search

In this subsection we wish to understand the large scale effect of ρ_{bug} namely, its effect on the test error and the cost function. Our key results here are that

$$err \propto \rho_{bug}^{5/91} \propto e^{\frac{5C'/91}{\lambda - \lambda_c}}, \tag{3}$$

for $C' d_f^{-1}/\log(L) + O(\log^{-2}(L)) < (\lambda_c - \lambda) < C'$ where ($d_f = 91/48$). Notably this expression is independent of L. In the domain $(\lambda_c - \lambda) <\approx C' d_f^{-1}/\log(L)$ or equivalently $\rho_{bug} <\approx L^{-d_f}$ a weak dependence on L remains and

$$err \propto L^{2-5/24} e^{\frac{C'}{\lambda - \lambda_c}}, \tag{4}$$

despite its appearance it can be verified that the above right hand side is smaller than $L^{-5/48}$ within its domain of applicability. Figure (5) shows a numerical verification of this last result (see Appendix for further details).

To derive Eqs. (3) and (4), we note that as a bug is created in a large maze, it quickly switches *On* the cells within the black "room" in which it was created. From this region it spreads according to BFS and turns *On* the entire cluster connected to the buggy room (see Fig. 4). To asses the effect this bug has on performance first note that solvable mazes would be labelled Solvable regardless of bugs however unsolvable mazes might appear solvable if a bug occurs on a cell which is connected to the center of the maze. Assuming we have an unsolvable maze, we thus ask what is the chance of it being classified as solvable.

Given a particular unsolvable maze instance (I), the chance of classifying it as solvable is given by $p_{err}(I) = 1 - (1 - \rho_{bug})^s = 1 - e^{-\rho_{bug} s} + O(\rho_{bug}^2)$ where s counts the number of sites in the cluster connected to the central site (central cluster). The probability distribution of s for percolation is known and given by $p(s) = Bs^{1-\tau}, \tau = 187/91$ (Cardy & Ziff (2003)), with B being an order of one constant which depends on the underlying lattice. Since clusters have a fractional dimension, the maximal cluster size is L^{d_f}. Consequently, $p_{err}(I)$ averaged over all I instances is given by $p_{err} = \int_0^{L^{d_f}} p(s)\left[1 - e^{-\rho_{bug} s}\right] ds$, which can be easily expressed in terms of Gamma functions

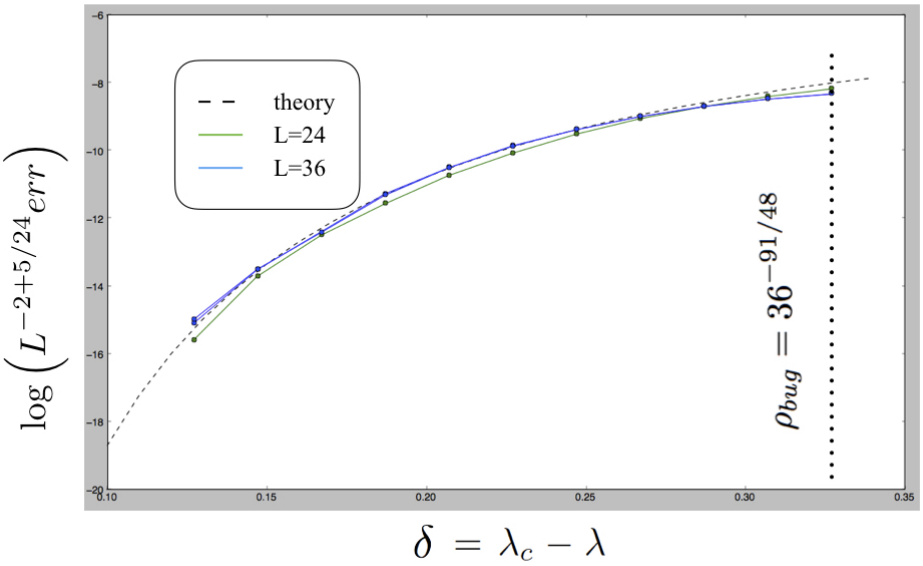


Figure 5: Logarithm of the numerically obtained err scaled by $L^{-2+5/24}$ as a function of a deviation (δ) from the optimal-BFS weights for two maze sizes along with a fit to Eq. (4). The dotted vertical line marks the end of the domain of applicability of Eq. (4).

$(\Gamma(x), \Gamma(a,x))$ (see Abramowitz (1974)). In the limit of $\rho_{bug} <\approx L^{-d_f}$, where its derivatives with respect to ρ_{bug} are maximal, it simplifies to

$$p_{err} = (\tau-2)^{-1} B \rho_{bug} L^{d_f(3-\tau)} \propto \rho_{bug} L^{2-5/24}, \tag{5}$$

whereas for $\rho_{bug} > L^{-d_f}$, its behavior changes to $p_{err} = (-B\Gamma(2-\tau))\rho_{bug}^{(\tau-2)} \propto \rho_{bug}^{5/91}$. Notably once ρ_{bug} becomes of order one, several of the approximation we took break down.

Let us relate p_{err} to the test error (err). In Sec. (2) the cost function was defined as the mislabeling chance over the average chance of being solvable ($p_{solvable}$). Following the above discussion the mislabelling chance is $p_{err}p_{solvable}$ and consequently $err = p_{err}$. Combining Eqs. 1 and 5 we obtain our key results, Eqs. (3, 4)

As a side, one should appreciate a potential training obstacle that has been avoided related to the fact that $err \propto \rho_{big}^{5/91}$. Considering $L \to \infty$, if ρ_{bug} was simply proportional to $(\lambda_c - \lambda)$, err will have a sharp singularity near zero. For instance, as one reduces err by a factor of $1/e$, the gradients increase by $e^{86/5} \approx 3E+7$. These effects are in accordance with ones intuition that a few bugs in a long algorithm will typically have a devastating effect on performance. Interestingly however, the essential singularity in $\rho_{bug}(\lambda)$, derived in the previous section, completely flattens the gradients near λ_c.

Thus the essentially singularity which comes directly from rare events in the dataset strongly regulates the test error and in a related way the cost function. However it also has a negative side-effect concerning the robustness of generalization. Given a finite dataset the rarity of events is bounded and so having $\lambda < \lambda_c$ may still provide perfect performance. However when encountering a larger dataset some samples with rarer events (i.e. larger black region) would appear and the network will fail sharply on these (i.e. the wrong prediction would get a high probability). Further implications of this dependence on rare events on training and generalization errors will be studied in future work.

6.2 Cost function near a total neglect minima

To provide an explanation for this phenomena let us divide the activations of the upper layer to its starting-point dependent and independent parts. Let H_n denote the activations at the top layer. We expand them as a sum of two functions

$$\mathrm{H}_n = \alpha A(H_0, \mathrm{I}) + \beta B(\mathrm{I}) \tag{6}$$

where the function A and B are normalized such that their variance on the data (α and β, respectively) is 1. Notably near the reported total neglect minima we found that $\alpha \ll \beta \approx e^{-10}$. Also note that for the biased dataset the maze itself is uncorrelated with the labels and thus β can be thought of as noise. Clearly any solution to the Maze testing problem requires the starting-point dependent part (α) to become larger than the independent part (β). We argue however that in the process of increasing α the activations will have to go through an intermediate "noisy" region. In this noisy region α grows in magnitude however much less than β and in particular obeys $\alpha < \beta^2$. As shown in the Appendix the negative log-likelihood, a commonly used cost function, is proportional to $\beta^2 - \alpha$ for $\alpha, \beta \ll 1$. Thus it penalizes random false predictions and, within a region obeying $\alpha < \beta^2$ it has a minimum (global with respect to that region) when $\alpha = \beta = 0$. The later being the definition of a total neglect minima.

Establishing the above $\alpha \ll \beta^2$ conjecture analytically requires several pathological cases to be examined and is left for future work. In this work we provide an argument for its typical correctness along with supporting numerics in the Appendix.

A deep convolution network with a finite kernel has a notion of distance and locality. For many parameters ranges it exhibits a typical correlation length (χ). That is a scale beyond which two activations are statistically independent. Clearly to solve the current problem χ has to grow to an order of L such that information from the input reaches the output. However as χ gradually grows, relevant and irrelevant information is being mixed and propagated onto the final layer. While β depends on information which is locally accessible at each layer (i.e. the maze shape), α requires information to travel from the first layer to the last. Consequently α and β are expected to scale differently, as $e^{-L/\chi}$ and $e^{-1/\chi}$ resp. (for $\chi << L$). Given this one finds that $\alpha \ll \beta^2$ as claimed.

Further numerical support of this conjecture is shown in the Appendix where an upper bound on the ratio α/β^2 is studied on 100 different paths leading from the total neglect miminum found during training to the checkerboard-BFS minimum. In all cases there is a large region around the total neglect minimum in which $\alpha \ll \beta^2$.

7 CONCLUSIONS

Despite their black-box reputation, in this work we were able to shed some light on how a particular deep CNN architecture learns to classify topological properties of graph structured data. Instead of focusing our attention on general graphs, which would correspond to data in non-Euclidean spaces, we restricted ourselves to planar graphs over regular lattices, which are still capable of modelling real world problems while being suitable to CNN architectures.

We described a toy problem of this type (Maze-testing) and showed that a simple CNN architecture can express an exact solution to this problem. Our main contribution was an asymptotic analysis of the cost function landscape near two types of minima which the network typically settles into: BFS type minima which effectively executes a breadth-first search algorithm and poorly performing minima in which important features of the input are neglected.

Quite surprisingly, we found that near the BFS type minima gradients do not scale with L, the maze size. This implies that global optimization approaches can find such minima in an average time that does not increase with L. Such very moderate gradients are the result of an essential singularity in the cost function around the exact solution. This singularity in turn arises from rare statistical events in the data which act as early precursors to failure of the neural network thereby preventing a sharp and abrupt increase in the cost function.

In addition we identified an obstacle to learning whose severity scales with L which we called neglect minima. These are poorly performing minima in which the network neglects some important features relevant for predicting the label. We conjectured that these occur since the gradual incorporation of these important features in the prediction requires some period in the training process in which predictions become more noisy. A ”wall of noise" then keeps the network in a poorly performing state.

It would be interesting to study how well the results and lessons learned here generalize to other tasks which require very deep architectures. These include the importance of rare-events, the essential

singularities in the cost function, the localized nature of malfunctions (bugs), and neglect minima stabilized by walls of noise.

These conjectures potentially could be tested analytically, using other toy models as well as on real world problems, such as basic graph algorithms (e.g. shortest-path) (Graves et al. (2016)); textual reasoning on the bAbI dataset (Weston et al. (2015)), which can be modelled as a graph; and primitive operations in "memory" architectures (e.g. copy and sorting) (Graves et al. (2014)). More specifically the importance of rare-events can be analyzed by studying the statistics of errors on the dataset as it is perturbed away from a numerically obtained minimum. Technically one should test whether the perturbation induces an typical small deviation of the prediction on most samples in the dataset or rather a strong deviation on just a few samples. Bugs can be similarly identified by comparing the activations of the network on the numerically obtained minimum and on some small perturbation to that minimum while again looking at typical versus extreme deviations. Such an analysis can potentially lead to safer and more robust designs were the network fails typically and mildly rather than rarely and strongly.

Turning to partial neglect minima these can be identified provided one has some prior knowledge on the relevant features in the dataset. The correlations or mutual information between these features and the activations at the final layer can then be studied to detect any sign of neglect. If problems involving neglect are discovered it may be beneficial to add extra terms to the cost function which encourage more mutual information between these neglected features and the labels thereby overcoming the noise barrier and pushing the training dynamics away from such neglect minimum.

ACKNOWLEDGMENTS

Rodrigo Andrade de Bem is a CAPES Foundation scholarship holder (Process no: 99999.013296/2013-02, Ministry of Education, Brazil).

REFERENCES

Abramowitz, Milton. *Handbook of Mathematical Functions, With Formulas, Graphs, and Mathematical Tables,*. Dover Publications, Incorporated, 1974. ISBN 0486612724.

Andriluka, Mykhaylo, Pishchulin, Leonid, Gehler, Peter, and Schiele, Bernt. 2d human pose estimation: New benchmark and state of the art analysis. In *CVPR*, 2014.

Biggs, Norman, Lloyd, E Keith, and Wilson, Robin J. *Graph Theory, 1736-1936*. Oxford University Press, 1976.

Bronstein, Michael M, Bruna, Joan, LeCun, Yann, Szlam, Arthur, and Vandergheynst, Pierre. Geometric deep learning: going beyond euclidean data. *arXiv preprint arXiv:1611.08097*, 2016.

Bruna, Joan, Zaremba, Wojciech, Szlam, Arthur, and LeCun, Yann. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013.

Cardy, John. Conformal invariance and percolation. *arXiv preprint math-ph/0103018*, 2001.

Cardy, John and Ziff, Robert M. Exact results for the universal area distribution of clusters in percolation, ising, and potts models. *Journal of statistical physics*, 110(1):1–33, 2003.

Choset, Howie and Nagatani, Keiji. Topological simultaneous localization and mapping (slam): toward exact localization without explicit localization. *ICRA*, 2001.

Daniely, Amit, Frostig, Roy, and Singer, Yoram. Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity. In *NIPS*, 2016.

Dauphin, Yann N, Pascanu, Razvan, Gulcehre, Caglar, Cho, Kyunghyun, Ganguli, Surya, and Bengio, Yoshua. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In *NIPS*, 2014.

Dokholyan, Nikolay V, Buldyrev, Sergey V, Havlin, Shlomo, King, Peter R, Lee, Youngki, and Stanley, H.Eugene. Distribution of shortest paths in percolation. *Physica A: Statistical Mechanics and its Applications*, 266(1 - 4):55 – 61, 1999. ISSN 0378-4371.

Elfes, Alberto. Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6): 46–57, 1989.

Ghahramani, Zoubin. Probabilistic machine learning and artificial intelligence. *Nature*, 521(7553): 452–459, 2015.

Graves, Alex, Wayne, Greg, and Danihelka, Ivo. Neural turing machines. *CoRR*, abs/1410.5401, 2014.

Graves, Alex, Wayne, Greg, Reynolds, Malcolm, Harley, Tim, Danihelka, Ivo, Grabska-Barwińska, Agnieszka, Colmenarejo, Sergio Gómez, Grefenstette, Edward, Ramalho, Tiago, Agapiou, John, et al. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538 (7626):471–476, 2016.

Henaff, Mikael, Bruna, Joan, and LeCun, Yann. Deep convolutional networks on graph-structured data. *arXiv preprint arXiv:1506.05163*, 2015.

Jain, Ashesh, Zamir, Amir R, Savarese, Silvio, and Saxena, Ashutosh. Structural-rnn: Deep learning on spatio-temporal graphs. In *CVPR*, 2016.

Johnson, Daniel D. Learning graphical state transitions. *ICLR*, 2017.

Johri, S and Bhatt, RN. Singular behavior of eigenstates in anderson's model of localization. *Physical review letters*, 109(7):076402, 2012.

Kramer, Bernhard and MacKinnon, Angus. Localization: theory and experiment. *Reports on Progress in Physics*, 56(12):1469, 1993.

Langlands, Robert, Pouliot, Philippe, and Saint-Aubin, Yvan. Conformal invariance in two-dimensional percolation. *arXiv preprint math/9401222*, 1994.

LeCun, Yann, Bengio, Yoshua, and Hinton, Geoffrey. Deep learning. *Nature*, 521(7553):436–444, 2015.

Li, Yujia, Tarlow, Daniel, Brockschmidt, Marc, and Zemel, Richard. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493*, 2015.

Masci, Jonathan, Boscaini, Davide, Bronstein, Michael, and Vandergheynst, Pierre. Geodesic convolutional neural networks on riemannian manifolds. In *ICCV Workshop*, 2015a.

Masci, Jonathan, Boscaini, Davide, Bronstein, Michael, and Vandergheynst, Pierre. Shapenet: Convolutional neural networks on non-euclidean manifolds. Technical report, 2015b.

Masucci, Adolfo Paolo, Smith, D, Crooks, A, and Batty, Michael. Random planar graphs and the london street network. *The European Physical Journal B-Condensed Matter and Complex Systems*, 71(2):259–271, 2009.

Mishkin, Dmytro and Matas, Jiri. All you need is a good init. *CoRR*, abs/1511.06422, 2015.

Nguyen, Anh, Yosinski, Jason, and Clune, Jeff. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *CVPR*, 2015.

Oh, Junhyuk, Singh, Satinder, and Lee, Honglak. Value prediction network. In *NIPS*, 2017.

Perozzi, Bryan, Al-Rfou, Rami, and Skiena, Steven. Deepwalk: Online learning of social representations. In *ACM SIGKDD*, 2014.

Scarselli, Franco, Gori, Marco, Tsoi, Ah Chung, Hagenbuchner, Markus, and Monfardini, Gabriele. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2009.

Schmidt, Michael and Lipson, Hod. Distilling free-form natural laws from experimental data. *science*, 324(5923):81–85, 2009.

Schrijver, Alexander. On the history of the shortest path problem. *Doc. Math*, 155, 2012.

Silver, David, van Hasselt, Hado, Hessel, Matteo, Schaul, Tom, Guez, Arthur, Harley, Tim, Dulac-Arnold, Gabriel, Reichert, David P., Rabinowitz, Neil C., Barreto, André, and Degris, Thomas. The predictron: End-to-end learning and planning. In *ICML*, 2017.

Smirnov, Stanislav and Werner, Wendelin. Critical exponents for two-dimensional percolation. *arXiv preprint math/0109120*, 2001.

Sukhbaatar, Sainbayar, Szlam, Arthur, Weston, Jason, and Fergus, Rob. End-to-end memory networks. In *NIPS*, pp. 2440–2448, 2015.

Swirszcz, Grzegorz, Czarnecki, Wojciech Marian, and Pascanu, Razvan. Local minima in training of deep networks. *arXiv preprint arXiv:1611.06310*, 2016.

Tamar, Aviv, Wu, Yi, Thomas, Garrett, Levine, Sergey, and Abbeel, Pieter. Value iteration networks. In *NIPS*, 2016.

Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, 2016.

Viana, Matheus P, Bordin, Patricia, Barthelemy, Marc, and Strano, Emanuele. The simplicity of planar networks. *Nature Scientific Reports*, 3(arXiv: 1312.3788):3495, 2013.

Weston, Jason, Chopra, Sumit, and Bordes, Antoine. Memory networks. *CoRR*, abs/1410.3916, 2014.

Weston, Jason, Bordes, Antoine, Chopra, Sumit, and Mikolov, Tomas. Towards ai-complete question answering: A set of prerequisite toy tasks. *CoRR*, abs/1502.05698, 2015.

Yu, Jeffrey Xu and Cheng, Jiefeng. Graph reachability queries: A survey. In *Managing and Mining Graph Data*, pp. 181–215. Springer, 2010.

Zhang, Chiyuan, Bengio, Samy, Hardt, Moritz, Recht, Benjamin, and Vinyals, Oriol. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Zhou, Zongzheng, Yang, Ji, Deng, Youjin, and Ziff, Robert M. Shortest-path fractal dimension for percolation in two and three dimensions. *Physical Review E*, 86(6):061101, 2012.

A VISUALIZATION OF THE OPTIMAL-BFS MINIMUM

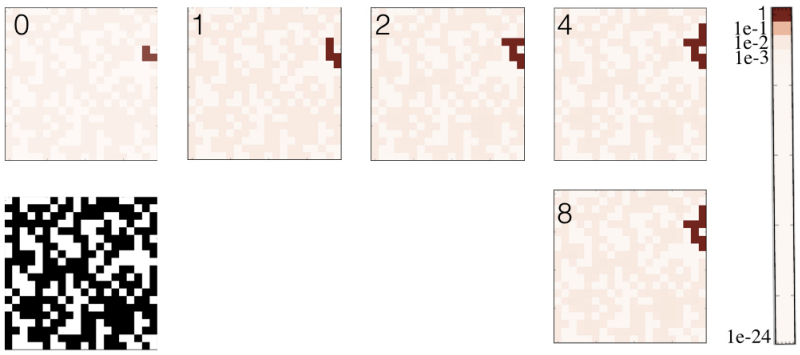


Figure 6: A numerical experiment showing how our maze classification architecture processes a particular sample consisting a maze (black and white image) and a hot-spot image marking the starting-point (panel (0)) when its weights are tuned to the optimal BFS solution. The first layer receives a hot-spot image which is *On* only near the starting-point of the maze H_0) (panel (0)). This *On* activation then spreads on the black cluster containing the start-point in (H_n with $n = 1, 2, 4, 8$, panels 1,2,4,8 resp.). Notably other region are *Off* (i.e. smaller than v_l) but they are not zero as shown by the faint imprint of the maze on H_n.

B TEST ERROR AWAY FROM THE OPTIMAL-BFS SOLUTION

We have implemented the architecture described in the main text using Theano (Theano Development Team (2016)) and tested how $cost$ changes as a function of $\delta = \lambda_c - \lambda$ ($\lambda_c = 9.727..$) for mazes of sizes $L = 24, 36$ and depth (number of layers) 128. These depths are enough to keep the error rate negligible at $\delta = 0$. A slight change made compared to Maze-testing as described in the main text, is that the hot-spot was fixed at a distance $L/2$ for all mazes. The size of the datasets was between $1E + 5$ and $1E + 6$. We numerically obtained the normalized performance ($cost_L(\delta)$) as a function of L and δ.

As it follows from Eq. (4) in the main text the curve, $\log(L^{-2+5/24} cost_L(\delta))$, for the $L = 24$ and $L = 36$ results should collapse on each other for $\rho_{bug} < L^{-d_f}$. Figure (5) of the main-test depicts three such curves, two for $L = 36$, to give an impression of statistical error, and one for $L = 24$ curve (green), along with the fit to the theory (dashed line). The fit, which involves two parameters (the proportionally constant in Eq. (4) of the main text and C) captures well the behavior over three orders of magnitude. As our results are only asymptotic, both in the sense of large L and $\lambda \to \lambda_c$, minor discrepancies are expected.

C LINEARIZATION OF THE SIGMOID-CONVOLUTIONAL NETWORK AROUND OFF ACTIVATION

To prepare the action of sigmoid-convolutional for linearization we find it useful to introduce the following variables on locations (r_b, r_w) with black (b) and white (w) cells

$$\psi_n(r_\alpha) = \mathrm{H}_n(r_\alpha) - a(r_\alpha) \tag{7}$$

$$a(r_b) = v_{l,c} \tag{8}$$

$$a(r_w) = e^{-5.5\lambda}. \tag{9}$$

Rewriting the action of the sigmoid-convolutional layer in terms of these we obtain

$$\psi_n(r_\alpha) + a(r_\alpha) = \sigma\left[\lambda\left(\psi_{n-1}(r_\alpha) + \sum_{\langle r',r\rangle}\psi_{n-1}(r') + 5\mathrm{I}(r_\alpha)\right) - 5.5\lambda + \lambda d(r_\alpha)\right], \quad (10)$$
$$d(r_\alpha) = a(r_\alpha) + \sum_{\langle r',r_\alpha\rangle} a(r')$$

where $\sum_{\langle r',r\rangle}$ means summing over the 4 sites neighboring r. Next we treating $\psi_n(r)$ as small and Taylor expand

$$\psi_n(r_w) = \lambda\frac{\sigma}{dx}|_{-5.5\lambda}\left(\psi_{n-1}(r_\alpha) + \sum_{\langle r',r_w\rangle}\psi_{n-1}(r') + d(r_w)\right) \quad (11)$$
$$\psi_n(r_b) = \lambda\frac{d\sigma}{dx}|_{\sigma^{-1}(v_{l,c})}\left(\psi_{n-1}(r_\alpha) + \sum_{\langle r',r_b\rangle}\psi_{n-1}(r') + (d(r_b) - 5v_{l,c})\right)$$

where $v_{l,c} \approx 0.02(0)$ is the low (and marginally stable) solution of the equation $v_{l,c} = \sigma(-0.5\lambda_c + 5v_{l,c})$.

Next in the consistency with our assumption that $|\psi_{n-1}(r)|$ is small, we can assume $|\psi_{n-1}(r)| < 1$, and obtain that $\psi_n(r_w) < e^{-5.5\lambda}(5 + e^{-0.5\lambda})$ and therefore, since we are working near $\lambda = 9.727..$ it is negligible. The equation of $\psi_n(r_b)$ now appears as

$$\psi_n(r_b) = \tilde{\lambda}\left(\psi_{n-1}(r_\alpha) + \sum_{\langle r'_b,r_b\rangle}\psi_{n-1}(r') + d(r_b) - 5v_{l,c})\right) + O(\lambda\frac{d^2\sigma}{dx^2}\psi^2) \quad (12)$$

where the summation of neighbor now includes only black cells and $\tilde{\lambda} = \lambda\frac{d\sigma}{dx}|_{\sigma^{-1}(v_{l,c})}$. Due to form the sigmoid function, $\lambda\frac{d^2\sigma}{dx^2}|_{\sigma^{-1}[\epsilon_\infty]}$ is of the same magnitude as $\tilde{\lambda}$, and consequently the relative smallness of this terms is guaranteed as long as $\psi_n \ll 1$.

We thus obtained a linearized version for the sigmoid-convolutional network which is suitable for stability analysis. Packing $\psi_n(r_b)$ and $d(r_b) - 5v_{l,c}$ into vectors ($\vec{\psi}_n$,$d(\vec{r}_b)$) the equation we obtained can be written as

$$\vec{\psi}_n = S\vec{\psi}_{n-1} + \vec{d} \quad (13)$$

with S being a symmetric matrix. Denoting by $\vec{\phi}_n^T$ and s_n the left eigenvectors and eigenvalues of S, we multiply the above equation from the left with $\vec{\phi}_n^T$ and obtain

$$c_n = s_n c_{n-1} + \vec{\phi}_n^T\vec{d} \quad (14)$$
$$\vec{\psi}_n = \sum_n c_n\vec{\phi}_n.$$

Stability analysis on this last equation is straightforward: For $|s_n| < 1$, a stable solution exists given by $c_n = \frac{\vec{\phi}_n^T\vec{d}}{(1-s_n)}$. Furthermore as the matrix S has strong disorder, $\vec{\phi}_n$ are localized in space. Consequently $\vec{\phi}_n^T\vec{d}$ is of the same order of magnitude as $\vec{d} \approx e^{-0.5\lambda} \approx 0.01$ and as long as $s_n < 0.9$, these stable solutions are well within the linear approximation we have carried. For $|s_n| > 1$, there are no stable solutions.

There is an important qualitative lesson to be learned from applying these results on an important test case: A maze with only black cells. In this case it is easy to verify directly on the non-linear sigmoid-convolutional map that a uniform solution becomes unstable exactly at $\lambda = \lambda_c$. Would we find the same result within our linear approximation?

To answer the above, first note that the maximal eigenvalue of S will be uniform with $s_{max} = 5\tilde{\lambda}$. Furthermore for an all black maze $\vec{d}$ would be exactly zero and the linear equation becomes

homogeneous. Consequently destabilization occurs exactly at $\tilde{\lambda} = 1/5$ and is not blurred by the inhomogeneous terms. Recall that λ_c is defined as the value at which the two lower solutions of $x = \sigma[-0.5\lambda_c + 5\lambda_c x]$ and it also satisfies the equation $v_{l,c} = \sigma[-0.5\lambda_c + 5\lambda_c v_{l,c}]$. Taking a derivative of the former and putting $x = v_{l,c}$ one finds that $1 = 5\lambda_c \frac{d\sigma[-0.5\lambda_c + 5v_{l,c}]}{dx}$. It is now easy to verify that even within the linear approximation destabilization occurs exactly at λ_c. The source of this agreement is the fact that $\vec{d}$ vanishes for a uniform black maze.

The qualitative lesson here is thus the following: The eigenvectors of S with large s, are associated with large black regions in the maze. It is only on the boundaries of such regions that $\vec{d}$ is non-zero. Consequently near $\lambda \approx \lambda_c$ the $\vec{d}$ term projected on the largest eigenvalues can, to a good accuracy, be ignored and stability analysis can be carried on the homogeneous equation $\vec{\psi} = S\vec{\psi}$ where $s_n < 1$ means stability and $s_n > 1$ implies a bug.

D LOG-LIKELIHOOD AND NOISY PREDICTIONS

Consider an abstract classification tasks where data point $x \in X$ are classified into two categories $l \in \{0, 1\}$ using a deterministic function $f : X \to \{0, 1\}$ and further assume for simplicity that the chance of $f(x) = a$ is equal to $f(x) = b$. Phrased as a conditional probability distribution $P_f(l|x)$ is given by $P_f(f(a)|x) = 1$ while $P_f(!f(a)|x) = 0$. Next we wish to compare the following family of approximations to P_f

$$P_{\alpha',\beta'}(l|x) = 1/2 + \alpha'(2l-1)(2f(x)-1) + \beta'(2l-1)(2g(x)-1) \tag{15}$$

where $g|X \to \{0, 1\}$ is a random function, uncorrelated with $f(x)$, outputting the labels $\{0, 1\}$ with equal probability. Notably at $\alpha' = 1/2, \beta' = 0$ it yields P_f while at $\alpha', \beta' = 0$ it is simply the maximum entropy distribution.

Let us measure the log-likelihood of $P_{\alpha',\beta'}$ under P_f for $\alpha', \beta' \ll 1$

$$\begin{aligned}
\mathcal{L}(\alpha', \beta') &= \sum_{(x,l)} P_f(x, l) \log\left(1/2 + \alpha'(2l-1)(2f(x)-1) + \beta'(2l-1)(2g(x)-1)\right) \qquad (16)\\
&\approx \sum_{(x,l)} P_f(x, l) \log(1/2) + 2\left[\alpha'(2l-1)(2f(x)-1) + \beta'(2l-1)(2g(x)-1)\right]\\
&- 2\left[\alpha'(2l-1)(2f(x)-1) + \beta'(2l-1)(2g(x)-1)\right]^2\\
&= \log(1/2) + 2\alpha' - 2\alpha'^2 - 2\beta'^2
\end{aligned}$$

We thus find that β' reduces the log-likelihood in what can be viewed as a penalty to false confidence or noise. Assuming, as argued in the main text, that α' is constrained to be smaller than β'^2 near $\beta' \approx 0$, it is preferable to take both α' and β' to zero and reach the maximal entropy distribution. We note by passing that the same arguments could be easily generalized to $f(x), g(x)$ taking real values leading again to an $O(\alpha) - O(\beta^2)$ dependence in the cost function.

Let us relate the above notations to the ones in the main text. Clearly $x = (\{\mathrm{I}\}, \mathrm{H}_0)$ and $\{0, 1\} = \{Unsolvable, Solvable\}$. Next we recall that in the main text α and β multiplied the vectors function representing the H_0-depended and H_0-independent parts of H_n. The probability estimated by the logistic regression module was given by

$$\begin{aligned}
P(Solvable|x) &= \frac{e^{\vec{K}_{Solvable}\cdot\vec{\mathrm{H}}_n}}{e^{-\vec{K}_{Solvable}\cdot\vec{\mathrm{H}}_n} + e^{-\vec{K}_{Unsolvable}\cdot\vec{\mathrm{H}}_n}} \qquad (17)\\
P(Unsolvable|x) &= \frac{e^{\vec{K}_{Unsolvable}\cdot\vec{\mathrm{H}}_n}}{e^{-\vec{K}_{Solvable}\cdot\vec{\mathrm{H}}_n} + e^{-\vec{K}_{Unsolvable}\cdot\vec{\mathrm{H}}_n}}
\end{aligned}$$

which yields, to leading order in α and β

$$P_{\alpha,\beta}(l|x) = 1/2 + \alpha(2l+1)\vec{K}_l^- \cdot A + \beta(2l+1)\vec{K}_l^- \cdot B \tag{18}$$

where $\vec{K}^- = (\vec{K}_{Solvable} - \vec{K}_{Unsolvable})/2$ and $(2l+1)$ understood as the taking the values ± 1. Consequently $(2f-1)$ and $(2g-1)$ are naturally identified with $\vec{K}_{Solvable} \cdot A/N_A$ and $\vec{K}_{Solvable} \cdot B/N_B$ respectively with N_A and N_B being normalization constants ensuring a variance of 1. While $(\alpha', \beta') = (N_A\alpha, N_B\beta)$. Recall also that by construction of the dataset, the g we thus obtain is uncorrelated with f.

E NUMERICAL SUPPORT FOR THE $\alpha \ll \beta^2$ CONJECTURE

Here we provide numerical evidence showing that $\alpha \ll \beta^2$ in a large region around the total neglect minima found during the training of our architecture on the biased dataset (i.e. the one where marginalizing over the starting-point yields a 50/50 chance of being solvable regardless of the maze shape).

For a given set of K_hot, K and b parameters we fix the maze shape and study the variance of the top layer activations given $O(100)$ different starting points. We pick the maximal of these and then average this maximal variance over $O(100)$ different mazes. This yields our estimate of α. In fact it is an upper bound on α as this averaged-max-variance may reflect wrong prediction provided that they depend on H_0.

We then obtain an estimate of β by again calculating the average-max-variance of the top layer however now with $\mathrm{H}_0 = 0$ for all maze shapes.

Next we chose a 100 random paths parametrized by γ leading from the total neglect minima ($\gamma = 0$) for the total neglect through a random point at $\gamma = 15$, and then to the checkerboard-BFS minima at $\gamma = 30$. The random point was placed within a hyper-cube of length 4 having the total neglect minima at its center. The path was a simple quadratic interpolation between the three point. The graph below shows the statistics of α/β^2 on these 100 different paths. Notably no path even had $\alpha > e^{-30}\beta^2$ within the hyper-cube. We have tried three different other lengths for the hyper cube (12 and 1) and arrived at the same conclusions.

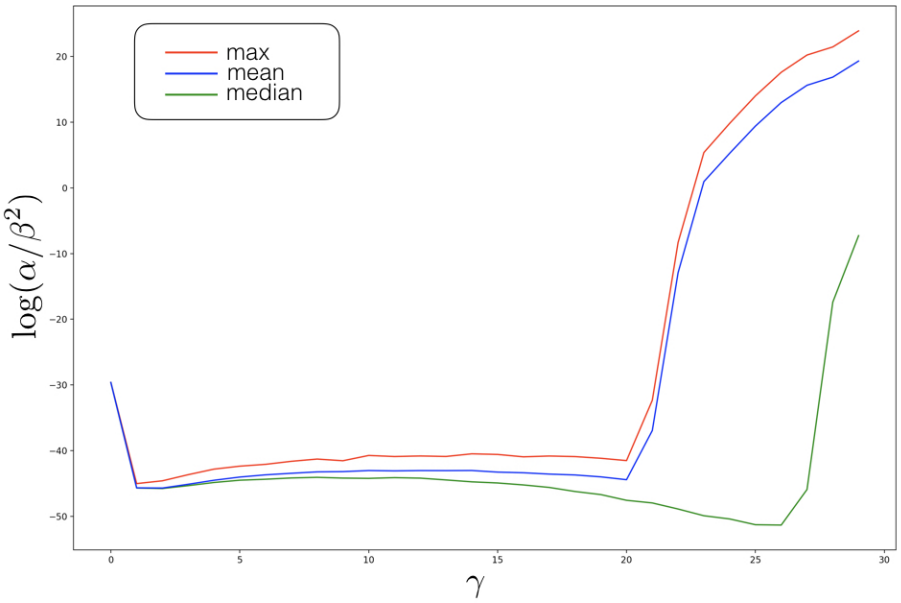


Figure 7: The natural logarithm of an upper bound to α/β^2 as a function of a paramterization (γ) of a path leading from the numerically obtained total neglect minima to the checkerboard BFS minima through a random point. The three different curves show the max,mean, and median based on a 100 different paths. Notably no path violated the $\alpha \ll \beta^2$ constrain in the vicinity of the total neglect minima.

B

3D Hand Shape and Pose from Images in the Wild

3D Hand Shape and Pose from Images in the Wild

Adnane Boukhayma[1], Rodrigo de Bem[1,2], Philip H.S. Torr[1]

[1] University of Oxford, UK
[2] Federal University of Rio Grande, Brazil
{adnane.boukhayma, rodrigo.andradedebem, philip.torr}@eng.ox.ac.uk

Abstract

We present in this work the first end-to-end deep learning based method that predicts both 3D hand shape and pose from RGB images in the wild. Our network consists of the concatenation of a deep convolutional encoder, and a fixed model-based decoder. Given an input image, and optionally 2D joint detections obtained from an independent CNN, the encoder predicts a set of hand and view parameters. The decoder has two components: A pre-computed articulated mesh deformation hand model that generates a 3D mesh from the hand parameters, and a re-projection module controlled by the view parameters that projects the generated hand into the image domain. We show that using the shape and pose prior knowledge encoded in the hand model within a deep learning framework yields state-of-the-art performance in 3D pose prediction from images on standard benchmarks, and produces geometrically valid and plausible 3D reconstructions. Additionally, we show that training with weak supervision in the form of 2D joint annotations on datasets of images in the wild, in conjunction with full supervision in the form of 3D joint annotations on limited available datasets allows for good generalization to 3D shape and pose predictions on images in the wild.

1. Introduction

Human hand pose estimation and reconstruction in 3D is a long standing problem in the computer vision and graphics communities that has applications in various domains such as virtual and augmented reality and human-machine interaction [35, 15, 46, 13]. With the abundance of affordable commodity depth cameras, the research literature focused naturally more on estimating 3D hand pose through depth observations (e.g. [62, 66, 10, 36, 61]), and many works also explored this problem in multi-view setups [33, 65, 41, 8, 31, 50]. When it comes to a monocular color input, the problem becomes inherently ill posed due to the increased depth and scale ambiguities, but that did not prevent several researchers [4, 9, 51, 57, 63, 39] from attempting to solve it in the past albeit with limited results. More recently, the unprecedented success of deep learning on similar tasks motivated new work with encouraging results for 3D hand pose from single images [68, 27, 7, 47, 14]. Nevertheless, this task remains particularly difficult: Unlike clothed human bodies or faces, hands have an almost uniform appearance and lack characteristic local features such as eyes and mouths in faces. Unlike bodies, they can have more complex pose configurations and they can be captured from a much wider range of views. Furthermore when observed in the wild, as in dataset MPII+NZSL [44] (Figure 9), their images usually contain external occlusion, self-occlusion, clutter and blur due to their motion. Besides, hands are often small in size compared to the scene so cropped patches around them have low resolutions.

The main obstacles for 3D hand pose estimation from images with deep learning include: (i) The lack of large datasets annotated with reliable 3D ground-truth and (ii) the incapability of the current 3D annotated datasets to make networks generalize greatly to challenging images in the wild.

The first point is tackled by the literature through training with synthetic images [68], populating datasets by transforming synthetic images into real looking ones [27], or leveraging auxiliary types of data in training like depth [7, 47]. We propose a different and simple yet efficient approach to alleviate both challenges (i) and (ii) by circumventing heavy dependence of 3D data in training: Instead of relying on images paired with 3D joint annotations to learn a prior on hand geometry, we exploit a recently proposed differentiable articulated mesh deformation hand model [40] built with linear blend skinning [18], and we reformulate the prediction problem into a learning-based model fitting, that can be trained using both 3D and 2D joint annotations. Training with 2D annotated images al-

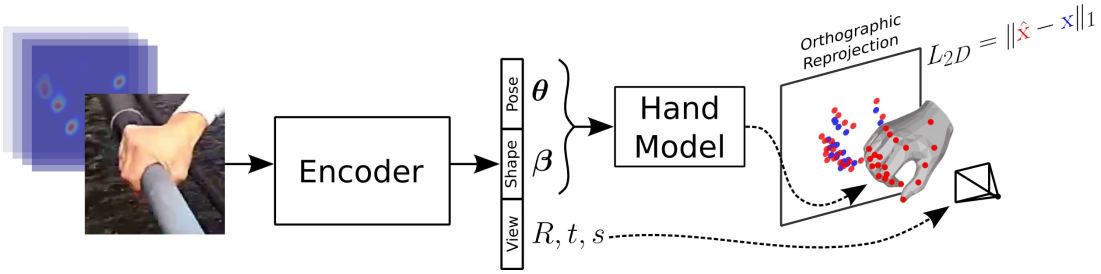


Figure 1: Our pipeline takes as input a hand image and optionally 2D joint heat-maps from an independent CNN. The encoder generates the shape, pose and view parameters. The hand parameters are fed to the hand model that generates a triangulated 3D mesh and its underlying 3D skeleton. The latter are re-projected into the image domain using a weak perspective camera model controlled by the view parameters. This network is trained end-to-end with a combination of weak 2D and full 3D joint supervision. The hand and view parameters are not supervised.

lows access to larger datasets (e.g. PANOPTIC [44]) with a fair share of annotated images in the wild (e.g. MPII+NZSL [44]) compared to datasets with 3D ground-truth, thus helping improve generalization to this type of challenging data. Given an input image, and optionally 2D joint detections obtained from an independent CNN, a deep convolutional encoder predicts the hand shape and pose parameters and view parameters. The model-based decoder uses the latter to generate a 3D triangulated hand mesh and its underlying skeleton, along with their re-projection in image domain (see Figure 1).

Our contributions in this paper are as follows: This work is the first to propose end-to-end learning of both 3D hand shape and pose from a single RGB image. We also show for the first time that the prior knowledge of factored hand shape and pose in a pre-computed linear blend skinning [18] hand model [40] combined with a deep-convolutional encoder yields state-of-the-art performance in 3D pose prediction from images, and produces geometrically valid and plausible 3D reconstructions, without the need for post-processing optimizations [27]. We show that this strategy combined with training on 2D annotated datasets of images in the wild produces good generalization in 3D hand reconstruction for challenging images in uncontrolled environments.

We evaluate our work both quantitatively in terms of 3D pose estimation and qualitatively using various public datasets. These evaluation sets account for cases of hand interaction with objects, occlusion and clutter, and contain egocentric view images, third person view images, and images in the wild. Our method obtains state of-the-art results on standard benchmarks, even compared to methods using additional depth information in training [7, 47], camera intrinsics [27, 34], and post-processing optimization [27]. Our method shows superior qualitative results on a challenging dataset of images in the wild (Figure 9 & supplementary material).

2. Related work

There is a rich literature on 3D hand pose and reconstruction from depth [62, 66, 10, 36, 61, 11, 43, 45, 19, 20, 24, 30, 37, 48, 52, 53, 59, 64], image and depth [26, 32, 49, 28], stereo [33, 65, 41] and multiple images [8, 31, 50]. We focus hereby on research material that solely considers a single color input image.

3D hand pose from a single image

Pre-deep learning There have been attempts to solve 3D hand pose estimation from a monocular color input prior to deep learning with both discriminative and generative approaches [4, 9, 51, 57, 63, 39]. However, most of these methods have limited performance and depend on various requirements such as careful initialization and prior knowledge of the background.

Deep learning The work of [68] was the first to propose 3D hand pose estimation from single images using deep learning. Their method consists of the concatenation of three networks that segment the hand, predict 2D joints, and then predict 3D joints subsequently. The work of [27] shows that the previous method generalizes poorly to real world images since a major part of their training data is synthetic. In turn, they ([27]) propose to use Cycle-GAN [67] to transform synthetic 3D annotated images of hands into real looking ones. The resulting data is used to train a regressor to predict 2D and 3D hand joints. A final optimization step fits a 3D skeleton to the former 2D and 3D predictions using the camera intrinsics. The method in [34] consists in an optimization that fits a hand model to 2D joint detections obtained from a state-of-the-art CNN [44]. We also use a pre-defined hand model [40] but within a pipeline trained end-to-end.

Depth regularization Recent works tackle depth ambiguity in 3D hand pose prediction from images by leveraging depth maps in training. [7] proposes to reduce the

dependency on noisy 3D annotations in real datasets by introducing a network that predicts full depth maps from the 3D joints. This depth regularizer is trained with ground-truth depth data for both real and synthetic training images, while the 3D predictions are only supervised by the reliable synthetic labels. The authors in [47] use multiple variational auto-encoders sharing the same latent space each auto-encoding a separate hand data modality (e.g. images, 2D joints, 3D joints). They show that the auxiliary auto-encoders help regularize the latent space and produce improved cross-modal predictions (e.g. image to 3D joints). [14] shows that predicting an implicit 2.5D heat-map representation yields improved 3D predictions even without explicit full depth-map supervision.

Hand models Many hand models have been proposed in the literature primarily aiming at tracking depth and color data, where the hand is modelled using various techniques such as assembled geometric primitives [32], sum of Gaussians [50], sphere meshes [58] or loop subdivision of a control mesh [20]. In order to better capture the shape of the hand, [32] defines scaling terms to allow bone length to vary, while [54] pre-calibrates the shape to fit the hand of interest. The work in [20] was the first to learn hand shape variation from scans with linear blend skinning [18]. The model proposed recently in [40] and referred to as MANO improves on the latter by learning pose dependent corrective blend shapes [25], thus modelling both hand shape and pose and generating more realistic posed meshes. We use the MANO [40] model in this work.

Model-based decoders Several works propose to combine deep convolutional encoders with generative models as decoders for human face [56, 55] and body [17, 60] 3D reconstruction. In many of these works, the decoder is a combination of a parametric model (e.g. linear face model [6], SMPL [25]) and a re-projection/rendering module. While most works fix these decoders, some propose to tune them after a supervised initialization [2, 22, 55]. This is the first work to propose a combination of a CNN encoder with a fixed generative hand model [40] for the problem of 3D hand reconstruction from images.

3. Overview

As illustrated in Figure 1, our pipeline takes as input an image of a hand and optionally 2D joint heat-maps from an independent hand detector. A deep convolutional encoder processes the input and generates a set of hand shape $\boldsymbol{\beta}$ and pose $\boldsymbol{\theta}$ parameters, and a set of view parameters R, t and s. The hand parameters are fed to a differentiable articulated mesh deformation hand model that generates a triangulated 3D mesh and its underlying 3D skeleton. These outputs are then re-projected into the image domain through a weak perspective camera model controlled by the view parameters. The re-projection module and the hand model together form a model-based decoder whose parameters are fixed and do not require training. The encoder is pre-trained with synthetic examples that we created as elaborated in Section 6. We note that the training of our pipeline is done end-to-end using 2D and 3D joint annotations without supervision on the hand and view parameters, except for a regularization on the hand parameters to ensure their magnitude is small. We detail and explain the functioning of the various parts of the pipeline in the following.

4. Hand model

We use the MANO hand model [40] which is based on the SMPL model for human bodies [25]. It is an articulated mesh deformation model represented with a differentiable function $M(\boldsymbol{\beta}, \boldsymbol{\theta})$ taking as input two sets of parameters $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ that control the shape and pose of the generated hand respectively:

$$M(\boldsymbol{\beta}, \boldsymbol{\theta}) = W(T(\boldsymbol{\beta}, \boldsymbol{\theta}), J(\boldsymbol{\beta}), \boldsymbol{\theta}, \mathcal{W}), \tag{1}$$

where W is a linear blend skinning [18] function applied to a template hand triangulated mesh T rigged with a kinematic tree of $K = 16$ joints. J represents the joint locations and it is learned as a sparse linear regressor from mesh vertices, and $\mathcal{W}$ are the blend weights.

In order to reduce the artifacts of linear blend skinning such as overly smooth outputs and mesh collapse around joints, the hand template T is obtained by deforming a mean mesh $\bar{\mathbf{T}}$ with both shape and pose corrective blend shapes, $\mathbf{S}_n$ and $\mathbf{P}_n$ respectively, as follows:

$$T(\boldsymbol{\beta}, \boldsymbol{\theta}) = \bar{\mathbf{T}} + \sum_{n=1}^{|\boldsymbol{\beta}|} \beta_n \mathbf{S}_n + \sum_{n=1}^{9K} (R_n(\boldsymbol{\theta}) - R_n(\boldsymbol{\theta}^*)) \mathbf{P}_n, \tag{2}$$

where $R_n(\boldsymbol{\theta})$ is the n^{th} element of a vector concatenating rotation matrix coefficients from all joints for pose $\boldsymbol{\theta}$ and $\boldsymbol{\theta}^*$ is the rest pose. The model constants $\{\bar{\mathbf{T}}, \mathbf{S}, \mathbf{P}, J, \mathcal{W}\}$ are learned using registered hand scans from 31 subjects performing roughly 51 hand poses.

In the SMPL model, the pose vector $\boldsymbol{\theta}$ stacks the angle-axis values of the joints. To help the hand model generate physically plausible poses, the authors in [40] reduce this pose representation to a linear embedding by performing Principal Component Analysis on angle-axis values of the joints in the data collected to build the model. The pose vector $\boldsymbol{\theta}$ contains the resulting main coefficients from PCA instead of the angle-axis values. 10 coefficients are retained for the pose ($\boldsymbol{\theta} \in \mathbb{R}^{10}$), and 10 coefficients are used to represent the shape as well ($\boldsymbol{\beta} \in \mathbb{R}^{10}$).

Given input shape and pose parameters, we obtain a hand mesh $M(\boldsymbol{\beta}, \boldsymbol{\theta})$ of $N = 778$ vertices and 1538 faces, along with the corresponding 3D joints $J(\boldsymbol{\beta}, \boldsymbol{\theta}) = R_{\boldsymbol{\theta}}(J(\boldsymbol{\beta}))$

where $R_{\boldsymbol{\theta}}$ is the global rigid transformation induced by pose $\boldsymbol{\theta}$. As the hand skeleton in MANO does not contain finger tip joints, we append J with 5 vertices from the hand mesh that correspond to these key-points. The final 3D joint output $J(\boldsymbol{\beta}, \boldsymbol{\theta})$ counts 21 key-points.

5. Camera model

In order to re-project the 3D hand mesh vertices $M(\boldsymbol{\beta}, \boldsymbol{\theta})$ and 3D joints $J(\boldsymbol{\beta}, \boldsymbol{\theta})$ into the 2D image plane, we use the weak perspective model. This approximation allows us to train with annotated images even in the absence of camera intrinsics, which is the case of images in the wild obtained from Youtube videos for instance (e.g. dataset MPII+NZSL). Given a global rotation matrix $R \in SO(3)$, a translation $t \in \mathbb{R}^2$ and a scaling $s \in \mathbb{R}^+$, the projection writes:

$$\hat{\mathrm{x}} = s\Pi(RJ(\boldsymbol{\beta}, \boldsymbol{\theta})) + t, \tag{3}$$

$$\hat{\mathrm{y}} = s\Pi(RM(\boldsymbol{\beta}, \boldsymbol{\theta})) + t, \tag{4}$$

where Π is the orthographic projection.

6. Encoder

Given an input hand image, the goal of the encoder is to predict the corresponding hand pose and shape parameters $\{\boldsymbol{\beta}, \boldsymbol{\theta}\}$ and camera parameters $\{R, t, s\}$. We use the ResNet-50 network [12] and we adjust the final fully connected layer to output a vector $v = \{R, t, s, \boldsymbol{\beta}, \boldsymbol{\theta}\} \in \mathbb{R}^{26}$. We note that global rotation R is encoded with axis-angle values and is hence represented with 3 parameters. We also experiment with feeding 2D hand joint heat-maps obtained with a state of the art method [44] as additional channel input to the hand RGB image.

Figure 2: Examples from our synthetic dataset created to pre-train the encoder.

Network pre-training We pre-train the encoder to ensure that the camera and hand parameters converge towards acceptable values. For this purpose, we create a synthetic dataset of paired hand images with their ground-truth camera and hand parameters using the same generative model that we use as a decoder. Hand geometries are obtained by sampling poses $\boldsymbol{\theta} \in [-2, 2]^{10}$ and shapes $\boldsymbol{\beta} \in [-0.03, 0.03]^{10}$ then applying rotations R, translations t and scalings s. Although the work of [40] does not model hand appearance, the authors provide the scans used to build the geometry model with their registered counterparts. The original scans come with 3D coordinates and RGB values for each vertex. We create example hand appearances using the registered scan topology: To each vertex in a registered mesh, we assign the RGB value of the closest vertex in the original corresponding scan, and we interpolate these values inside faces. The textured hands are finally rendered on top of random background images. Figure 2 shows examples from the resulting dataset.

7. Training objective

We combine multiple losses to train our pipeline: A 2D joint re-projection loss L_{2D}, a 3D joint loss L_{3D}, a hand mask loss L_{mask} and a model parameter regularization loss L_{reg}.

$$L = L_{2D} + \alpha_{3D} L_{3D} + \alpha_{mask} L_{mask} + \alpha_{reg} L_{reg}, \tag{5}$$

where $\alpha_{3D} = 10^2$, $\alpha_{mask} = 10^2$ and $\alpha_{reg} = 10^1$ are weighting factors.

2D joint re-projection loss This loss ensures that the re-projected hand joints in the image plane coincide with the ground-truth 2D hand joint annotations:

$$L_{2D} = \|\hat{\mathrm{x}} - \mathrm{x}\|_1, \tag{6}$$

where x is a vector containing the ground-truth 2D hand joint coordinates. We use the L_1 loss to account for inaccuracies in hand annotations in our training datasets.

3D joint loss When ground-truth 3D hand joint annotations are available (e.g STEREO dataset), this loss minimises the distance between the latter and the 3D hand joints generated by the hand model:

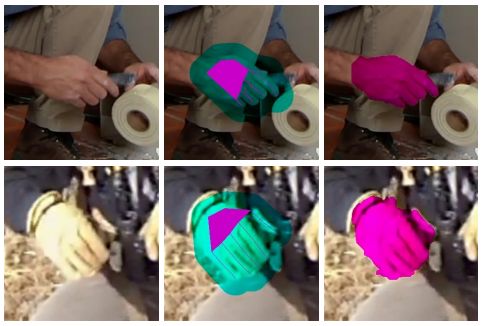
$$L_{3D} = \|RJ(\boldsymbol{\beta}, \boldsymbol{\theta}) - \mathrm{x}_{3D}\|_2^2, \tag{7}$$

where x_{3D} is a vector containing the ground-truth 3D hand joint coordinates.

Hand mask loss We introduce this novel loss to help speed up the convergence of our training and refine hand shape predictions. This loss penalizes re-projected hand vertices that lie outside of the hand region in a binary mask, which is pre-computed prior to training:

$$L_{mask} = 1 - \frac{1}{N} \sum_i H(\hat{\mathrm{y}}_i), \tag{8}$$

where H is an occlusion-aware hand mask, i.e $H(u) = 1$ if pixel u is inside the hand region even if the hand is occluded in the image, and $H(u) = 0$ otherwise. Notice that



(a) (b) (c)

Figure 3: GrabCut [42] hand segmentation initialized with 2D joint annotation. (a) Input image, (b) foreground, background and undecided regions from 2D joints, (c) final segmentation.

these masks cannot be obtained with hand skin segmentation methods (e.g.[23, 5]) as they are sensitive to occlusions.

We obtain an approximation of these masks (Figure 3) for training images using the GrabCut [42] algorithm, by initializing the foreground, background and probable foreground/background regions using the 2D hand joint annotations: As illustrated in Figure 3b, we create an initial foreground by drawing lines of 1 pixel width connecting joints according to the hand skeleton hierarchy. Pixels inside triangles formed by joints that belong anatomically to the hand surface are appended to the foreground as well. The undecided area is defined as the region within 70 pixels at most from the foreground, and the remaining pixels are assigned to the initial background.

Regularization loss This loss acts on the hand model parameters at the encoder output by reducing their magnitude for physically plausible hand reconstructions and reduced mesh distortions:

$$L_{reg} = \|\boldsymbol{\theta}\|_2^2 + \alpha_{\boldsymbol{\beta}}\|\boldsymbol{\beta}\|_2^2, \tag{9}$$

where $\alpha_{\boldsymbol{\beta}} = 10^4$ is a weighting factor.

8. Evaluation

We evaluate our method's 3D pose estimates quantitatively and its 3D reconstructions qualitatively on several datasets and with respect to state-of-the-art methods. Without access to camera intrinsics, and trained merely with 2D and 3D joint annotations, our method outperforms deep learning based competing methods, including those using additional depth information in training or camera intrinsics in evaluation. We show particularly superior 3D reconstructions on images in the wild that present challenging situations such as blur, low resolution, occlusion, extremely varying viewpoints and hand pose configurations.

Similar to [44], input images are assumed to be crops of fixed size around the hand. To achieve this, we use a hand key-point detector [44] to find the tightest rectangular box of edge size l containing the hand. Images are then cropped with a square patch of size $2.2l$ centred at the same 2D location as the previously detected box. The resulting cropped images are subsequently resized to have a width and height of 320. As done in [44], we use the right hand model and images of left hands are flipped horizontally.

Finally, we train our pipeline (Figure 1) using the Adam solver [21] with a learning rate of 10^{-4} and weight decay of 10^{-5}.

Datasets Our training set is made of dataset PANOPTIC [44] that counts 14847 images, the training set of MPII+NZSL [44] that counts 1912 images following the split in [44], and the training set of STEREO [65] that counts 15000 images following the split in [68]. This amounts to 31729 training images, 15000 (STEREO) with 3D joint annotations, and the remaining 16729 (PANOPTIC & MPII+NZSL) with 2D joint annotations only.

The PANOPTIC dataset [44] contains hands in various poses observed from multiple views in the Panoptic studio [16]. The MPII+NZSL dataset [44] is a combination of manually annotated images from The MPII Human Pose dataset [3] containing images from YouTube videos, and images from the New Zealand Sign Language (NZSL) Exercises of the Victoria University of Wellington [38]. The STEREO dataset [65] shows an actor's hand in third person view counting with the fingers and moving the hand randomly.

For evaluation, we use the DEXTER+OBJECT dataset [49] which shows interactions of an actor's hand with a cuboid object from a third person view. To evaluate robustness to occlusions and clutter, we use the EGODEXTER dataset [28] that displays a hand from an egocentric view interacting with various objects. We finally use the testing set of MPII+NZSL [44] to asses performance in the presence of blur, low resolution, varying viewpoints and hand pose configurations, among other characteristics of datasets of images in the wild.

Metrics To quantitatively evaluate 3D hand pose estimations, we report the percentage of correct points in 3D (3D PCK) and the average 3D Euclidean distance between the estimated 3D joints and the ground-truth when the latter is available, where distances are expressed in millimeters (mm). When only ground-truth 2D joint annotations are available (dataset MPII+NZSL), we report 2D PCK and the average 2D Euclidean distance between the estimated 2D re-projected joints and the ground-truth, where distances are expressed in pixels (px).

Comparison to competing methods We compare our results on the STEREO dataset to state-of-the-art methods in terms of 3D PCK in Figures 4 and 5, and we show 3D joint

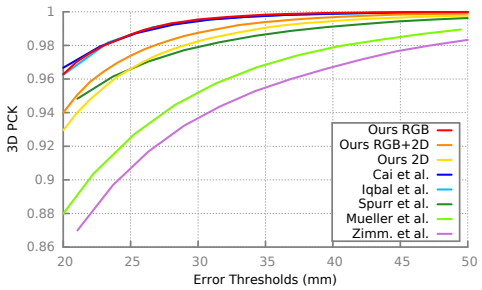


Figure 4: 3D PCK for STEREO.

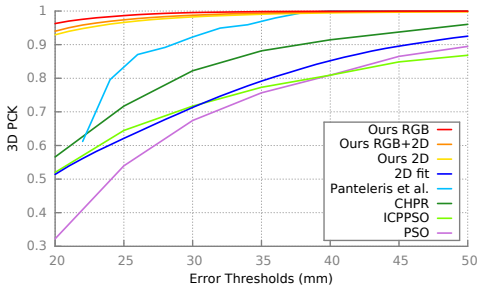


Figure 5: 3D PCK for STEREO.

	Ours RGB	Ours RGB+2D	Ours 2D	2D fit
3D distance	**9.76**	10.18	10.46	23.21

Table 1: Average 3D joint distance (mm) to ground-truth for STEREO.

errors in Table 1. Figure 4 shows deep learning based methods (Cai et al. [7], Iqbal et al. [14], Spurr et al. [47], Mueller et al. [27], Zimm. et al [68]) and Figure 5 shows methods that do not rely on deep learning (Panteleris et al. [34], PSO, ICPPSO, CHPR [65]). For this experiment, we add a keypoint at the center of the hand palm in the MANO model [40] as an interpolation of several mesh vertices to match the annotation of the STEREO dataset. We reproduce the evaluation protocol initially introduced in [68] by training on 10 sequences and testing on the remaining 2 and aligning predictions to the ground-truth hand root joint. Additionally, for a fair comparison to works [7, 47, 14], we crop the hand images for this experiment such that the final image size is 150% the size of the hand. Using RGB image input only, we obtain state-of-the results even though some of the competing methods use depth data in training ([7, 14]) in addition to images, while others ([27]) post-process their output with an optimization that fits their hand skeleton to their 3D and 2D joint predictions, and which uses the camera intrinsics as an additional input.

Figure 6 shows the performance of our method under occlusions and clutter with 3D PCK on the DEXTER+OBJECT dataset, and Table 2 shows 3D joint errors. Additionally,

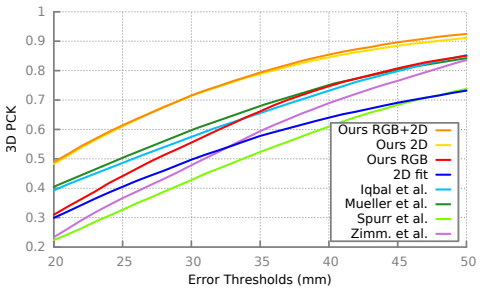


Figure 6: 3D PCK for DEXTER+OBJECT.

	Ours RGB	Ours RGB+2D	Ours 2D	2D fit	Spurr et al.	Zimm. et al.
3D distance	33.16	**25.53**	25.93	41.18	40.20	34.75

Table 2: Average 3D joint distance (mm) to ground-truth for DEXTER+OBJECT.

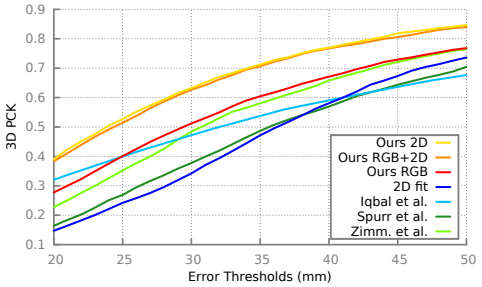


Figure 7: 3D PCK for EGODEXTER.

	Ours RGB	Ours RGB+2D	Ours 2D	2D fit	Spurr et al.	Zimm. et al.
3D distance	51.87	45.58	**45.33**	56.59	56.92	52.77

Table 3: Average 3D joint distance (mm) to ground-truth for EGODEXTER.

Figure 7 shows our results on a hand in ego-centric view and in interaction with various objects in terms of 3D PCK on the EGODEXTER dataset, and Table 3 shows 3D joint errors. Our method outperforms the competition in these settings as illustrated in the Figures. We note that we show relative 3D pose estimates for all methods except [14] where the authors report absolute values.

	Ours RGB	Ours RGB+2D	Ours 2D	2D fit	Zimm. et al.
2D distance	23.04	**18.95**	20.65	22.36	59.40

Table 4: Average re-projected 2D joint distance (px) to ground-truth for MPII+NZSL

We expect our method to perform particularly well on datasets of images in the wild, as our training set contains this type of data and accounts for hands in low resolution, blurry, occluded and in challenging views and pose configu-

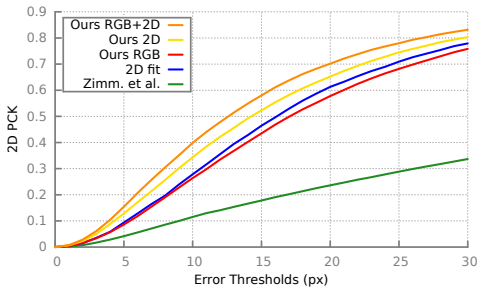


Figure 8: 2D PCK for MPII+NZSL.

rations. In fact, we compare our results to [68] on the testing set of MPII+NZSL dataset in Figure 8 and Table 4 through 2D PCK and 2D joint error respectively. We outperform [68] with a substantial margin as the Figure shows. The superiority of our method on this dataset is visually confirmed in Figure 9.

Comparison to 2D fitting In the case where 2D joint detections are used as input, an alternative way of solving 3D hand pose estimation is to perform a 2D fitting between the re-projected hand model joints and the key-points detected on the image, in a similar fashion to the work proposed by [34]. Our implementation of this strategy consists in minimizing the following objective function with respect to the weak perspective camera parameters $\{R, t, s\}$ and the hand shape and pose parameters $\{\boldsymbol{\beta}, \boldsymbol{\theta}\}$:

$$E(R,t,s,\boldsymbol{\beta},\boldsymbol{\theta}) = \sum_i p_i \left(s\Pi(RJ_i(\boldsymbol{\beta},\boldsymbol{\theta})) + t - \mathrm{x}_i\right)^2 + \alpha_{\boldsymbol{\beta}}\|\boldsymbol{\beta}\|_2^2 + \|\boldsymbol{\theta}\|_2^2, \tag{10}$$

where p_i is the i^{th} hand joint estimate confidence provided by the detector CNN [44]. Similarly to the loss in Equation 9, regularization in the second line of Equation 10 is important to ensure plausible 3D hand reconstructions. We perform this optimization using Powell's Dogleg method [29] within the Chumpy [1] framework.

We compare this method (2D fit) to our proposed approach on datasets STEREO, DEXTER+OBJECT and EGODEXTER with 3D PCK in Figures 5, 6 and 7 and 3D joint error in Tables 1, 2 and 3 respectively, and also on dataset MPII+NZSL with 2D PCK in Figure 8 and 2D joint error in Table 4. Results show that our approach outperforms the 2D fitting based strategy for all datasets. We observe that while the optimization catches up slightly with our method in 2D (MPII+NZSL), its performance drops considerably in 3D. Our method benefits clearly from solving the fitting problem in a learning framework and leverages visual cues in predicting the 3D hand position and configuration, while the 2D fitting relies merely on the 2D joint detection information. We also outperform the 2D fitting based method in [34] that uses a similar hand model to [32] and a perspective projection model on dataset STEREO in Figure 5.

Ablation study We evaluate the difference between using images only (Ours RGB), using 2D joint heat-maps obtained from a state-of-the-art hand detector [44] only (Ours 2D), and finally using both together as input (Ours RGB+2D). We carry comparisons on datasets STEREO, DEXTER+OBJECT and EGODEXTER with 3D PCK in Figures 5, 6 and 7 and 3D joint error in Tables 1, 2 and 3 respectively, and also on dataset MPII+NZSL with 2D PCK in Figure 8 and 2D joint error in Table 4. On dataset STEREO, training on images alone yields the best performance, while training with a combination of images and 2D joint heatmaps is generally the most suitable approach for the other datasets that we tested on.

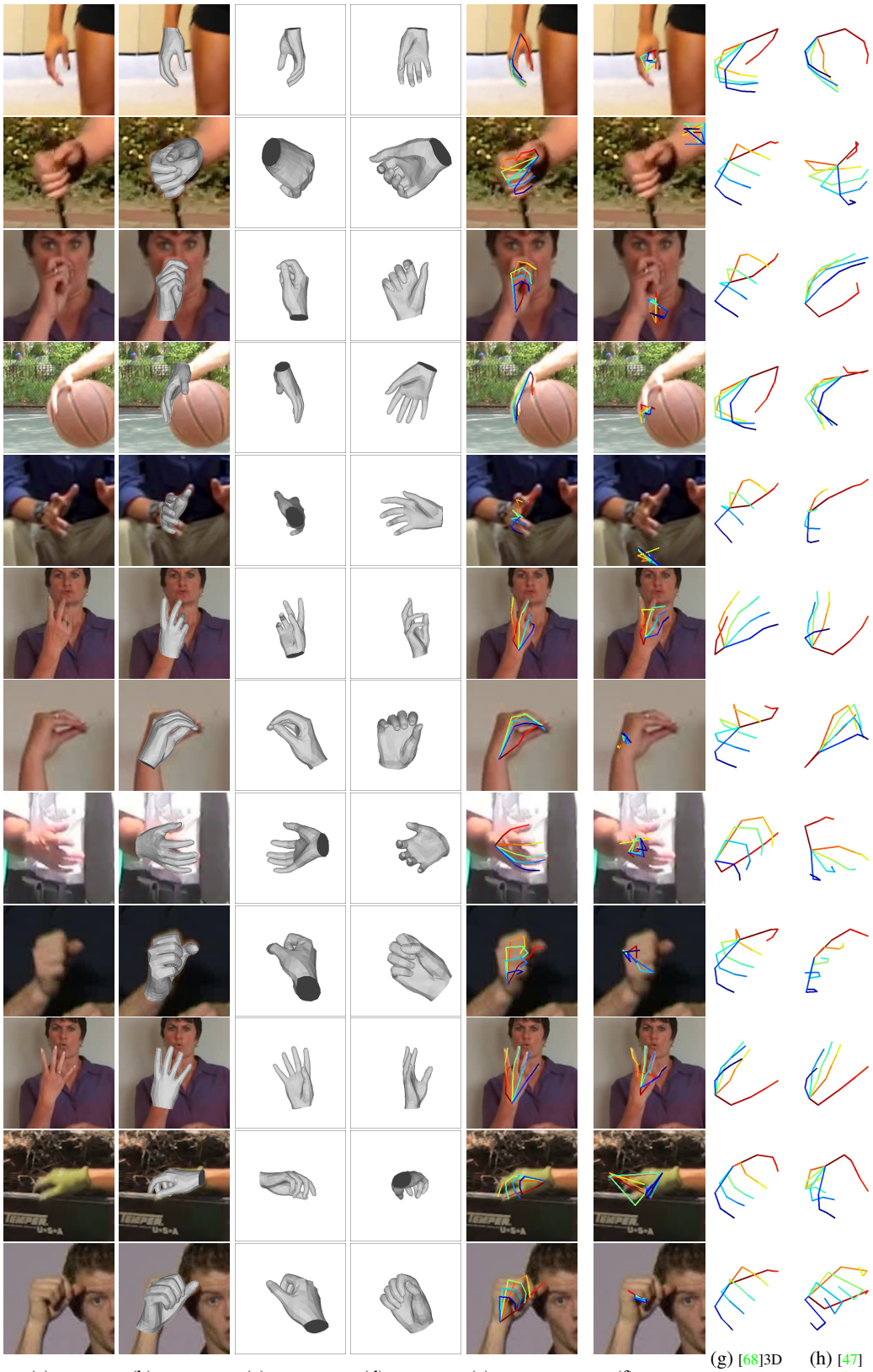
Qualitative Figure 9 shows our 3D hand reconstructions on the challenging testing set of MPII+NZSL. As shown in this Figure, the input data (9a) displays images of hands that are sometimes blurry, low resolved, occluded, viewed from varying viewpoints and in varying pose configurations. We show our 3D mesh overlaid on the input image (9b) and in alternative views (9c, 9d). We also compare our hand skeleton (9e) to the 2D and 3D pose predictions of [68] (9f, 9g) and the 3D predictions of [47] (9h). Our method obtains visually plausible results while the methods in [68] and [47] fail to predict good 3D pose estimates for many cases in the MPII+NZSL dataset. We show more examples in the supplementary material.

9. Conclusion

We presented a method to predict 3D hand pose and shape from a single RGB image. We combine a deep convolutional encoder with a generative hand model as decoder and train the resulting network end-to-end with 2D and 3D hand joint annotated images. The encoder predicts hand parameters that are inputted to the hand model, and view parameters that are used to re-project the generated 3D hand into the image domain. We generate state-of-the-art results on 3D pose benchmarks and show compelling 3D reconstruction on a challenging set of images in the wild. This method could benefit greatly from a hand appearance model by leveraging a photometric loss in training as proposed in [56, 55] for faces. One possible extension to this work could be to allow some components of the MANO [40] model such as the corrective blend shapes $\mathbf{S}$ and $\mathbf{P}$ (Equation 2) to be fine-tuned in training for improved performance.

Acknowledgement

This work was supported by the ERC grant ERC-2012-AdG 321162-HELIOS, the EPSRC grant Seebibyte EP/M013774/1 and the EPSRC/MURI grant EP/N019474/1.



(a) Input (b) Our mesh (c) Back view (d) Side view (e) Our skeleton (f) [68]2D (g) [68]3D (h) [47]

Figure 9: Our 3D hand reconstruction on examples from the challenging testing set of MPII+NZSL compared to the 3D hand pose predictions of [68] and [47].

References

[1] http://chumpy.org. 7

[2] V. F. Abrevaya, S. Wuhrer, and E. Boyer. Multilinear autoencoder for 3d face model learning. In *WACV*, 2018. 3

[3] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *CVPR*, 2014. 5

[4] V. Athitsos and S. Sclaroff. Estimating 3d hand pose from a cluttered image. In *CVPR*, 2003. 1, 2

[5] S. Bambach, S. Lee, D. J. Crandall, and C. Yu. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In *ICCV*, 2015. 5

[6] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *Conference on Computer graphics and interactive techniques*, 1999. 3

[7] Y. Cai, L. Ge, J. Cai, and J. Yuan. Weakly-supervised 3d hand pose estimation from monocular rgb images. In *ECCV*, 2018. 1, 2, 6

[8] T. E. de Campos and D. W. Murray. Regression-based hand pose estimation from multiple cameras. In *CVPR*, 2006. 1, 2

[9] M. de La Gorce, D. J. Fleet, and N. Paragios. Model-based 3d hand pose estimation from monocular video. *IEEE transactions on pattern analysis and machine intelligence*, 2011. 1, 2

[10] L. Ge, Y. Cai, J. Weng, and J. Yuan. Hand pointnet: 3d hand pose estimation using point sets. In *CVPR*, 2018. 1, 2

[11] L. Ge, H. Liang, J. Yuan, and D. Thalmann. Robust 3d hand pose estimation in single depth images: from single-view cnn to multi-view cnns. In *CVPR*, 2016. 2

[12] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *ECCV*, 2016. 4

[13] W. Hürst and C. Van Wezel. Gesture-based interaction via finger tracking for mobile augmented reality. *Multimedia Tools and Applications*, 2013. 1

[14] U. Iqbal, P. Molchanov, T. Breuel, J. Gall, and J. Kautz. Hand pose estimation via latent 2.5 d heatmap regression. In *ECCV*, 2018. 1, 3, 6

[15] Y. Jang, S.-T. Noh, H. J. Chang, T.-K. Kim, and W. Woo. 3d finger cape: Clicking action and position estimation under self-occlusions in egocentric viewpoint. *IEEE Transactions on Visualization and Computer Graphics*, 2015. 1

[16] 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. 5

[17] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik. End-to-end recovery of human shape and pose. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[18] L. Kavan and J. Žára. Spherical blend skinning: A real-time deformation of articulated models. In *Symposium on Interactive 3D Graphics and Games*, 2005. 1, 2, 3

[19] C. Keskin, F. Kıraç, Y. E. Kara, and L. Akarun. Hand pose estimation and hand shape classification using multi-layered randomized decision forests. In *ECCV*, 2012. 2

[20] S. Khamis, J. Taylor, J. Shotton, C. Keskin, S. Izadi, and A. Fitzgibbon. Learning an efficient model of hand shape variation from depth images. In *CVPR*, 2015. 2, 3

[21] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[22] S. Laine, T. Karras, T. Aila, A. Herva, S. Saito, R. Yu, H. Li, and J. Lehtinen. Production-level facial performance capture using deep convolutional neural networks. In *Symposium on Computer Animation*, 2017. 3

[23] C. Li and K. M. Kitani. Pixel-level hand detection in egocentric videos. In *CVPR*, 2013. 5

[24] P. Li, H. Ling, X. Li, and C. Liao. 3d hand pose estimation using randomized decision forest with segmentation index points. In *ICCV*, 2015. 2

[25] 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) (Proc. SIGGRAPH Asia)*, 2015. 3

[26] A. Makris and A. Argyros. Model-based 3d hand tracking with on-line hand shape adaptation. 2015. 2

[27] F. Mueller, F. Bernard, O. Sotnychenko, D. Mehta, S. Sridhar, D. Casas, and C. Theobalt. Ganerated hands for real-time 3d hand tracking from monocular rgb. In *CVPR*, 2018. 1, 2, 6

[28] F. Mueller, D. Mehta, O. Sotnychenko, S. Sridhar, D. Casas, and C. Theobalt. Real-time hand tracking under occlusion from an egocentric rgb-d sensor. In *ICCV*, 2017. 2, 5

[29] J. Nocedal and S. J. Wright. *Nonlinear Equations*. Springer, 2006. 7

[30] M. Oberweger, P. Wohlhart, and V. Lepetit. Training a feedback loop for hand pose estimation. In *ICCV*, 2015. 2

[31] I. Oikonomidis, N. Kyriazis, and A. A. Argyros. Markerless and efficient 26-dof hand pose recovery. In *ACCV*, 2010. 1, 2

[32] I. Oikonomidis, N. Kyriazis, and A. A. Argyros. Efficient model-based 3d tracking of hand articulations using kinect. In *BMVC*, 2011. 2, 3, 7

[33] P. Panteleris and A. Argyros. Back to rgb: 3d tracking of hands and hand-object interactions based on short-baseline stereo. *Hands17 Workshop ICCV*, 2017. 1, 2

[34] P. Panteleris, I. Oikonomidis, and A. Argyros. Using a single rgb frame for real time 3d hand pose estimation in the wild. In *WACV*, 2018. 2, 6, 7

[35] T. Piumsomboon, A. Clark, M. Billinghurst, and A. Cockburn. User-defined gestures for augmented reality. In *IFIP Conference on Human-Computer Interaction*, 2013. 1

[36] G. Poier, D. Schinagl, and H. Bischof. Learning pose specific representations by predicting different views. In *CVPR*, 2018. 1, 2

[37] C. Qian, X. Sun, Y. Wei, X. Tang, and J. Sun. Realtime and robust hand tracking from depth. In *CVPR*, 2014. 2

[38] D. A. R. McKee, D. McKee and E. Pailla. Nz sign language exercises. *Deaf Studies Department of Victoria University of Wellington*. 5

[39] J. Romero, H. Kjellstrm, and D. Kragic. Hands in action: real-time 3d reconstruction of hands in interaction with objects. In *ICRA*, 2010. 1, 2

[40] J. Romero, D. Tzionas, and M. J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (TOG)*, 2017. 1, 2, 3, 4, 6, 7

[41] R. Rosales, V. Athitsos, L. Sigal, and S. Sclaroff. 3d hand pose reconstruction using specialized mappings. In *ICCV*, 2001. 1, 2

[42] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics (TOG)*, 2004. 5

[43] T. Sharp, C. Keskin, D. Robertson, J. Taylor, J. Shotton, D. Kim, C. Rhemann, I. Leichter, A. Vinnikov, Y. Wei, et al. Accurate, robust, and flexible real-time hand tracking. In *ACM CHI*, 2015. 2

[44] T. Simon, H. Joo, I. Matthews, and Y. Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *CVPR*, 2017. 1, 2, 4, 5, 7

[45] A. Sinha, C. Choi, and K. Ramani. Deephand: Robust hand pose estimation by completing a matrix imputed with deep features. In *CVPR*, 2016. 2

[46] J. Song, G. Sörös, F. Pece, S. R. Fanello, S. Izadi, C. Keskin, and O. Hilliges. In-air gestures around unmodified mobile devices. In *ACM Symposium on User Interface Software and Technology*, 2014. 1

[47] A. Spurr, J. Song, S. Park, and O. Hilliges. Cross-modal deep variational hand pose estimation. In *CVPR*, 2018. 1, 2, 3, 6, 7, 8

[48] S. Sridhar, F. Mueller, A. Oulasvirta, and C. Theobalt. Fast and robust hand tracking using detection-guided optimization. In *CVPR*, 2015. 2

[49] S. Sridhar, F. Mueller, M. Zollhoefer, D. Casas, A. Oulasvirta, and C. Theobalt. Real-time joint tracking of a hand manipulating an object from rgb-d input. In *ECCV*, 2016. 2, 5

[50] S. Sridhar, H. Rhodin, H.-P. Seidel, A. Oulasvirta, and C. Theobalt. Real-time hand tracking using a sum of anisotropic gaussians model. In *3DV*, 2014. 1, 2, 3

[51] B. Stenger, P. R. Mendonça, and R. Cipolla. Model-based 3d tracking of an articulated hand. In *CVPR*, 2001. 1, 2

[52] X. Sun, Y. Wei, S. Liang, X. Tang, and J. Sun. Cascaded hand pose regression. In *CVPR*, 2015. 2

[53] D. Tang, J. Taylor, P. Kohli, C. Keskin, T.-K. Kim, and J. Shotton. Opening the black box: Hierarchical sampling optimization for estimating human hand pose. In *ICCV*, 2015. 2

[54] J. Taylor, R. Stebbing, V. Ramakrishna, C. Keskin, J. Shotton, S. Izadi, A. Hertzmann, and A. Fitzgibbon. User-specific hand modeling from monocular depth sequences. In *CVPR*, 2014. 3

[55] A. Tewari, M. Zollhöfer, P. Garrido, F. Bernard, H. Kim, P. Pérez, and C. Theobalt. Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz. In *CVPR*, 2018. 3, 7

[56] A. Tewari, M. Zollhöfer, H. Kim, P. Garrido, F. Bernard, P. Pérez, and C. Theobalt. Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. In *iCCV*, 2017. 3, 7

[57] A. Thayananthan, B. Stenger, P. H. Torr, and R. Cipolla. Shape context and chamfer matching in cluttered scenes. In *CVPR*, 2003. 1, 2

[58] A. Tkach, M. Pauly, and A. Tagliasacchi. Sphere-meshes for real-time hand modeling and tracking. *ACM TOG*, 2016. 3

[59] J. Tompson, M. Stein, Y. Lecun, and K. Perlin. Real-time continuous pose recovery of human hands using convolutional networks. *ACM ToG*, 2014. 2

[60] H.-Y. Tung, H.-W. Tung, E. Yumer, and K. Fragkiadaki. Self-supervised learning of motion capture. In *NIPS*, 2017. 3

[61] C. Wan, T. Probst, L. Van Gool, and A. Yao. Dense 3d regression for hand pose estimation. In *CVPR*, 2018. 1, 2

[62] X. Wu, D. Finnegan, E. O'Neill, and Y.-L. Yang. Handmap: Robust hand pose estimation via intermediate dense guidance map supervision. In *ECCV*, 2018. 1, 2

[63] Y. Wu, J. Y. Lin, and T. S. Huang. Capturing natural hand articulation. In *ICCV*, 2001. 1, 2

[64] C. Xu and L. Cheng. Efficient hand pose estimation from a single depth image. In *ICCV*, 2013. 2

[65] J. Zhang, J. Jiao, M. Chen, L. Qu, X. Xu, and Q. Yang. 3d hand pose tracking and estimation using stereo matching. *arXiv preprint arXiv:1610.07214*, 2016. 1, 2, 5, 6

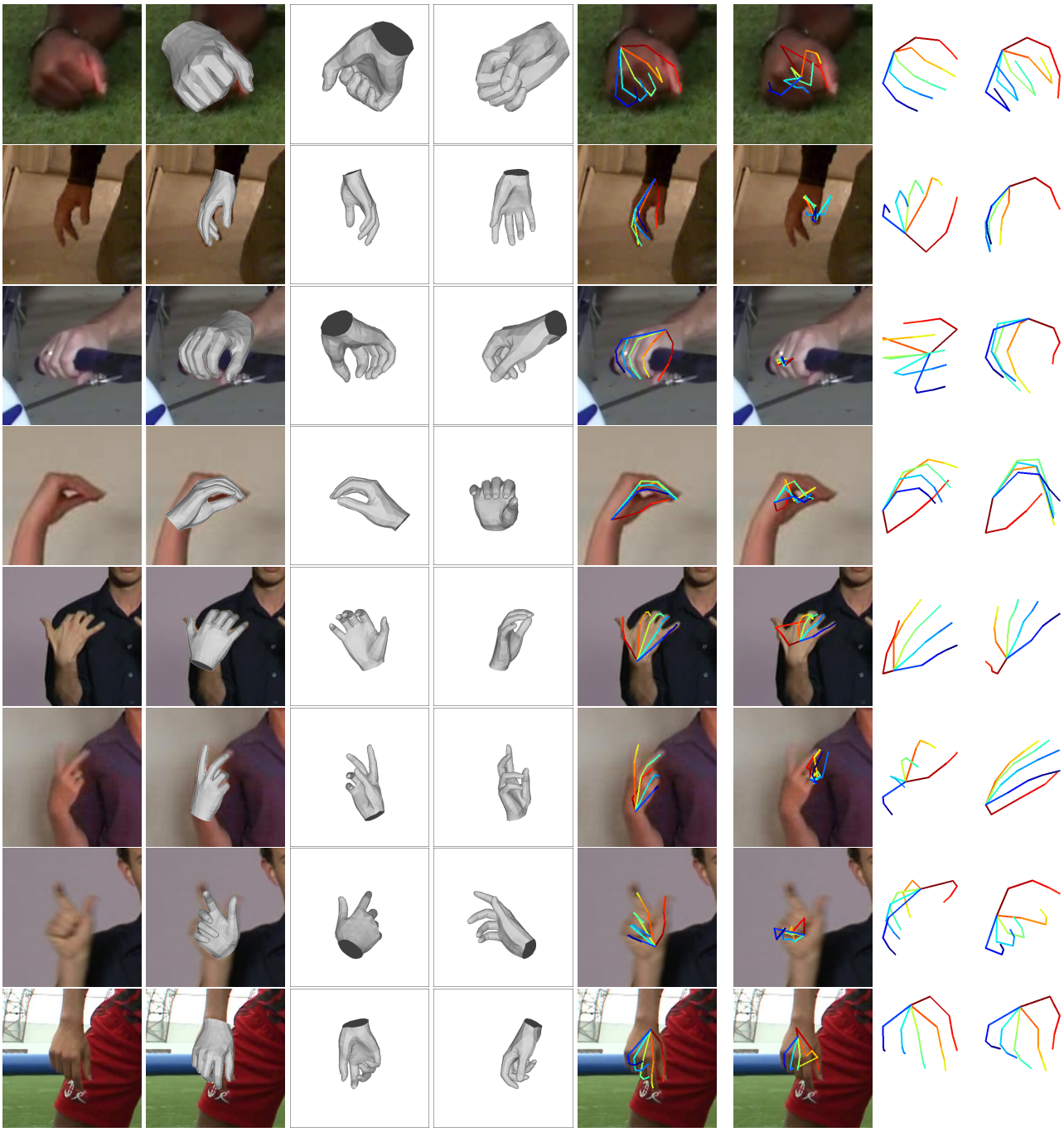
[66] Y. Zhou, J. Lu, K. Du, X. Lin, Y. Sun, and X. Ma. Hbe: Hand branch ensemble network for real-time 3d hand pose estimation. In *ECCV*, 2018. 1, 2

[67] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 2

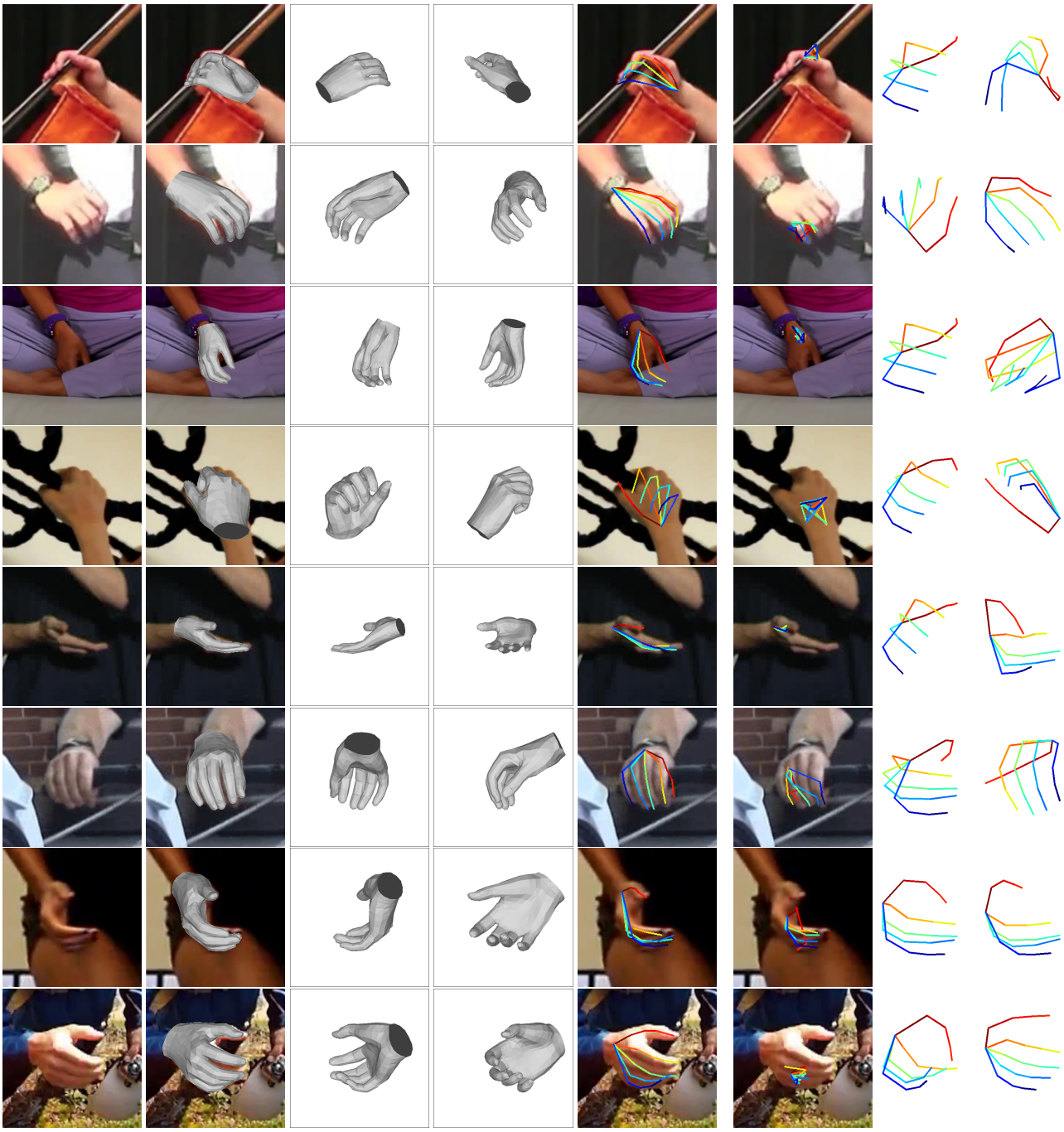
[68] C. Zimmermann and T. Brox. Learning to estimate 3d hand pose from single rgb images. In *ICCV*, 2017. 1, 2, 5, 6, 7, 8

3D hand shape and pose from images in the wild

– SUPPLEMENTARY MATERIAL –



(a) Input (b) Our mesh (c) Back view (d) Side view (e) Our skeleton (f) [2] 2D (g) [2] 3D (h) [1] 3D



(a) Input (b) Our mesh (c) Back view (d) Side view (e) Our skeleton (f) [2] 2D (g) [2] 3D (h) [1] 3D

Our 3D hand reconstruction on examples from the challenging testing set of MPII+NZSL compared to the 3D hand pose predictions of [2] and [1].

References

[1] A. Spurr, J. Song, S. Park, and O. Hilliges. Cross-modal deep variational hand pose estimation. In *CVPR*, 2018. 1, 2

[2] C. Zimmermann and T. Brox. Learning to estimate 3d hand pose from single rgb images. In *ICCV*, 2017. 1, 2

References

3Lateral (2018). 3Lateral. URL: http://www.3lateral.com/ (visited on 11/29/2018) (cit. on pp. 4, 5).

Achilles, F., Ichim, A. E., Coskun, H., Tombari, F., Noachtar, S., and Navab, N. (2016). Patient MoCap: Human pose estimation under blanket occlusion for hospital monitoring applications. In: *19th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 491–499 (cit. on pp. 6, 7, 82).

Adams, A., Baek, J., and Davis, M. A. (2010). Fast high-dimensional filtering using the permutohedral lattice. In: *Computer Graphics Forum*. Vol. 29. 2, pp. 753–762 (cit. on pp. 40, 52, 114).

Aggarwal, J. K., Cai, Q., Liao, W., and Sabata, B. (1998). Nonrigid Motion Analysis: Articulated and Elastic Motion. In: *Computer Vision and Image Understanding* **70** (2), pp. 142–156 (cit. on p. 11).

Aggarwal, J. K. and Cai, Q. (1999). Human Motion Analysis: A Review. In: *Computer Vision and Image Understanding* **73** (3), pp. 428–440 (cit. on p. 11).

Aggarwal, J., Cai, Q., Liao, W., and Sabata, B. (1994). Articulated and elastic non-rigid motion: a review. In: *Proceedings of 1994 IEEE Workshop on Motion of Non-rigid and Articulated Objects*, pp. 2–14 (cit. on p. 11).

Aggarwal, J. and Cai, Q. (1997). Human motion analysis: a review. In: *Proceedings IEEE Nonrigid and Articulated Motion Workshop*, pp. 90–102 (cit. on p. 11).

Akhter, I. and Black, M. J. (2015). Pose-conditioned joint angle limits for 3D human pose reconstruction. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1446–1455 (cit. on pp. 3, 111).

Alibaba (2018). FashionAI Global Challenge 2018. URL: https://tianchi.aliyun.com/markets/tianchi/FashionAIeng?{_}lang=en{_}US (visited on 12/05/2018) (cit. on p. 7).

Amazon (2018a). Echo Look - Hands-Free Camera and Style Assistant. URL: https://www.amazon.com/Amazon-Echo-Look-Camera-Style-Assistant/dp/B0186JAEWK (visited on 12/05/2018) (cit. on pp. 7, 8).

Amazon (2018b). Prime Air. URL: http://amzn.to/primeair (visited on 12/15/2018) (cit. on p. 6).

Andrej Karpathy (2012). The state of Computer Vision and AI: we are really, really far away. URL:

http://karpathy.github.io/2012/10/22/state-of-computer-vision/ (visited on 12/27/2018) (cit. on p. 118).

Andriluka, M., Pishchulin, L., Gehler, P., and Schiele, B. (2014). 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3686–3693 (cit. on pp. 15, 25, 27, 28, 31, 33, 42, 43, 82, 112, 117).

Arbrea Labs (2018). Arbrea Labs – Advanced Solutions for Cosmetic Surgery. URL: https://arbrea-labs.com/ (visited on 12/04/2018) (cit. on p. 7).

Balakrishnan, G., Zhao, A., Dalca, A. V., Durand, F., and Guttag, J. (2018). Synthesizing Images of Humans in Unseen Poses. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8340–8348 (cit. on pp. 4, 54, 82, 86).

Balmain (2018). Balmain's New Virtual Army. URL: https://www.balmain.com/us/balmain/balmains-new-virtual-army (visited on 12/04/2018) (cit. on p. 8).

Bao, J., Chen, D., Wen, F., Li, H., and Hua, G. (2017). CVAE-GAN: Fine-Grained Image Generation through Asymmetric Training. In: *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2764–2773 (cit. on pp. 56, 60).

Belagiannis, V. and Zisserman, A. (2017). Recurrent Human Pose Estimation. In: *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pp. 468–475 (cit. on pp. 12, 28, 29, 44, 47).

Bengio, Y. (2009). Learning Deep Architectures for AI. In: *Foundations and Trends® in Machine Learning* **2** (1), pp. 1–127 (cit. on pp. 12, 18).

Bishop, C. M. and Christopher, M (2006). Pattern Recognition and Machine Learning. 2nd ed. Vol. 4. 4. Springer Science, p. 738 (cit. on pp. 19, 21).

Boeing (2018). William Fetter's Boeing Man. URL: https://secure.boeingimages.com/archive/William-Fetter-s-Boeing-Man-2F3XC5YCZNC.html (visited on 12/08/2018) (cit. on p. 2).

Borji, A. and Itti, L. (2013). State-of-the-art in visual attention modeling. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* **35** (1), pp. 185–207 (cit. on p. 35).

Borshukov, G., Piponi, D., Larsen, O., Lewis, J. P., and Tempelaar-lietz, C. (2005). Universal capture - image-based facial animation for "The Matrix Reloaded". In: *ACM SIGGRAPH 2005 Courses on - SIGGRAPH '05*. ACM Press, p. 16 (cit. on pp. 2, 3).

Bourdev, L. and Malik, J. (2009). Poselets: Body part detectors trained using 3D human pose annotations. In: *2009 IEEE 12th International Conference on Computer Vision (ICCV)*, pp. 1365–1372 (cit. on p. 31).

Bulat, A. and Tzimiropoulos, G. (2016). Human pose estimation via convolutional part heatmap regression. In: *14th European Conference on Computer Vision (ECCV)*, pp. 717–732 (cit. on pp. 12, 28, 44, 50, 82).

Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. (2017). Realtime Multi-person 2D Pose Estimation Using Part Affinity Fields. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1302–1310 (cit. on pp. 12, 66, 82).

Carreira, J., Agrawal, P., Fragkiadaki, K., and Malik, J. (2016). Human Pose Estimation with Iterative Error Feedback. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4733–4742 (cit. on pp. 29, 50).

Caruana, R. (1998). Multitask learning. In: *Learning to learn*. Vol. 28. Springer, pp. 95–133 (cit. on pp. 29, 31, 36).

Cédras, C. and Shah, M. (1995). Motion-based recognition a survey. In: *Image and Vision Computing* **13** (2), pp. 129–155 (cit. on p. 11).

Chan, C., Ginosar, S., Zhou, T., and Efros, A. A. (2018). Everybody Dance Now. In: *CoRR* **abs/1808.0** (cit. on pp. 82, 86).

Chen, X. and Yuille, A. (2014). Articulated Pose Estimation by a Graphical Model with Image Dependent Pairwise Relations. In: *Advances in Neural Information Processing Systems 27 (NIPS)*, pp. 1736–1744 (cit. on pp. 12, 29, 30).

Chu, X., Yang, W., Ouyang, W., Ma, C., Yuille, A. L., and Wang, X. (2017). Multi-context Attention for Human Pose Estimation. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5669–5678 (cit. on pp. 12, 29, 30, 47, 50, 54, 62, 82).

CNBC (2018). Video game industry is booming with continued revenue. URL: https://www.cnbc.com/2018/07/18/video-game-industry-is-booming-with-continued-revenue.html (visited on 12/05/2018) (cit. on p. 4).

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The Cityscapes Dataset for Semantic Urban Scene Understanding. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 3213–3223 (cit. on p. 6).

Dai, J., He, K., and Sun, J. (2016). Instance-Aware Semantic Segmentation via Multi-task Network Cascades. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3150–3158 (cit. on pp. 29, 30).

Dalal, N. and Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Vol. 1, pp. 886–893 (cit. on p. 30).

de Bem, R., Arnab, A., Golodetz, S., Sapienza, M., and Torr, P. (2018). Deep Fully-Connected Part-Based Models for Human Pose Estimation. In: *10th Asian Conference on Machine Learning (ACML)*, pp. 327–342 (cit. on pp. 54, 62).

de La Gorce, M., Fleet, D. J., and Paragios, N. (2011). Model-Based 3D Hand Pose Estimation from Monocular Video. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33** (9), pp. 1793–1805 (cit. on p. 11).

Dempster, A, Laird, N, and Rubin, D (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. In: *Journal of the Royal Statistical Society. Series B (Methodological)* **39** (1), pp. 1–38 (cit. on p. 21).

Deng, L. and Jaitly, N. (2016). Deep Discriminative and Generative Models for Pattern Recognition. In: *Handbook of Pattern Recognition and Computer Vision*, pp. 27–52 (cit. on p. 17).

Department for Work and Pensions - GOV.UK (2017). Family Resources Survey: financial year 2015/16. URL: https://www.gov.uk/government/statistics/family-resources-survey-financial-year-201516 (visited on 12/06/2018) (cit. on p. 7).

Elgammal, A. and Chan-Su Lee (2004). Inferring 3D body pose from silhouettes using activity manifold learning. In: *2004 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 681–688 (cit. on p. 92).

Elhayek, A., Aguiar, E. de, Jain, A., Tompson, J., Pishchulin, L., Andriluka, M., Bregler, C., Schiele, B., and Theobalt, C. (2015). Efficient ConvNet-based marker-less motion capture in general scenes with a low number of cameras. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3810–3818 (cit. on pp. 12, 26).

Epic Games (2018). Unreal Engine. URL: https://www.unrealengine.com/ (visited on 12/04/2018) (cit. on p. 5).

Esser, P., Sutter, E., and Ommer, B. (2018). A Variational U-Net for Conditional Appearance and Shape Generation. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8857–8866 (cit. on p. 86).

Facebook (2018). Oculus. URL: https://www.oculus.com/ (visited on 12/05/2018) (cit. on p. 8).

Fan, S., Ng, T.-T., Koenig, B. L., Herberg, J. S., Jiang, M., Shen, Z., and Zhao, Q. (2018). Image Visual Realism: From Human Perception to Machine Computation. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* **40** (9), pp. 2180–2193 (cit. on p. 10).

Fan, X., Zheng, K., Lin, Y., and Wang, S. (2015). Combining local appearance and holistic view: Dual-Source Deep Neural Networks for human pose estimation. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1347–1355 (cit. on p. 29).

Fashion United (2018). Global fashion industry statistics - International apparel. URL: https://fashionunited.com/global-fashion-industry-statistics (visited on 12/05/2018) (cit. on p. 7).

Felzenszwalb, P. F. and Huttenlocher, D. P. (2005). Pictoral structures for object recognition. In: *International Journal of Computer Vision* **61** (1), pp. 55–79 (cit. on p. 29).

Fernando, B., Gavves, E., Oramas, J. M., Ghodrati, A., and Tuytelaars, T. (2015). Modeling video evolution for action recognition. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5378–5387 (cit. on p. 29).

Ferrari, V., Marin-Jimenez, M., and Zisserman, A. (2008). Progressive search space reduction for human pose estimation. In: *2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8 (cit. on p. 43).

Fetter, W. A. (1982). A Progression of Human Figures Simulated by Computer Graphics. In: *IEEE Computer Graphics and Applications* **2** (9), pp. 9–13 (cit. on p. 2).

Fischler, M. A. and Elschlager, R. A. (1973). The Representation and Matching of Pictorial Structures Representation. In: *IEEE Transactions on Computers* **C-22** (1), pp. 67–92 (cit. on p. 29).

Fleet, D. J. (2011). Motion Models for People Tracking. In: *Visual Analysis of Humans - Looking at People*. Pp. 171–198 (cit. on p. 54).

Franco, J. S. and Boyer, E. (2005). Fusion of multi-view silhouette cues using a space occupancy grid. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1747–1753 (cit. on p. 11).

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. In: *Biological Cybernetics* **36** (4), pp. 193–202 (cit. on p. 19).

Gavrila, D. M. (1999). The visual analysis of human movement: A survey. In: *Computer Vision and Image Understanding* **73** (1), pp. 82–98 (cit. on p. 11).

Geiger, A, Lenz, P, Stiller, C, and Urtasun, R (2013). Vision meets robotics: The KITTI dataset. In: *The International Journal of Robotics Research* **32** (11), pp. 1231–1237 (cit. on p. 6).

Gkioxari, G., Toshev, A., and Jaitly, N. (2016). Chained Predictions Using Convolutional Neural Networks. In: *14th European Conference on Computer Vision (ECCV)*, pp. 728–743 (cit. on pp. 12, 28, 50).

Gong, K., Liang, X., Zhang, D., Shen, X., and Lin, L. (2017). Look into person: Self-supervised structure-sensitive learning and a new benchmark for human parsing. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 932–940 (cit. on pp. 33, 117).

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative Adversarial Nets. In: *Advances in Neural Information Processing Systems 27 (NIPS)*, pp. 2672–2680 (cit. on pp. 12, 13, 20, 21, 54, 56, 82, 83, 87).

Goodfellow, I., Bengio, Y., and Courville, A. (2016). Deep Learning. MIT Press (cit. on pp. 3, 12, 18, 92).

Goodrich, M. A. and Schultz, A. C. (2007). Human-Robot Interaction: A Survey. In: *Foundations and Trends® in Human-Computer Interaction* **1** (3), pp. 203–275 (cit. on p. 5).

Güler, R. A., Neverova, N., and Kokkinos, I. (2018). DensePose: Dense Human Pose Estimation In The Wild. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7297–7306 (cit. on p. 12).

Hand Talk (2018). Hand Talk - Sign Language Translator. URL: https://www.handtalk.me/ (visited on 12/06/2018) (cit. on p. 7).

Haritaoglu, I., Harwood, D., and Davis, L. (2000). W4: real-time surveillance of people and their activities. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22** (8), pp. 809–830 (cit. on p. 8).

Hayder, Z., Salzmann, M., and He, X. (2014). Object co-detection via efficient inference in a fully-connected CRF. In: *13th European Conference on Computer Vision (ECCV)*, pp. 330–345 (cit. on pp. 28, 115).

Haykin, S. (1998). Neural Network: A Comprehensive Foundation. 2nd. Prentice Hall, pp. 71–80 (cit. on p. 19).

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In: *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1026–1034 (cit. on pp. 34, 44, 67, 96).

Heitz, G., Gould, S., Saxena, A., and Koller, D. (2009). Cascaded classification models: Combining models for holistic scene understanding. In: *Advances in Neural Information Processing Systems 21 (NIPS)*, pp. 641–648 (cit. on p. 30).

Hilton, A, Beresford, D, Gentils, T, Smith, R, and Sun, W. (1999). Virtual people: capturing human models to populate virtual worlds. In: *Proceedings of Computer Animation*, pp. 174–185 (cit. on p. 1).

Hilton, A., Guillemaut, J.-Y., Kilner, J., Grau, O., and Thomas, G. (2011). 3D-TV Production From Conventional Cameras for Sports Broadcast. In: *IEEE Transactions on Broadcasting* **57** (2), pp. 462–476 (cit. on p. 8).

Hu, P. and Ramanan, D. (2015). Bottom-Up and Top-Down Reasoning with Hierarchical Rectified Gaussians. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5600–5609 (cit. on pp. 29, 50).

Hubel, D. H. and Wiesel, T. N. (1962). Receptive Fields, Binocular Interaction and Functional Architecture in Cats Visual Cortex. In: *Journal of Physiology-London* **160** (1), 106–& (cit. on p. 19).

Ian Spriggs (2018). The Portrait of Erica. URL: http://www.ianspriggs.com/portrait-of-erica/ (visited on 12/27/2018) (cit. on p. 119).

Insafutdinov, E., Pishchulin, L., Andres, B., Andriluka, M., and Schiele, B. (2016). DeeperCut: A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model. In: *14th European Conference on Computer Vision (ECCV)* (cit. on pp. 12, 29, 30, 44, 47, 50, 66).

Ionescu, C., Papava, D., Olaru, V., and Sminchisescu, C. (2014). Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* **36** (7), pp. 1325–1339 (cit. on pp. 15, 16, 53, 55, 65, 82, 95, 112, 113).

Isola, P., Zhu, J. Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976 (cit. on pp. 21, 54, 56, 57, 76, 86, 87).

Itti, L., Koch, C., and Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* **20** (11), pp. 1254–1259 (cit. on pp. 35, 115).

Jain, A., Tompson, J., LeCun, Y., and Bregler, C. (2015). MoDeep: A deep learning framework using motion features for human pose estimation. In: *12th Asian Conference on Computer Vision (ACCV)*, pp. 302–315 (cit. on pp. 28, 29).

James Vincent (2018). Nvidia has created the first game demo using AI-generated graphics - The Verge. URL: https://www.theverge.com/2018/12/3/18121198/ai-generated-video-game-graphics-nvidia-driving-demo-neurips (visited on 12/27/2018) (cit. on p. 120).

Jhuang, H., Gall, J., Zuffi, S., Schmid, C., and Black, M. J. (2013). Towards Understanding Action Recognition. In: *2013 IEEE International Conference on Computer Vision (CVPR)*, pp. 3192–3199 (cit. on pp. 3, 54, 111).

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe: Convolutional Architecture for Fast Feature Embedding. In: *Proceedings of the ACM International Conference on Multimedia - MM '14*, pp. 675–678 (cit. on pp. 44, 67, 97).

Johansson, G. (1973). Visual perception of biological motion and a model for its analysis. In: *Perception & Psychophysics* **14** (2), pp. 201–211 (cit. on pp. 1, 11).

Johansson, G. (1975). Visual Motion Perception. In: *Scientific American* **232** (6), pp. 76–88 (cit. on pp. 1, 2, 11).

Johnson, S. and Everingham, M. (2010). Clustered Pose and Nonlinear Appearance Models for Human Pose Estimation. In: *Procedings of the British Machine Vision Conference (BMVC)*, pp. 12.1–12.11 (cit. on pp. 15, 25, 27, 28, 33, 42, 43, 70, 72, 82, 112).

Johnson, S. and Everingham, M. (2011). Learning Effective Human Pose Estimation from Inaccurate Annotation. In: *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 43).

Kanade, T., Rander, P., and Narayanan, P. J. (1997). Virtualized reality: Constructing virtual worlds from real scenes. In: *IEEE Multimedia* **4** (1), pp. 34–47 (cit. on p. 2).

Kanazawa, A., Black, M. J., Jacobs, D. W., and Malik, J. (2017). End-to-end Recovery of Human Shape and Pose. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 12).

Karras, T., Aila, T., Laine, S., and Lehtinen, J. (2018). Progressive Growing of GANs for Improved Quality, Stability, and Variation. In: *International Conference on Learning Representations (ICRL)* (cit. on p. 60).

Ke, L., Chang, M.-C., Qi, H., and Lyu, S. (2018). Multi-Scale Structure-Aware Network for Human Pose Estimation. In: *15th European Conference on Computer Vision (ECCV)*, pp. 731–746 (cit. on pp. 29, 50).

Kiefel, M. and Gehler, P. V. (2014). Human pose estimation with fields of parts. In: *13th European Conference on Computer Vision (ECCV)*, pp. 331–346 (cit. on pp. 30, 39, 41).

Kingma, D. P. and Ba, J. (2014a). Adam: A Method for Stochastic Optimization. In: *International Conference on Learning Representations (ICLR)* (cit. on pp. 44, 66, 96).

Kingma, D. P. and Welling, M. (2014b). Auto-Encoding Variational Bayes. In: *International Conference on Machine Learning (ICML)* (cit. on pp. 12, 20, 22, 23, 54, 56, 58, 61, 82, 87, 91, 93).

Kingma, D. P., Rezende, D. J., Mohamed, S., and Welling, M. (2014c). Semi-Supervised Learning with Deep Generative Models. In: *Advances in Neural Information Processing Systems 27 (NIPS)*, pp. 3581–3589 (cit. on pp. 23, 56, 83, 88, 113).

Kleinsmith, A. and Bianchi-Berthouze, N. (2007). Recognizing Affective Dimensions from Body Posture. In: *Affective Computing and Intelligent Interaction*, pp. 48–58 (cit. on p. 54).

Kolesnikov, A. and Lampert, C. H. (2016). Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In: *14th European Conference on Computer Vision (ECCV)*, pp. 695–711 (cit. on p. 115).

Koller, D. and Friedman, N. (2009). Probabilistic Graphical Models: Principles and Techniques. MIT Press (cit. on pp. 29, 39).

Krähenbühl, P. and Koltun, V. (2011). Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials. In: *Advances in Neural Information Processing Systems 24 (NIPS)*, pp. 109–117 (cit. on pp. 28, 30, 37, 41, 112).

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In: *Advances in Neural Information Processing Systems 25 (NIPS)*, pp. 1097–1105 (cit. on pp. 12, 44).

Kulkarni, T. D., Whitney, W., Kohli, P., and Tenenbaum, J. B. (2015). Deep Convolutional Inverse Graphics Network. In: *Advances in Neural Information Processing Systems 28 (NIPS)*, pp. 2539–2547 (cit. on p. 83).

Lafferty, J., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: *Proceedings of the 18th International Conference on Machine Learning (ICML)*, pp. 282–289 (cit. on p. 29).

Larsen, A. B. L., Sønderby, S. K., Larochelle, H., and Winther, O. (2016). Autoencoding beyond pixels using a learned similarity metric. In: *Proceedings of the 33nd International Conference on Machine Learning (ICML)*, pp. 1558–1566 (cit. on pp. 55–57, 60, 83, 87–89, 112).

Lassner, C., Pons-Moll, G., and Gehler, P. V. (2017). A Generative Model of People in Clothing. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 853–862 (cit. on pp. 8, 13, 15, 16, 21, 53–57, 62, 65–67, 73–76, 79, 84, 86, 87, 95, 96, 113, 115, 117).

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation Applied to Handwritten Zip Code Recognition. In: *Neural Computation* **1** (4), pp. 541–551 (cit. on pp. 18, 19).

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. In: *Proceedings of the IEEE* **86** (11), pp. 2278–2324 (cit. on p. 19).

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. In: *Nature* **521** (7553), pp. 436–444 (cit. on pp. 12, 18, 20).

LeCun, Y. A., Bottou, L., Orr, G. B., and Müller, K.-R. (2012). Efficient BackProp. In: *Neural networks: tricks of the trade*. Vol. 53. Lecture Notes in Computer Science LNCS˜1524 9. Springer Verlag, pp. 9–48 (cit. on p. 18).

Li, K., Wu, Z., Peng, K.-C., Ernst, J., and Fu, Y. (2018). Tell Me Where to Look: Guided Attention Inference Network. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9215–9223 (cit. on p. 63).

Li, S., Liu, Z. Q., and Chan, A. B. (2015). Heterogeneous Multi-task Learning for Human Pose Estimation with Deep Convolutional Neural Network. In: *International Journal of Computer Vision* **113** (1), pp. 19–36 (cit. on p. 29).

Liang, X., Liu, S., Shen, X., Yang, J., Liu, L., Dong, J., Lin, L., and Yan, S. (2015). Deep Human Parsing with Active Template Regression. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37** (12), pp. 2402–2414 (cit. on p. 66).

Lifshitz, I., Fetaya, E., and Ullman, S. (2016). Human pose estimation using deep consensus voting. In: *14th European Conference on Computer Vision (ECCV)*, pp. 246–260 (cit. on pp. 29, 50).

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In: *13th European Conference on Computer Vision (ECCV)*, pp. 740–755 (cit. on p. 117).

LISA (2015). VIVA: Vision for intelligent vehicles and applications challenge. URL: http://cvrr.ucsd.edu/vivachallenge/ (visited on 01/04/2019) (cit. on p. 6).

Liu, J., Fan, Q., Pankanti, S., and Metaxas, D. N. (2016a). People detection in crowded scenes by context-driven label propagation. In: *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1–9 (cit. on pp. 28, 115).

Liu, M.-Y., Breuel, T., and Kautz, J. (2017). Unsupervised Image-to-Image Translation Networks. In: *Advances in Neural Information Processing Systems 30 (NIPS)*, pp. 700–708 (cit. on p. 68).

Liu, Z., Luo, P., Qiu, S., Wang, X., and Tang, X. (2016b). DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1096–1104 (cit. on pp. 16, 53, 66, 75, 95, 113).

Long, J, Shelhamer, E, and Darrell, T (2015). Fully convolutional networks for semantic segmentation. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440 (cit. on p. 29).

Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., and Black, M. J. (2015). SMPL: a skinned multi-person linear model. In: *ACM Transactions on Graphics* **34** (6), pp. 1–16 (cit. on pp. 66, 115).

Ma, L., Sun, Q., Georgoulis, S., Van Gool, L., Schiele, B., and Fritz, M. (2017a). Disentangled Person Image Generation. In: *arXiv preprint arXiv:1712.02621* (cit. on pp. 84–86).

Ma, L., Jia, X., Sun, Q., Schiele, B., Tuytelaars, T., and Van Gool, L. (2017b). Pose Guided Person Image Generation. In: *Advances in Neural Information Processing Systems 30 (NIPS)*, pp. 406–416 (cit. on pp. 16, 55, 57, 62, 66, 67, 75–78, 84–86, 96, 103, 104, 106, 107).

Magnenat-Thalmann, N. and Thalmann, D. (2005). Virtual humans: Thirty years of research, what next? In: *Visual Computer* **21** (12), pp. 997–1015 (cit. on p. 2).

Magnenat-Thalmann, N. and Thalmann, D. (2006). Handbook of Virtual Humans. Ed. by N. Magnenat-Thalmann and D. Thalmann. John Wiley & Sons, pp. 1–443 (cit. on p. 1).

Mansimov, E., Parisotto, E., Ba, J. L., and Salakhutdinov, R. (2015). Generating Interpretable Images with Controllable Structure. In: *arXiv preprint*, pp. 1–12 (cit. on p. 20).

Marcard, T. von, Rosenhahn, B., Black, M. J., and Pons-Moll, G. (2017). Sparse Inertial Poser: Automatic 3D Human Pose Estimation from Sparse IMUs. In: *Computer Graphics Forum* **36** (2), pp. 349–360 (cit. on p. 82).

Massiceti, D., Siddharth, N., Dokania, P. K., and Torr, P. H. S. (2018). FlipDial: A Generative Model for Two-Way Visual Dialogue. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 13, 83).

Massive Software (2017). Massive Software – Simulating Life. URL: http://www.massivesoftware.com/ (visited on 12/04/2018) (cit. on p. 6).

McKinsey&Company (2017). *The State of Fashion 2018*. Tech. rep. (cit. on p. 7).

McKinsey&Company (2018). *The State of Fashion 2019*. Tech. rep. (cit. on p. 7).

Mescheder, L., Nowozin, S., and Geiger, A. (2017). Adversarial Variational Bayes: Unifying Variational Autoencoders and Generative Adversarial Networks. In: *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pp. 2391–2400 (cit. on pp. 13, 20, 56).

Metaxas, B. R. · .R.K. · . D. (2008). Human Motion. Ed. by B. Rosenhahn, R. Klette, and D. Metaxas. Vol. 36. Computational Imaging and Vision. Springer (cit. on p. 11).

Microsoft (2018a). Kinect. URL: https://en.wikipedia.org/wiki/Kinect (visited on 12/03/2018) (cit. on p. 5).

Microsoft (2018b). Microsoft HoloLens | The leader in mixed reality technology. URL: https://www.microsoft.com/en-us/hololens (visited on 12/05/2018) (cit. on p. 8).

Mikolajczyk, K., Schmid, C., and Zisserman, A. (2004). Human Detection Based on a Probabilistic Assembly of Robust Part Detectors. In: *8th European Conference on Computer Vision (ECCV)*, pp. 69–82 (cit. on p. 29).

Moeslund, T. B. and Granum, E. (2001). A survey of computer vision-based human motion capture. In: *Computer Vision and Image Understanding* **81** (3), pp. 231–268 (cit. on pp. 1, 11).

Moeslund, T. B., Hilton, A., and Krüger, V. (2006). A survey of advances in vision-based human motion capture and analysis. In: *Computer Vision and Image Understanding* **104** (2-3), pp. 90–126 (cit. on pp. 1, 11).

Moeslund, T. B., Hilton, A., Krüger, V., and Sigal, L., eds. (2011). Visual Analysis of Humans. Springer (cit. on pp. 1, 11, 26, 82).

NASA (1995). *Space Flight Human-System Standard Volume 1*. Tech. rep. NASA-STD-3001. National Aeronautics and Space Administration - NASA (cit. on pp. 33, 62).

Neal, R. M. and Hinton, G. E. (1998). A View of the Em Algorithm that Justifies Incremental, Sparse, and other Variants. In: *Learning in Graphical Models*. Springer, pp. 355–368 (cit. on p. 21).

Neverova, N., Alp Güler, R., and Kokkinos, I. (2018). Dense Pose Transfer. In: *15th European Conference on Computer Vision (ECCV)*, pp. 128–143 (cit. on p. 86).

Newell, A., Yang, K., and Deng, J. (2016). Stacked hourglass networks for human pose estimation. In: *14th European Conference on Computer Vision (ECCV)*, pp. 483–499 (cit. on pp. 29–31, 33, 50, 54, 62, 67).

NVIDIA (2018). Artificial Intelligence Computing Leadership from NVIDIA. URL: https://www.nvidia.com/ (visited on 12/05/2018) (cit. on p. 5).

Obama White House Flickr (2010). President Barack Obama. URL: https://www.flickr.com/photos/obamawhitehouse/4921383047/ (visited on 12/27/2018) (cit. on p. 118).

Obdrzalek, S., Kurillo, G., Ofli, F., Bajcsy, R., Seto, E., Jimison, H., and Pavel, M. (2012). Accuracy and robustness of Kinect pose estimation in the context of coaching of elderly population. In: *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, pp. 1188–1193 (cit. on p. 6).

Ohn-Bar, E. and Trivedi, M. M. (2016). Looking at Humans in the Age of Self-Driving and Highly Automated Vehicles. In: *IEEE Transactions on Intelligent Vehicles* **1** (1), pp. 90–104 (cit. on pp. 5, 6).

OptiTrack (2018). OptiTrack - Motion Capture Systems. URL: https://optitrack.com/ (visited on 12/03/2018) (cit. on p. 3).

Organic Motion (2006). Organic Motion - Tracklab. URL: https://tracklab.com.au/organic-motion/ (visited on 12/05/2018) (cit. on p. 3).

OxSight (2018). OxSight. URL: https://www.oxsight.co.uk/ (visited on 12/05/2018) (cit. on p. 7).

Pantic, M., Pentland, A., Nijholt, A., and Huang, T. S. (2006). Human Computing and Machine Understanding of Human Behavior: A Survey. In: *Artifical Intelligence for Human Computing*. Vol. 4451 LNAI. Springer, pp. 47–71 (cit. on pp. 3, 111).

Peters, C. and Ennis, C. (2009). Modeling Groups of Plausible Virtual Pedestrians. In: *IEEE Computer Graphics and Applications* **29** (4), pp. 54–63 (cit. on p. 6).

Pfister, T., Charles, J., and Zisserman, A. (2015). Flowing convnets for human pose estimation in videos. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1913–1921 (cit. on pp. 12, 28).

Pishchulin, L., Andriluka, M., Gehler, P., and Schiele, B. (2013). Strong appearance and expressive spatial models for human pose estimation. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 3487–3494 (cit. on pp. 29, 31, 47, 50).

Pishchulin, L., Insafutdinov, E., Tang, S., Andres, B., Andriluka, M., Gehler, P., and Schiele, B. (2015). DeepCut: Joint Subset Partition and Labeling for Multi Person Pose Estimation. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 12, 29, 30, 44).

Pons-Moll, G. and Rosenhahn, B. (2011). Model-based pose estimation. In: *Visual analysis of humans*. Springer, pp. 139–170 (cit. on p. 54).

Poppe, R. (2007). Vision-based human motion analysis: An overview. In: *Computer Vision and Image Understanding* **108** (1-2), pp. 4–18 (cit. on p. 11).

Poppe, R. (2010). A survey on vision-based human action recognition. In: *Image and Vision Computing* **28** (6), pp. 976–990 (cit. on pp. 3, 111).

Prisacariu, V. A. and Reid, I. (2011). Robust 3D hand tracking for human computer interaction. In: *Face and Gesture 2011*, pp. 368–375 (cit. on p. 7).

Pumarola, A., Agudo, A., Sanfeliu, A., and Moreno-Noguer, F. (2018). Unsupervised Person Image Synthesis in Arbitrary Poses. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8620–8628 (cit. on p. 87).

Quan, T. M., Hildebrand, D. G. C., and Jeong, W.-K. (2016). FusionNet: A deep fully residual convolutional neural network for image segmentation in connectomics. In: *CoRR* **abs/1612.0** (cit. on p. 57).

Rafi, U., Leibe, B., Gall, J., and Kostrikov, I. (2016). An Efficient Convolutional Network for Human Pose Estimation. In: *Procedings of the British Machine Vision Conference (BMVC)*, pp. 109.1–109.11 (cit. on pp. 12, 28, 50).

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic Backpropagation and Approximate Inference in Deep Generative Models. In: *International Conference on Machine Learning (ICML)*, pp. 1278–1286 (cit. on pp. 12, 20, 54, 56, 58, 82, 87).

Romero, J., Labs, B., Martin, X., Laptev, I., Mahmood, N., Black, M. J., and Schmid, C. (2017). Learning from Synthetic Humans. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 109–117 (cit. on p. 117).

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In: *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234–241 (cit. on p. 86).

Rosales, R. and Sclaroff, S. (2000a). Inferring body pose without tracking body parts. In: *2000 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 721–727 (cit. on p. 20).

Rosales, R. and Sclaroff, S. (2000b). Specialized mappings and the estimation of human body pose from a single image. In: *Proceedings - Workshop on Human Motion, HUMO 2000*, pp. 19–24 (cit. on p. 20).

Rosales, R., Siddiqui, M., Alon, J., and Sclaroff, S. (2001). Estimating 3D body pose using uncalibrated cameras. In: *2001 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8 (cit. on p. 20).

Rosales, R. and Sclaroff, S. (2000c). Learning and synthesizing human body motion and posture. In: *Proceedings - 4th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2000*, pp. 506–511 (cit. on p. 20).

Sapp, B. and Taskar, B. (2013). MODEC: Multimodal decomposable models for human pose estimation. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3674–3681 (cit. on p. 43).

Schaffalitzky, F. (2016). Human interaction with unmanned aerial vehicles. URL: https://patents.google.com/patent/US9921579B1/en (cit. on p. 5).

Schulman, J., Heess, N., Weber, T., and Abbeel, P. (2015). Gradient Estimation Using Stochastic Computation Graphs. In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 3528–3536 (cit. on pp. 23, 83, 113).

Seemann, E., Kai Nickel, and Stiefelhagen, R. (2004). Head pose estimation using stereo vision for human-robot interaction. In: *Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 626–631 (cit. on p. 82).

SenseTime (2018). Body Feature Point. URL: https://www.sensetime.com/en/Technology/face.html (visited on 12/03/2018) (cit. on p. 4).

Shan, Q., Adams, R., Curless, B., Furukawa, Y., and Seitz, S. M. (2013). The Visual Turing Test for Scene Reconstruction. In: *2013 International Conference on 3D Vision*, pp. 25–32 (cit. on p. 10).

Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., and Blake, A. (2011). Real-time human pose recognition in parts from single depth images. In: *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Vol. 56. 1, pp. 1297–1304 (cit. on pp. 5, 82).

Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., Cook, M., and Moore, R. (2013). Real-time human pose recognition in parts from single depth images. In: *Communications of the ACM* **56** (1), p. 116 (cit. on p. 5).

Siarohin, A., Sangineto, E., Lathuiliere, S., and Sebe, N. (2018). Deformable GANs for Pose-based Human Image Generation. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3408–3416 (cit. on pp. 66, 84, 86).

Siddharth, N., Paige, B., Meent, J.-W. van de, Desmaison, A., Goodman, N. D., Kohli, P., Wood, F., and Torr, P. H. S. (2017). Learning Disentangled Representations with Semi-Supervised Deep Generative Models. In: *Advances in Neural Information Processing Systems 30 (NIPS)*, pp. 5925–5935 (cit. on pp. 23, 56, 83, 88, 97, 113).

Sigal, L. (2014). Human Pose Estimation. In: *Computer Vision*. Ed. by K. Ikeuchi. Vol. 11. Springer, pp. 362–370 (cit. on pp. 1, 9).

Sigal, L. and Black, M. J. (2010). Guest editorial: State of the art in image- and video-based human pose and motion estimation. In: *International Journal of Computer Vision* **87** (1-2), pp. 1–3 (cit. on p. 11).

Singh, A., Patil, D., and Omkar, S. (2018). Eye in the Sky: Real-time Drone Surveillance System (DSS) for Violent Individuals Identification using ScatterNet Hybrid Deep Learning Network. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 1629–1637 (cit. on pp. 5, 6).

Sminchisescu, C. (2006). 3D Human motion analysis in monocular video techniques and challenges. In: *Proceedings - IEEE International Conference on Video and Signal Based Surveillance 2006, AVSS 2006*. Springer, pp. 185–211 (cit. on p. 9).

Soatto, S. and Bissacco, A. (2008). Visual Detection and Classification of Humans, Their Pose and Their Motion. In: *Homeland security*. Artech House (cit. on p. 20).

Sogou (2017). Sogou - Company Profile. URL: http://ir.sogou.com/ (visited on 12/05/2018) (cit. on p. 4).

Sohn, K. (2015). Learning Structured Output Representation using Deep Conditional Generative Models. In: *Advances in Neural Information Processing Systems 28 (NIPS)*, pp. 3483–3491 (cit. on pp. 55, 56, 59, 112).

Starck, J. and Hilton, A. (2007). Surface capture for performance-based animation. In: *IEEE Computer Graphics and Applications* **27** (3), pp. 21–31 (cit. on p. 2).

Takahashi, K., Uemura, T., and Ohya, J. (2000). Neural-network-based real-time human body posture estimation. In: *2000 IEEE Signal Processing Society Workshop on Neural Networks for Signal Processing*, pp. 477–486 (cit. on p. 20).

Taylor, G. W., Sigal, L, Fleet, D. J., and Hinton, G. E. (2010a). Dynamical binary latent variable models for 3D human pose tracking. In: *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 631–638 (cit. on p. 54).

Taylor, G. W., Fergus, R., Williams, G., Spiro, I., and Bregler, C. (2010b). Pose-sensitive embedding by nonlinear nca regression. In: *Advances in Neural Information Processing Systems 23 (NIPS)*, pp. 2280–2288 (cit. on pp. 12, 20, 26).

The Captury (2018). The Captury - Markerless motion capture technology. URL: http://thecaptury.com/ (visited on 12/05/2018) (cit. on p. 4).

Theis, L., Oord, A. van den, and Bethge, M. (2015). A note on the evaluation of generative models. In: *International Conference on Learning Representations (ICLR)* (cit. on p. 67).

Thies, J., Zollhofer, M., Stamminger, M., Theobalt, C., and NieBner, M. (2016). Face2Face: Real-Time Face Capture and Reenactment of RGB Videos. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2387–2395 (cit. on pp. 4, 82).

Tompson, J., Jain, A., LeCun, Y., and Bregler, C. (2014). Joint Training of a Convolutional Network and a Graphical Model for Human Pose Estimation. In: *Advances in Neural Information Processing Systems 27 (NIPS)*, pp. 1799–1807 (cit. on pp. 29, 30, 50, 54, 62).

Tompson, J., Goroshin, R., Jain, A., LeCun, Y., and Bregler, C. (2015). Efficient object localization using Convolutional Networks. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 648–656 (cit. on pp. 29, 30, 50).

Toshev, A. and Szegedy, C. (2014). DeepPose: Human Pose Estimation via Deep Neural Networks. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1653–1660 (cit. on p. 28).

Trumble, M., Gilbert, A., Hilton, A., and Collomosse, J. (2018). Deep Autoencoder for Combined Human Pose Estimation and Body Model Upscaling. In: *15th European Conference on Computer Vision (ECCV)*, pp. 800–816 (cit. on pp. 12, 86).

Tu, Z. (2008). Auto-context and its application to high-level vision tasks. In: *2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1744–1757 (cit. on p. 30).

Tulyakov, S., Liu, M.-Y., Yang, X., and Kautz, J. (2018). MoCoGAN: Decomposing Motion and Content for Video Generation. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1526–1535 (cit. on pp. 57, 87).

University of Surrey (2018). Surrey to build world's first translation system for British Sign Language. URL: https://www.surrey.ac.uk/news/surrey-build-worlds-first-translation-system-british-sign-language (visited on 12/05/2018) (cit. on p. 7).

Valenza, E, Simion, F, Cassia, V. M., and Umiltà, C (1996). Face preference at birth. In: *Journal of experimental psychology. Human perception and performance* **22** (4), pp. 892–903 (cit. on p. 10).

Vicon (2018). Vicon Motion Systems. URL: http://www.vicon.com (cit. on p. 3).

Vicon Motion Systems (2018). Create subjects and props - Shogun 1.2 Documentation. URL: https://docs.vicon.com/display/Shogun12/Create+subjects+and+props (visited on 12/15/2018) (cit. on p. 4).

Walker, J., Marino, K., Gupta, A., and Hebert, M. (2017). The Pose Knows: Video Forecasting by Generating Pose Futures. In: *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 3352–3361 (cit. on pp. 54, 57, 84, 87, 116).

Wan, C., Probst, T., Van Gool, L., and Yao, A. (2017). Crossing nets: Dual generative models with a shared latent space for hand pose estimation. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 13, 20, 21, 56).

Wang, L., Hu, W., and Tan, T. (2003). Recent developments in human motion analysis. In: *Pattern Recognition* **36** (3), pp. 585–601 (cit. on p. 11).

Wang, T.-C., Liu, M.-Y., Zhu, J.-Y., Yakovenko, N., Tao, A., Kautz, J., and Catanzaro, B. (2018). Video-to-Video Synthesis. In: *Advances in Neural Information Processing Systems 31 (NIPS)*, pp. 1152–1164 (cit. on pp. 6, 120).

Wang, Z., Bovik, A., Sheikh, H., and Simoncelli, E. (2004). Image Quality Assessment: From Error Visibility to Structural Similarity. In: *IEEE Transactions on Image Processing* **13** (4), pp. 600–612 (cit. on p. 67).

Wang, Z., Merel, J., Reed, S., Wayne, G., Freitas, N. de, and Heess, N. (2017). Robust Imitation of Diverse Behaviors. In: *Advances in Neural Information Processing Systems 30 (NIPS)* (cit. on p. 83).

Wei, S.-E., Ramakrishna, V., Kanade, T., and Sheikh, Y. (2016). Convolutional Pose Machines. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4724–4732 (cit. on pp. 12, 28, 30, 31, 33, 44, 47, 54, 62, 82).

Xinhua News Agency (2018). World's first AI news anchor makes his China debut. URL: http://www.xinhuanet.com/english/2018-11/08/c_137591813.htm (visited on 12/04/2018) (cit. on p. 4).

Yang, W., Li, S., Ouyang, W., Li, H., and Wang, X. (2017). Learning Feature Pyramids for Human Pose Estimation. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1290–1299 (cit. on p. 97).

Yang, Y. and Ramanan, D. (2011). Articulated pose estimation with flexible mixtures-of-parts. In: *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1385–1392 (cit. on pp. 29, 43, 61, 67, 96).

Zhang, W., Zhu, M., and Derpanis, K. G. (2013). From actemes to action: A strongly-supervised representation for detailed action understanding. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2248–2255 (cit. on p. 54).

Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., and Torr, P. H. (2015). Conditional random fields as recurrent neural networks. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1529–1537 (cit. on pp. 28, 30, 37, 40, 41, 51, 112).

Zyda, M. and Michael (2005). From visual simulation to virtual reality to games. In: *Computer* **38** (9), pp. 25–32 (cit. on p. 8).