Articulated human pose estimation in images and video

Aichun Zhu

Thèse
de doctorat
de l'UTT

Aichun ZHU

Articulated Human Pose Estimation in Images and Video

Spécialité :
Optimisation et Sûreté des Systèmes

2016TROYO013

Année 2016

utt
université de technologie
Troyes

THESE

pour l'obtention du grade de

DOCTEUR de l'UNIVERSITE DE TECHNOLOGIE DE TROYES

Spécialité : OPTIMISATION ET SURETE DES SYSTEMES

présentée et soutenue par

Aichun ZHU

le 30 mai 2016

Articulated Human Pose Estimation in Images and Video

JURY

M. M. DAOUDI	PROFESSEUR TELECOM LILLE 1	Président
M. A. CHEROUAT	PROFESSEUR DES UNIVERSITES	Directeur de thèse
M. F. DORNAIKA	PROFESSOR	Rapporteur
M. H. SAHBI	CHARGE DE RECHERCHE CNRS - HDR	Rapporteur
M. H. SNOUSSI	PROFESSEUR DES UNIVERSITES	Directeur de thèse

Acknowledgments

I would like to express my gratitude to all those who helped me during my doctoral studies and the writing of this thesis.

First and foremost, I would like to express my gratitude to my supervisors, Prof. Hichem Snoussi and Prof. Abel CHEROUAT, for their constant encouragement and guidance of my research. They provides me with an excellent atmosphere for doing research through the three and a half years. I learn from them not only the research methods and knowledge, but also the attitude to life and research.

A special thank goes to reviewers: Prof. Dornaika, Prof. Sahbi and Prof. Daoudi who kindly agreed to serve as reviewers of my PhD thesis.

I wish to express my gratitude to China Scholarship Council (CSC) and University of Technology of Troyes (UTT) for the financial support during these three and a half years on France.

I would like to express my sincere gratitude to Mr. Xiaolu Gong, Ms. Ling Gong in University of Technology of Troyes, and Tian Wang in Beihang University, for their valuable comments on my research. Thanks the secretaries of the pôle ROSAS Ms. Veronique Banse, Ms. Bernadette Andre and Ms. Marie-José Rousselet, and the secretaries of the doctoral school: Ms. Isabelle Leclercq, Ms. Pascale Denis and Ms. Therese Kazarian, for their help throughout my PhD study.

I want to thank my friends in UTT for their valuable supports and aids, and all my other friends in France or in China. Special thanks to Lei Qin, Shijie Zhu, Xiaowei Lv, Tong Qiao, Xiaohui Zhang, Yongwen Liu, Wenliang Wang and Xiaoyi Chen, they always help me and give me their best suggestions.

Lastly, I offer sincere thanks to my parents, my brother, my girlfriend and all my family members, for their loving considerations and great confidence in me all through these years.

Résume

L'estimation de la pose du corps humain est un problème difficile en vision par ordinateur et les actions de toutes les difficultés de détection d'objet. Cette thèse se concentre sur les problèmes de l'estimation de la pose du corps humain dans les images ou vidéo, y compris la diversité des apparences, les changements de scène et l'éclairage de fond de confusion encombrement. Pour ésoudre ces problèmes, nous construisons un modèle robuste comprenant les éléments suivants. Tout d'abord, les méthodes top-down et bottom-up sont combinés à l'estimation pose humaine. Nous étendons le modèle structure picturale (PS) de coopérer avec filtre à particules recuit (APF) pour robuste multi-vues estimation de la pose. Deuxièmement, nous proposons plusieurs parties de mélange à base (MMP) modèle d'une partie supérieure du corps pour l'estimation de la pose qui contient deux étapes. Dans la phase de pré-estimation, il y a trois étapes: la détection du haut du corps, catégorie estimation du modèle pour le haut du corps, et la sélection de modèle complet pour pose estimation. Dans l'étape de l'estimation, nous abordons le problème d'une variété de poses et les activités humaines. Enfin, le réseau de neurones à convolution (CNN) est introduit pour l'estimation de la pose. Un Local Multi-résolution réseau de neurones à convolution (LMR-CNN) est proposé pour apprendre la représentation pour chaque partie du corps. En outre, un modèle hiérarchique sur la base LMR-CNN est définie pour faire face à la complexité structurelle des parties de branche. Les résultats expérimentaux démontrent l'efficacité du modèle proposé.

Les mots clés: Vision par ordinateur; Détection du signal; Posture; Machines à vecteurs de support ; Réseaux neuronaux (informatique).

Abstract

Human pose estimation is a challenging problem in computer vision and shares all the difficulties of object detection. This thesis focuses on the problems of human pose estimation in still images or video, including the diversity of appearances, changes in scene illumination and confounding background clutter. To tackle these problems, we build a robust model consisting of the following components. First, the top-down and bottom-up methods are combined to estimate human pose. We extend the Pictorial Structure (PS) model to cooperate with Annealed Particle Filter (APF) for the robust multi-view pose estimation. Second, we propose an upper body based Multiple Mixture Parts (MMP) model for human pose estimation that contains two stages. In the pre-estimation stage, there are three steps: upper body detection, model category estimation for upper body, and full model selection for pose estimation. In the estimation stage, we address the problem of a variety of human poses and activities. Finally, a Deep Convolutional Neural Network (DCNN) is introduced for human pose estimation. A Local Multi-Resolution Convolutional Neural Network (LMR-CNN) is proposed to learn the representation for each body part. Moreover, a LMR-CNN based hierarchical model is defined to meet the structural complexity of limb parts. The experimental results demonstrate the effectiveness of the proposed deep learning approach for pose estimation.

Keywords: Computer vision; Signal detection; Posture; Support vector machines; Neural networks (Computer science).

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Contents

The goal of computer vision is to model and replicate the way humans see. This includes reasoning about interactions between humans and objects. To enable such high-level reasoning, a machine must first have access to interpretations of posture obtained from visual data. The task of determining postures of a person in images or videos is referred to as pose estimation.

1.1 Problem formulation

Human pose estimation is a fundamental problem in computer vision and has numerous important applications such as sports, action recognition and human-computer interaction. The articulated pose estimation is formulated as follows: Given an image which contains a human body and an articulation model (a model of the body), one has to describe the current body configuration in terms of a set of limbs and rotational joints that connect them into a tree structure (See Fig. 1.1). Since the early methods of Hogg [2] and O' Rourke and Badler [3] in the 1980s, the estimation of articulated human pose has received much attention. In the past few decades, we have witnessed the evolution of estimating the articulated pose of a person in controlled, often indoor environments. Despite many years of research, however, articulated pose estimation remains a challenging problem in unconstrained scenario. It shares all the difficulties of object detection, such as the diversity of appearances, changes in scene illumination and camera view-point, confounding background clutter, and occlusion [4, 5].

Among the most significant challenges are: (1) cluttered Background, (2) variability of clothing in images, (3) variability in lighting conditions, (4) occlusion and self-occlusion in the scene, (5) motion Blur, (6) high dimensionality of the pose, etc.

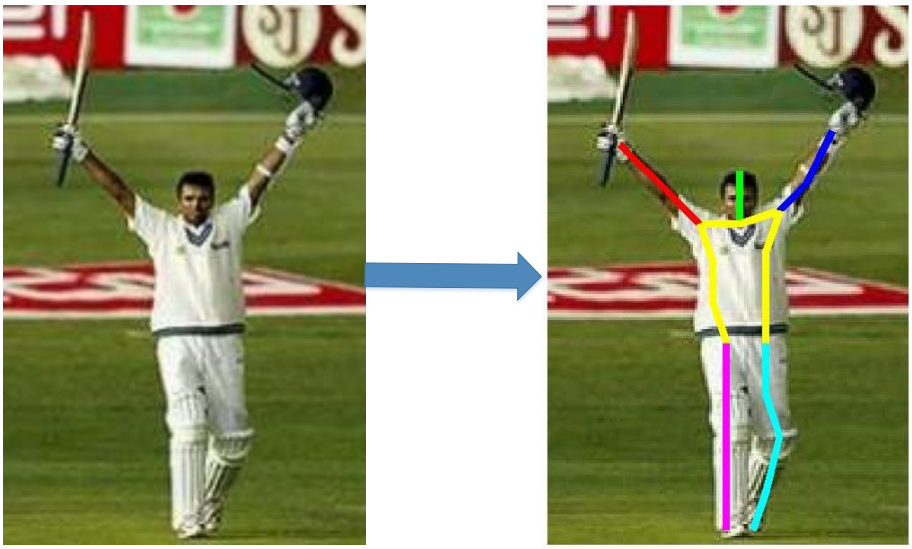


Figure 1.1: Articulated human pose estimation.

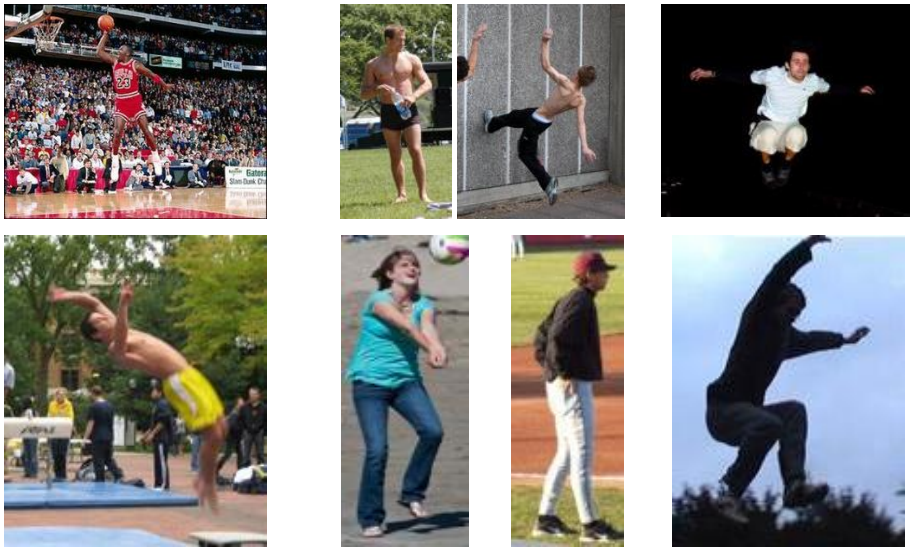
1.2 Challenges

Cluttered Background: The presence of background clutter is a common problem in object detection. We address this problem in the context of estimating articulated objects from still images. Images of people are seen everywhere. It is likely that the background of an image contains very irregular and complex objects. These background objects tend to distract from foreground objects that can lead to false-positive results. As shown in the top row of Fig. 1.2 (a), this image contains many people in the background. In the bottom row of Fig. 1.2 (a), there are different objects in the background including trees, people and building.

Clothing: People wear a wide variety of types of clothes, often with various textures and colors. As shown in the top row of Fig. 1.2 (b), the two images depict a man wearing tight shorts or trousers only. In the bottom row, two images show more typical clothing objects.

Lighting conditions: Images can be taken at any time, in any location leading to different lighting conditions. For example, due to the existence of the shadows, the overall brightness of the same scene may be different. Moreover, cameras tend to automatically adjust exposure for the environmental lighting that may be lead to over-exposed or under-exposed problems. The top row of Fig. 1.2 (c) presents a person with invisible arms that due to the under-exposed dark background, while the bottom row shows an over-exposed example.

Occlusion and self-occlusion: In natural scenes, one of the most common problems is occlusion or self-occlusion of objects. In the case of articulated human



(a) Cluttered background (b) Varied clothing (c) Varying illumination

Figure 1.2: Challenges for pose estimation.

(a) Occlusion and self-occlusion (b) Motion blur

Figure 1.3: Challenges for pose estimation.

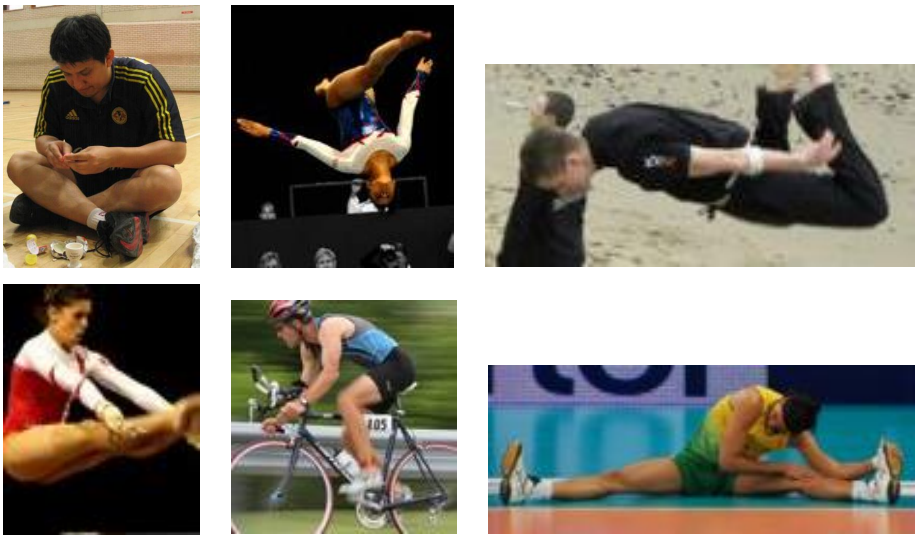


Figure 1.4: Challenges for pose estimation.

pose estimation, the occlusion occurs often as people in groups and the self-occlusion occurs as different viewpoints of objects. Examples of this can be seen in Fig. 1.3 (a). The top row shows the images of the self-occluded objects that are partially occluded by themselves. The bottom row shows objects of interest are occluded by any other objects in the same scenes.

Motion Blur: Motion blur means the apparent streaking of rapidly moving objects due to relatively long exposure times on consumer cameras. The extreme motion blur makes the inference of human body part extremely challenging. As shown in Fig. 1.3 (b), the limb appearances of the players are motion-blurred and have no clearly solid regions.

High-dimensional pose space: Articulated objects like human bodies can take on a large variety of possible body poses, some are shown in Fig.1.4. Generally, human body contains 13 major joints: 1 neck, 2 shoulders, 2 elbows, 2 wrists, 2 hips, 2 knees and 2 ankles. Each joint has one or more degrees of freedom. In a constrained scenario like walking, this large space can be reduced dramatically. However, in an unconstrained scenario, the pose space can be complex extremely. No knowledge of scenario is used in human pose estimation, which is a really challenging task.

1.3 Summary of the Thesis

The main goal of this thesis is to address the problem of pose estimation in computer vision. In particular, the problems of bottom-up pose estimation in images are studied. In the following, the main contributions of this thesis and the layout are briefly summarized.

1.3.1 Main contributions

The contributions of this thesis can be divided into three parts.

Firstly, a new approach is proposed to estimate articulated human pose based on foreground learning by Flexible Mixtures of Parts (FMP) model [6], which has shown strong ability to detect and estimate human motion from still images. In our work, we make use of the sequence to learn and subtract the background, and then jointly track and detect body parts in multiple views. Part models are trained based on PARSE dataset [7]. We evaluate our approach on the HumanEva-II dataset, which is a standard benchmark for 3D pose estimation.

Secondly, we address the problem of building a better model in a pictorial structure framework. An upper body based Multiple Mixture Parts model (MMP) is proposed, which contains two stages. The first stage includes three steps: upper body detection, model category estimation and model selection. Different categories are proposed for defining upper body models, and each upper body model corresponds to one mixture model. Each mixture model is trained on the dataset in which all the images share the same upper body category. As for model selection, there are two-level models. One is the local part model and the other is the combined model. The combined models are defined between pairs of joints, while local part models are built between each joint and middle point. Two-level models are used to achieve a more accuracy estimation. The first stage only categorizes poses, so it is referred to as the pre-estimation stage, while the other is the estimation stage. In the second stage, the MMP model is proposed to join different mixture models. Each mixture model in MMP corresponds to one upper body category. Different mixture models not only have different kinematic constraints, but also have different numbers of part models.

Finally, Convolutional Neural Networks (CNNs) are introduced into our work. A deep hierarchical model based on Local Multi-Resolution Convolutional Neural Network (LMR-CNN) is proposed for articulated pose estimation. The deep hierarchical model contains two-level structure: limb parts and joint parts. Thus, an extra network is used to training limb parts and is called *Limbnet*. Concerning the convolutional neural network, a LMR-CNN is proposed to train and learn the representation of each body parts by combining different levels of part contexts.

1.3.2 Outline

In Chapter 2, the state of the art of pose estimation is introduced. First, the top-down pose estimation methods are presented. Then, a focus is taken towards more popular approaches and specifically those built upon the framework of pictorial structure model. Finally, the very recent CNNs based methods for pose estimation are discussed.

In Chapter 3, the top-down and bottom-up methods are combined to estimate human pose. The annealed particle filter is top-down method while Flexible Mixtures of Parts (FMP) model is a bottom-up solution.

In Chapter 4, an upper body based Multiple Mixture Parts model (MMP) is proposed for human pose estimation. Different categories are proposed for defining upper body models, and each upper body model corresponds to one mixture model. Different mixture models are combined together to build a full MMP.

In Chapter 5, first relative mixtures are introduce in FMP. Then a Local Multi-Resolution Convolutional Neural Network (LMR-CNN) is proposed to learn the representation for each body part. Finally, a LMR-CNN based hierarchical model is defined to cope with the structural complexity of limb parts.

Chapter 6 concludes this PhD thesis and discusses the future work.

1.4 Publications

Most of the material presented in this thesis appears in the following publications that represent original work, of which the author has been the main contributor.

Journal articles

1. **A. Zhu**, H. Snoussi, T. Wang, and A. Cherouat, "Human pose estimation with multiple mixture parts model based on upper body categories," *Journal of Electronic Imaging*, vol. 24, no. 4, p. 043021, 2015.

2. **A. Zhu**, H. Snoussi, T. Wang, and A. Cherouat, "Human Pose Estimation via Deep Hierarchical Model based on Local Multi-resolution Convolutional Neural Network," *Neurocomputing.* (Under review)

Conference papers

1. **A. Zhu**, H. Snoussi, and A. Cherouat, "Articulated human motion tracking with foreground learning," in *European Signal Processing Conference*, 2014, pp. 366-370.

2. **A. Zhu**, H. Snoussi, and A. Cherouat, "Articulated pose estimation via multiple mixture parts model," in *IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2015.

3. T. Wang, J. Chen, **A. Zhu**, H. Snoussi, "Event analysis based on multiple video sensors for cooperative environment perception," *International Conference in Informatics and Computing.* Dec. 2015

4. Y. Zhang, T. Wang, **A. Zhu**, J. Chen, H. Snoussi, "Detection of Abnormal Event in Complex Situations Using Strong Classifier Based on BP Adaboost," *International Conference on Intelligent Computation .* (Under review)

Chapter 2

State of the art of pose estimation

Contents

2.1 Overview

Pose estimation is the focus of this thesis. A wide variety of research studies have been implemented regarding human pose estimation, due to many applications based on analyzing human pose in images and video. Looking back at past work on articulated human pose estimation, three distinct categories emerge: early methods, pictorial structure-based methods and poselets, deep convolutional neural networks. These methods are discussed in the next sections.

2.2 Top-down pose estimation

Top-down approaches match a direct model with the image observation. The direct model means that a priori human model is used as the model representing the observed object. This human model is then continuously updated by the observations. Hence, it provides any desired information at any time. The models used in direct model based methods are generally very detailed. They are explicitly identified within a computer program and are intensively used during the observation. One of the most important benefits of introducing a human model is the ability to handle occlusion by various kinematic constrains. A number of joints and the sticks are used to represent a direct human model, and these joints are connected by the sticks [8, 9].

2.2.1 Earlier work

The most of earliest pose estimation algorithms addressed the problem with an explicit geometric representation of human shape and kinematic structure to reconstruct pose. The human model is concretely represented by a state space where each axis represents a degree of freedom of a joint. Hence, one point expresses one pose in the state space corresponding to the points in the image. The problem is how to use the state representation and how to relate image data to pose data. A general approach addresses this problem employing an analysis-by-synthesis methodology to optimize the similarity between the model projection and the observed image data [2]. Thus, this model-based analysis-by-synthesis methodology contains two parts: the first is the prediction of the pose; then, the predicted model is used for the comparison between the model projection and observed images.

Clearly the predicted state space describes a large number of possible poses which makes it unreasonable for matching the observed data. Thus, the idea of introducing constraints is used to prune the state space. In the pioneering work [2, 10], Hogg has introduced an approach for determining the 3D positions and postures of walking persons from 2D images. Moeslund and Granum [11] directly partition the state space into legal and illegal regions.

Another method to reduce the state space is the use of a known cyclic motion (e.g. running and walking). In [12], the person gait parallel to the image plane is considered. All pose parameters are estimated by using a cyclic motion model of the person gait. This is an efficient pruning for the cyclic motion. Ong and Gong [13] map training data into the state space and use the hierarchical Principal Component Analysis (PCA) to extract a subspace to estimate the degree of ambiguity in the 2D cues. In [14], Pavlović et al. take this idea a step farther by learning dynamic models from the observed state space trajectories. More efficiently, Moeslund and Granum [11] reduce the dimensionality of the state space by representing the human model with different degrees of freedom of a structural model.

Concerning the comparison between the model projection and observed images, Hogg [2, 10] proposed to use image subtraction to obtain the edges of a human, and

compared edges from image and human model. A more sophisticated system in [12] combined edge segments with a specific motion model to obtain a more effective result. Wachter and Nagel [15] use both edges and regional information in the matching of image and human model.

A silhouette is a region-based data and has the advantage over edges of being robust to noise. Kameda et al. [16] compute the similarity of the silhouette between the image and the human model. In the work by Hu et al. [17], the similarity of the silhouette is computed with a local match strategy which is based the positive and negative matching results. They also apply genetic algorithm to obtain better matching results. Furthermore, the structure-oriented Kalman filter is reserved in large morphologic scale to improve the matching accuracy. In [18–20] , the contour is used to calculate the similarity between the image and human model data.

2.2.2 Monocular pose estimation

Reconstruction of human pose from monocular image sequences is an important and challenging research field with numerous applications. In monocular human pose estimation, the kinematic constraints are typically employed in the human direct model [21,22]. In the work by Wachter and Nagel [22] the extended Kalman filter is used to estimate human pose with kinematic constraints. Sminchisescu and Triggs [23] have investigated the application of stochastic sampling to estimate monocular human pose. They use a robust cost metric combining robustly extracted optical flow, edge energy, motion boundaries and model priors for image matching. The inflated-covariance-scaled sampling is introduced to guide the particles and reduce the incorrect local minima. In the further research [24], they use simple kinematic reasoning to enumerate the potential forwards/backwards flips which cause visual ambiguities.

Probabilistic approaches using human body parts together with human kinematics have also been investigated for monocular human pose estimation. In [25], proposal maps are introduced to represent the estimated likelihood of body parts in 3D pose space with an explicit 3D model. A data-driven Markov Chain Monte Carlo approach (MCMC) is used to search the human pose space. MCMC was applied to estimate 3D poses from single images of sports players in a variety of complex poses, but it still suffers from high computational cost. Moeslund et al. [26, 27] employ a data driven sequential MCMC to estimate human pose. A part detector is used to locate the position of the hand in the image. This estimation is applied to correct the prediction and reduce the number of particles. Navaratnam et al. [28] propose a hierarchical part-based kinematic model for the upper-body pose estimation. The human body is treated as a collection of parts that are linked in a kinematic chain. Kinematic constraints in a collection of linked parts are represented hierarchically.

Top-down model-based single view pose estimation suffers from accumulation of errors. In case of ambiguity, such as self-occlusion, it has the high possibility of selecting the wrong pose. These errors make the pose recovery difficult.

2.2.3 Multiple view pose estimation

Reconstruction of human pose from multiple view images is more efficient solution for complex movements. This is used to overcome the problems of the single view pose estimation. Deterministic gradient descent based approaches are employed to estimate human pose in multiple view scenes. In [29], Delamarre and Faugeras estimate the motion of an articulated object in two or more fixed cameras considering the quality of the images in all views. Furthermore, the physical forces applied to each body part in a kinematic 3D human model. These forces guide the minimization of the differences between images and projected 3D model. Many work employed an analysis-by-synthesis methodology in deterministic gradient descent based approach for more complex motions. The work by Plänkers and Fua [30] use stereo and silhouette cues to handle complicated motions that involve self-occlusions. A common limitation of gradient descent approaches is the use of a state estimation based on Gaussian distribution. Thus, it is restricted to the unimodal probability distribution. In practice the pose estimation is usually a multimodal and non-Gaussian problem. To achieve more robust tracking, stochastic sampling strategies are employed to search of the pose state space.

Particle filter is stochastic technique for pose estimation and tracking. Particle filtering [31, 32] is one of the common approaches for human motion tracking, which used the pose in the current frame and a dynamic model to predict the next pose. Particle filter (PF) uses multiple predictions, obtained by drawing samples of pose and location prior, and then propagating them using the dynamic model by comparing them with the observed image data and calculating the likelihood. The pose prior is usually quite diffused but the likelihood function of the dynamic model may be very peaky, containing many local maxima which are difficult to account for in detail. The principal difficulty with the application of particle filter is the high dimensionality of the state space in pose estimation. Thus, the number of particles increases exponentially with dimensionality. In [33], MacCormick and Isard introduce the technique of partitioned sampling to reduce the dimensionality in the state space for efficient 2D pose estimation of articulated objects. There are two features of partitioned sampling in the field of articulated objects: first is the number of samples devoted to each partition can be varied for significant computational improvements; and secondly, that the number of likelihood evaluations can be halved by expressing the likelihood as a easily calculated function. Furthermore, it is a self-initialising and real-time system, and shows the robustness and accuracy for more complex interactive tasks. However, this approach is only applied to the hand tracking, and does not extend to whole-body pose estimation.

Annealed particle filter (APF) is proposed by Deutscher et al in [34, 35], and it is used to capture the makerless human motion in a multi-camera system. They combine a simulated annealing with particle filter which is shown to be effective at searching the high-dimensional configuration spaces in articulated pose estimation and body motion tracking. This approach uses a continuation principle to gradually introduce the influence of narrow peaks in the fitness function. The traditional

particle filter has the problem that it can be easily distracted by local maxima. In the annealed particle filter, the sparse particle set is able to move gradually towards the global maximum without being distracted by local ones. Furthermore, the improves and extends the APF in two ways. Firstly, a mechanism is implemented in the search space to achieve a soft partitioning. They propose a means to make the diffusion step in APF adaptive during annealing. This can lead to what can be interpreted as a soft hierarchical search strategy which automatically partitions the search space, and hence to further gains in efficiency. Second is that they introduce a crossover operator (similar to that found in genetic algorithms) into the particle filtering framework. They demonstrate that this operator improves the ability of the tracker to search the configuration spaces of articulated objects.

Sigal and Balan [36] present HumanEva dataset for quantitative evaluation of competing methods of articulated human pose. HumanEva is a standard benchmark for multi-view 3D human pose estimation in the laboratory setting. This dataset consists of HumanEva-I and HumanEva-II by a set of multi-view sequences. The dataset contains walking, jogging, hand gestures, throwing and catching a ball, and boxing action styles from three different subjects. They also present a baseline method for articulated object tracking. A relatively standard Bayesian framework is used for the optimization in the form of sequential importance resampling and APF. They combine the edge-based likelihood function, silhouette-based likelihood function and bi-directional silhouette-based likelihood function together in the posterior representation. APF has been widely used for articulated human motion tracking due to its ability to precisely estimate the statistics of multi-modal and non-Gaussian processes. However, the performance of annealed particle filter drops when the frame rate is lower or the motion is moving fast.

There are some work that combined stochastic search with gradient descent for local part estimation to recover full-body motion. Carranza et al. [37] demonstrate multi-view full-body pose estimation combining a deterministic grid search with gradient descent. For each body part a grid search first finds the set of valid poses by minimizing the overlap between the observed 2D shapes and the projections of the model. A fitness function is used to evaluate the valid poses to find the best pose. Then this best pose is refined by gradient descent optimization. Although their method does not require specific 3D reconstruction, an exact body model and segmentation of the person in the different view points is crucial to reach a meaningful measurement. In related work Kehl et al. [38] propose Stochastic Meta Descent (SMD) with stochastic sampling for full-body pose estimation form multiple views. They introduces the SMD optimization which allows the approach to avoid convergence to local minima.

2.2.4 Motion priors

There are many research focus on the motion prior models that are derived from training data from single or multiple view. Most statistical motion models can only be used for specific movements (walking and jogging) with specified constraints.

When only a single class of movements is regarded, motion priors can help to improve the performance in pose estimation [39].

Sidenbladh et at. [40–42] combine stochastic sampling with a strong learned motion priors of specific movements. The samples are propagated in a particle filter framework by the dynamics of the sample. An exemplar-based approach is used in [42] where the motion examples are indexed to show possible movement directions. In [43], the human appearance and the image motion priors are combined together to model the likelihood of observing various image for a given movement. These methods employ an analysis-by-synthesis methodology to the human motion reconstruction. Similarly, a hierarchical Principal Component Analysis (PCA) is used to encode geometry and kinematics and Hidden Markov model (HMM) is used to represent human dynamics for monocular pose estimation [44]. Agarwal and Triggs [45] cluster their training data into body poses with similar dynamics for more general motions (walking and running). Their work demonstrates that strong priors on human motion allows 2D pose estimation for the motion that is moving fast.

There are some research that have investigated the use of motion priors for 3D motion reconstruction. In [46], Howe et al. use snippets of motion from a database to infer 3D pose from tracked image features of simple movements. From a sequence of 2D poses, the 3D motion is reconstructed by finding the MAP estimate of the short motion sequences. Sigal et al. [47] adopt limb and head detectors that is incorporate into the learned motion model to infer the monocular human pose of walking with automatic initialization. Homan pose and motion estimation is solved with non-parametric belief propagation via stochastic sampling over a loopy graph. The work by Urtasun and Fua [48] use temporal motion models from sequences of motion capture data. The 3D human motion is reconstructed using a deterministic optimization scheme at a much reduced computational cost. PCA is applied to provide a low-dimensional parametrization. They fit full-body models to stereo data for walking and running. Furthermore, by using a multi-activity database, the parametric motion model is then used to constrain the movements with variable speed from stereo. In [49], Urtasun et al. use a Gaussian Process Latent Variable Models (GPLVM) to learn prior models specific movements such as golf-swings or walking from the monocular image sequences. GPLVM generate smooth mappings between pose space and latent space, which is useful for the use of gradient descent to improve the human pose estimation. In later work [50, 51], a Gaussian Process Dynamical Model (GPDM) is learned a dynamical motion model in the latent space from training data. The work by Moon and Pavlović [52] has investigated the effect of specific dynamics in the embedding on human motion estimation of 3D articulated objects in monocular image sequences.

In summary, the introduction of a specific human body motion has achieved the full-body pose estimation of complex movements from multiple views or single view. Nevertheless, there are two main drawbacks of top-down pose estimation. First is the fact that the manual initialization in the first frame of a video sequence is needed since the initial estimation is often obtained from the previous frame.

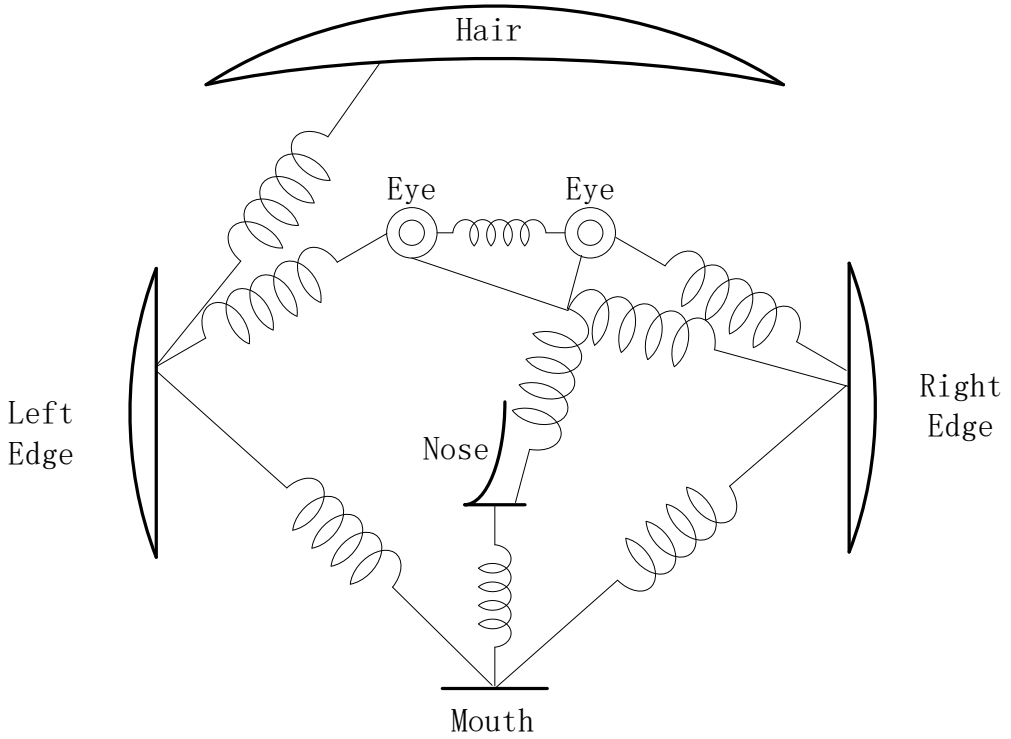


Figure 2.1: Schematic representation of face model, indicating components and their linkages.

Another drawback is the high computational cost of forward rendering the human body model (3D or 2D) and calculating the similarity between the human model projection and the observed images.

2.3 Bottom-up pose estimation

Bottom-up pose estimation approaches are characterized by detecting human body parts and then assembling these into a human body structure. The body parts are usually described by 2D templates and located by part detectors. Bottom-up approaches have the advantage that no manual initialization is needed and no specific model prior is required. Thus, the bottom-up pose estimation methods have less limit in its application and more robust to rapid movements. In the past decades, there are more and more researchers focus on these bottom-up approaches [53–58]. Among these methods, pictorial structure based methods are the most successful techniques for bottom-up pose estimation.

2.3.1 Pictorial structure

A pictorial structure is method to model an observed object by a collection of parts arranged in a deformable configuration. The appearance models are used to model

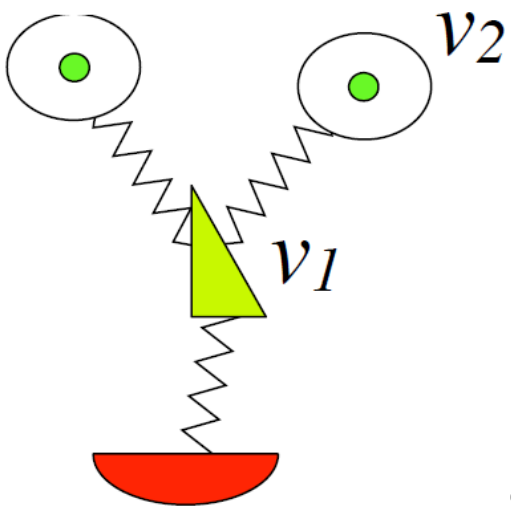


Figure 2.2: Simplify the original schematic representation to a tree structure.

each part separately, and the deformable configuration is represented by spring-like connections between the parts. The problem of matching a pictorial structure to an image is that of finding the best placement of the parts in an observed image, where the quality of a placement depends both on how well each part matches the image and on how well the placements agree with the deformable configuration.

Dates back to 1973, Pictorial Structures (PS) is introduced by the work of Fischler and Elschlager [57]. They propose the schematic representation of face model in the pictorial structures framework. This representation that simplifies the translation problem is the fact that the components (picture pieces, local evaluation arrays, etc.) and the relational forms (springs) are two-dimensional rather than one dimensional entities. The body parts of a human is modeled as a conditional random field (CRF). As shown in the Fig. 2.1, the components of a face is described in this schematic representation including (hair, right edge, left edge, nose, mouth and two eyes). The components of the face are linked by "springs." This "springs" joining the rigid parts of the face serve both to constrain relative movement and to measure the "cost" of the movement by how much they are "stretched." This spring-like connections between pairs of parts represent the deformable configuration of the face. The generic appearance models are used for the components of the face. Furthermore, a dynamic programming approach is developed which takes advantage of the decomposition to reduce drastically the computational requirements.

A natural way to express such a PS model is in terms of an undirected graph $G = (V, E)$, where $V = v_1, \cdots, v_n$ is a set nodes of the n parts, and there is an edge $(v_i, v_j) \in E$ for each pair-connected parts v_i and v_j. An instance of the observed subject is given by a configuration $L = (l_1, , \cdots, l_n)$. Each l_i denotes the position (location) of part v_i in the observed image. The location of each part is able to specify its position or more complex parameterizations. In [57], the problem of matching a PS model to an observed image is defined as the minimization of an

energy(cost) function. For each part v_i, a match cost function $m_i(I, l_i)$ measures the degree of mismatch when this part is placed at location l_i in the image I. For each pair of connected parts (v_i, v_j), there is a cost function $d_{ij}(l_i, l_j)$ measuring the degree of deformation of the model when part v_i is located at l_i and part v_j is located at l_j in the image. The goal is to find the best matched configuration, as measured by the match cost function $m_i(I, l_i)$ and the deformation cost function $d_{ij}(l_i, l_j)$. This best match can be expressed as:

$$L^* = \arg\min_L \Big(\sum_{(v_i,v_j)\in E} d_{ij}(l_i, l_j) + \sum_{v_i \in V} m_i(I, l_i) \Big) \tag{2.1}$$

This is a minimization problem that is quite general and appears in many fields of the computer vision. Generally the deformation costs are only a function of the relative position between two connected parts.

In the work of Felzenszwalb and Huttenlocher [55, 58], an efficient algorithm is proposed for the energy minimization problem in a pictorial structure. They also extend this part-based model to many objects, including faces, people and animals. The connections between parts are usually assumed to form a tree structure that allows efficient inference at test time. As shown in Fig. 2.2, a tree structure is used to represent the human object. The problem is to find best location l_2 for each v_2 corresponding to v_1. This can be solved by removing v_2, and repeating with smaller tree, until only a single part. Furthermore, they demonstrate that the restriction to a tree structure allow to use standard dynamic programming techniques, and the restriction in the form of each pair connected parts allow to use the distance transforms.

2.3.2 Pictorial structure based methods for pose estimation in still images

The pictorial structure model for an observed object is given by a collection of parts with connections between certain pairs of parts. More specifically, for articulated human objects, the parts can be divided into the torso, arms, head and legs of the human. In PS model, the required number of human body parts depends on the application and the required accuracy. For example, an articulated human body model with 14 parts is able to provide more accurate results as against a model with 6 body parts. The pictorial structure based human pose model is illustrated in Fig.2.3.

Following the work in [55, 57], many researchers focus on pictorial structure based models for pose estimation in still images. In [59], the pictorial structure are extended with correlations between human body parts in an image. For the walking objects, correlations between upper arm and leg are used for the robust estimation of human poses. Ronfard et al. [60] use the pictorial structures concept but replace simple part detectors by dedicated detectors that learn appearance model for each part using Support Vector Machines (SVM). The Dalal-Triggs detector [61] use a single filter on histogram of oriented gradients (HOG) features to represent and

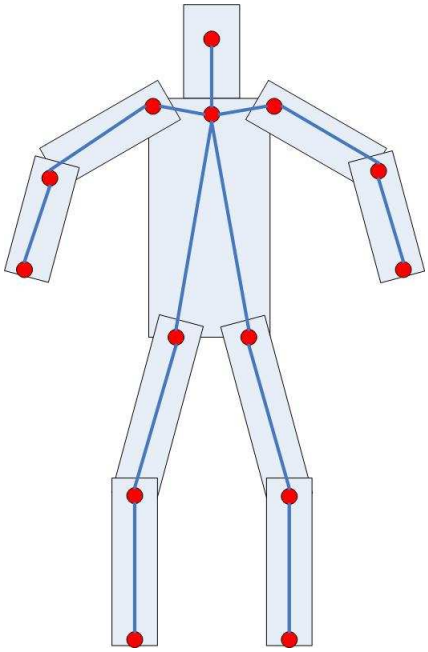


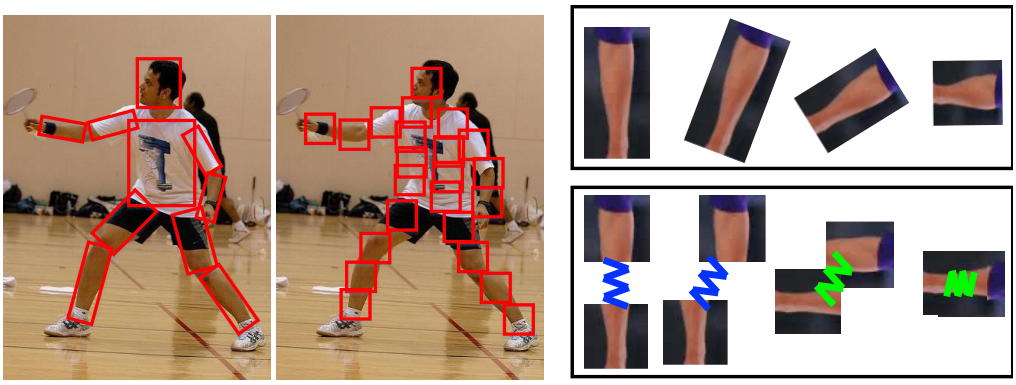
Figure 2.3: Human body model based on pictorial structure: each node and link corresponds to a part and a physical connection between parts.

detect human. This detector uses a sliding window approach, where a filter is applied at all positions and scales in an observed image. After the introduction of HOG descriptor, many researchers use HOG to build their appearance models for the human detection and pose estimation.

2.3.2.1 Deformable part model

Deformable part model [62, 63] is a method based on the pictorial structure framework. At first, it is proposed for the object detection. Then a great deal of work extend deformable part model for human pose estimation [1, 4, 64–68]. As described in [62, 63], a deformable part model is defined by a "root" filter plus a set of parts filters and associated deformation models. The score of this deformable part model at a particular location and scale within an image is equal to the score of the root filter at the given location plus the sum over the maximum scores of the part filters on its location, and minus a deformation cost between each pair of parts. The root scores and part filter scores are defined by the dot product between a filter and a subwindow of a HOG feature pyramid computed from an input image. The feature pyramid use higher resolution features for obtaining high recognition performance. Following Eq. 2.1, the deformable part model can be described as:

$$score(l_0, \cdots, l_n) = \sum_{i=0}^{n} \alpha_i \cdot \phi(I, l_i) - \sum_{ij \in E} \gamma_{ij} \cdot \psi(l_i, l_j) + b \tag{2.2}$$



(a) (b)

Figure 2.4: (a) The left is the pose detection with classic pictorial structure model, and the right is the estimation with flexible mixture of parts model. (b)The top is a single part that have different orientation and scale in classic model. The bottom is the small part by translating large parts connected with a spring.

where $\phi(I, l_i)$ is a feature vector extracted from the location l_i for part i in image$I$. $\psi(l_i, l_j) = [x_i - x_j, (x_i - x_j)^2, y_i - y_j, (y_i - y_j)^2]^T$ is the relative positions between part i and part j. E is a set of links between two different parts. α_i and γ_{ij} are vector of model parameters. The score can be expressed in terms o f a dot product, $\beta \cdot \Phi(I, L)$, between a model parameters β and a feature vector,

$$\beta = (\alpha_0, \cdots, \alpha_n, \cdots, \gamma_{ij}, \cdots, b). \tag{2.3}$$

$$\Phi(I, L) = (\phi(I, l_0), \cdots, \phi(I, l_n), \cdots, -\psi(l_i, l_j), \cdots, 1). \tag{2.4}$$

This illustrates a connection between deformable part model and linear classifiers. The model parameters are learned with the Support Vector Machines (SVM). The best possible locations are optimized by maximizing the score function $score(l_0, \cdots, l_n)$. Furthermore, a mixture model is defined to improve the performance. The mixture number $m_i \in \{1, \cdots, M\}$ is defined for part i. As in the case of a single model, the score function of a mixture model also can be expressed by a dot product between a vector of model parameters β and a vector $\Phi(I, L)$. While the object is defined with mixture components, the vector of the mixture model parameters β is the concatenation of the model parameter vectors for each component. The vector $\Phi(I, L)$ is sparse, with non-zero entries defined by $\Phi(I, L')$.

$$\beta = (\beta_1, \cdots, \beta_m) \tag{2.5}$$

$$\Phi(I, L) = (0, \cdots, 0, \Phi(I, L'), 0, \cdots, 0) \tag{2.6}$$

2.3.2.2 Flexible mixtures of parts model

In recent years, Flexible Mixtures of Parts (FMP) [1] model is one of the most successful methods for articulated human pose detection in static images. This model is based on the deformable part models. Unlike in traditional models, they use a mixture of pictorial structures with small, non-oriented parts. Their parts corresponded to the mid and end points of each limb. The parts were modeled as a mixture in order to capture the orientations of the limbs.As shown in figure 2.4, the smaller templates are used in FMP model that is more flexible to represent each body part. All body parts are related in a tree structure, and can be efficiently optimized with dynamic programming. The effective discriminative parts-based model [62] is used to learn all parameters, including local appearance, co-occurrence relations and spatial relations (based on structured SVM).

In [69], Tian et al. extend FMP to the spatial hierarchy of mixture model for human pose estimation. This model uses an exponential number of poses with a compact mixture representation on each part. They sample the pose type from the learned model employe latent tree model as the root nodes to handle geometric deformation. The work by Park and Ramanan [70], a N-best algorithms is proposed to generate a set of N high-scoring candidates. The the single-best pose is computed from N-best candidates by dynamic programming. They demonstrate that the locally ambiguous can be refined by their proposed approach. Wang and Li [66] use mid-level body parts in their latent tree model and propose an algorithm for automatically learning the tree to connect all the parts. Their model contains 14 single parts and 10 combined parts. The combined model is used to have more effective appearance model. They demonstrate that their model performs well in human pose estimation. Contour-based features have been proposed for articulated pose estimation [71], in an attempt to solve some of the background confusion situations.

2.3.2.3 Poselets and hierarchical models

The limitation of the hierarchical poselet models is building detectors of non-rigid limbs inaccurate as they are variable in appearance.

One of the motivations for using a small rigid part-based model is that it allows normalisation over configuration for each part. Due to lighting changes, clothing and body shape, an appearance model need capture variation in the space of possible appearance. This approach however leads to appearance models which roughly represent parallel edges or tapered cylinders leading to false positive detections. If enough image data were available, one could hope to build appearance models for pose estimation with at least two connected parts. This connected part-based appearance is far more salient than the appearance of a single part. Thus, the appearance model have more context. This is the motivation of the Poselet approach in the work by Bourdev and Malik [72].

A Poselet is a detector trained for a particular configuration and appearance of rigid parts or large portions of human bodies (e.g. torso + left arm). Wang et al. [64] propose hierarchical poselet model for pose estimation by loopy belief propagation

algorithm. They bridge the gap between part-based methods and exemplar-based methods. Exemplar-based methods search for images with similar whole body configurations. The limitation of exemplar-based approaches is that they cannot handle a test image of which the legs in are similar to a training image and arms are similar to another training image. The approach can be used for 3D human pose estimation in single images [72], human detection [73] and attribute classification [74]. Pishchulin et al. in [75] defined a tree model in which the unary and pairwise terms are conditioned on poselets evidence. This conditional model is defined by all body parts that are connected a-priori, but which becomes a tractable PS model. The poselets serve as a mid-level representation that jointly encodes articulation of several human body parts in an observed image. Similarly, the methods in [66] use mid-level body parts in their latent tree model and propose an algorithm for automatically learning the tree to connect all the parts. They combine poselets and small body parts together in the tree structure model. Here poselets are used to handle large variance in appearance. They demonstrate their proposed model perform well in several datasets. A very recent work [76] improves the model with more levels of parts and achieves a good performance. Instead of performing inference on a learned graphical model, they build a hierarchical inference machine for articulated human pose estimation. This method is a sequential prediction algorithm that emulates the mechanics of message passing to predict a confidence for each variable. However it should be noted that the poselet-based methods have the limitation of the enough number of training data. Thus, it is unlikely that the Poselet approach will be effective without a significant number of training data.

Many recent works also introduced higher-level parts in hierarchical models for pose estimation. The motivation of this approach is to combine the benefits of both part-based approaches and the multiple part Poselet approach. Most hierarchical methods include a whole person detector at their root and individual parts at the leaves. Early work on hierarchical models for 2-D human parsing is the AND/OR graph in [77]. They define the appearance models on sub-parts of body segments and put all small pieces together in the hierarchical model. Wang et al. [64] propose an approach of hierarchical Poselets based on the pictorial structure model. Such approaches lead to increased performance at the estimation of lower limbs but fail to deal with the poses which are included in training data. [65] use large-scale parts which can be integrated into a hierarchical, coarse-to-fine representation. Their model strikes a balance between model complexity and model richness by sharing appearance models of part types and by decomposing complex poses into pairwise relationships. Duan et al. [78] proposed hierarchical composite model via an optimization procedure for joint learning. [69] employed latent tree model as the root nodes to handle geometric deformation. They propose a hierarchical spatial model that can capture an exponential number of poses by sampling the pose mixture from the learned model.

2.3.3 Pictorial structure based methods for pose estimation in video

Pictorial structure model is widely used to estimate the human pose in video. Compared to the pose estimation in still images, the temporal component of videos provides an additional cue for estimation, as strong dependencies of human body part positions exist between temporally close video frames.

The strike-a-pose work [79] searches at least one frame in the video sequence for a predefined characteristic pose, easier to detect than a general pose. Based on this idea, they build a person-specific appearance model for human pose estimation. Eichner and Ferrari [56] present better appearance models for the pictorial structure. They show that some parts have rather stable location in the reference frames and the appearance models of different parts are statistically related. For example, the lower arms of a person are colored either like the torso or like the face. Only rarely they have an entirely different color. Thus, the appearance of some parts can be predicted from the appearance of other parts. They learn a location prior of parts with regard to the reference frame and an appearance transfer mechanism of different parts from training data. These cues are exploited to generate appearance models for body parts. In [80], an upper body pose estimation method is proposed. This approach is used to estimate upper body pose in uncontrolled images, without prior knowledge of background, clothing, or the position and scale of the human body parts in an image or video. A generic upper body detector is used to restrict the position and appearance of the human body parts in an image. This upper body detector is trained by using a sliding window mechanism followed by non-maximum suppression. Sapp and Taskar [67] proposed a multimodal at the global level, they use 32 pose modes to model a side-body. This model is trained with both large-scope and local part-level cues. They employ a structured cascade model selection step which controls the trade-off between speed and accuracy.

The optical flow is another method that is used as a cue either for body part detection or for pose propagation from frame-to-frame. Sapp et al. [81] introduce optical flow as feature to locate foreground contours. Each submodel in the their defined tree structure tracks a single joint through time. The foreground contours integrate well with their pose estimation method. In [82], Fragkiadaki et al. exploit optical flow for the pose segmentation. They combine coarse piece-wise affine with reliable pixel correspondences from optical flow. A fine-grain optical flow is used to track elbows and move limbs of the articulation chain. This "articulated" flow can accurately follow the articulated objects (human body) with large rotations or mixed displacements of rigid parts. By the work of Cherian et al. [83], a method for estimating articulated human pose in videos is presented, which is also based on the optical flow. The optical flow is used to extend flexible mixture-of-parts model [1] in a single image. First a set of pose candidates is generated in each frame with an optical flow based method for human pose estimation. Then they compute the K best poses [70] in each frame to obtain a diverse set of candidate poses. Furthermore, they decompose the K best candidate human pose into limbs

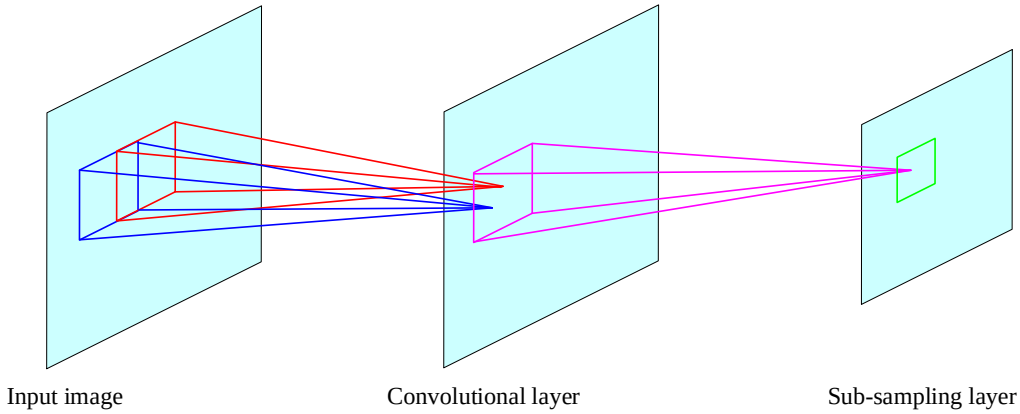


Figure 2.5: Illustration of a convolutional neural network.

and track them to generate body-part sequences. Finally, the complete pose is recomposed by mixing these part sequences.

2.4 Deep convolutional neural network for pose estimation

Over the last couple of years, deep learning techniques have made tremendous progress in computer vision [84]. Deep Convolutional Neural Networks (DCNNs) are a type of these techniques in deep learning framework and has become the method of choice in many fields of computer vision. DCNNs have shown outstanding performance on visual classification tasks and more recently on object detection.

2.4.1 Deep convolutional neural network

Robustness to particular transformations is a desired property in many computer vision tasks. Typical variances of images and videos include translation, rotation and scaling [85]. Tangent propagation [86] is one method in neural networks to handle transformations. Convolutional Neural Networks (CNNs) are a different approach to implementing transformation invariance in neural networks, which are inspired from biological processes. The concept of a convolutional neural network is illustrated in Fig. 2.5. It contains a single layer of convolutional units, followed by a sub-sampling (or pooling) layer , as described in [86]. Many recent works have demonstrated the power of DCNNs in many computer vision tasks, such as text recognition [87–89], face recognition [90], visual classification [91, 92], object detection [93–95], action recognition [96, 97]. These networks comprise several layers of non-linear feature extractors and are therefore said to be "Deep" [98].

2.4.2 Deep convolutional neural network in computer vision

In recent computer vision researches, CNN is a popular deep learning approach. In [99], multiple levels of representation are learned to model complex non-linear relations. DCNN has demonstrated outstanding performance for image classification tasks [100–102]. More recently, CNN architectures have been successfully applied to object localization and detection [103, 104]. Long et al. [105] presented fully convolutional networks that allow for per-pixel predictions like semantic segmentation. In [103], DetectorNet addresses the problem of object detection and proposes a multi-scale inference procedure to produce high-resolution object detections. OverFeat [104] generates dense, multi-scale CNN features for object classification, localization and detection from an image by examining every sliding window.

Girshick et al. [106] propose the R-CNN method by applying high-capacity convolutional neural networks to bottom-up region proposals [107] for localizing and segmenting objects. It outperforms OverFeat and improves the performance by more than 30% relative to the state of the art on PASCAL VOC 2012. [108] adopt the R-CNN [106] to localize part and verify that the use of region proposals can help localize smaller parts. Based on this, R-CNN is shown to be effective for fine-grained recognition. In [109], He et al. proposed a spatial pyramid pooling in DCNN for visual recognition. This network structure can generate a fixed-length representation regardless of the scale of input images. The experiment results demonstrate that their proposed method is very effective in classification and detection tasks. However, these method does not consider the complex relations between different parts and is not applicable to human pose estimation.

2.4.3 Deep convolutional neural network for pose estimation

For pose estimation, the best performing algorithms today [110–112] are based on deep convolutional networks. There were early examples of using convolution neural networks for pose comparisons [113]. More recently, Toshev et al. [111] develope DeepPose which is a cascade of CNN-based joint regressors are applied to capture context and reason about human pose in a holistic fashion. The DeepPose networks use a full image as the input and formulate the methods without any explicit feature representations or part detectors. In [114], Jain et al. introduce a multi-layer CNN architecture and combine low-level features with a higher-level weak spatial model to improve the performance. Following [114], Tompson et al. [115] attempt to combine a CNN Part-Detector with a part-based Spatial-Model into a unified learning framework. This method can significantly increase the pose estimation performance. In [110], Chen and Alan specify a graphical model with image dependent pairwise relations for human pose estimation. In this model, CNN is used to learn conditional probabilities for the presence of parts and the spatial relationships between parts. Recently, Fan et al. [116] propose dual-source deep convolutional neural networks to join the body part appearance and the holistic view of each body part for more accurate human pose estimation.

Temporal information in videos was initially used with DCNNs for action recognition [97], where a two-stream DCNN architecture incorporates spatial and temporal networks. Here temporal information is optical flow that is used as an input feature in this networks. Following this work, [117, 118] investigate the use of temporal information to estimate the upper-body or full-body poses in videos. The optical flow or RGB features are computed from nearby frames into the network, and joint positions are localized in the current frame. More recently, Pfister et al. [119] propose a method for pose estimation in videos that is able to utilizes apparance across multiple frames. The fully convolutional spatial network predicts a confidence heatmap for each body joint in these frames. They demonstrate that the heatmaps of positions from neighbouring frames can be warped and aligned using optical flow from the current frame.

Chapter 3

Pose estimation with annealed particle filter

Contents

3.1 Introduction

Top-down approaches use a direct model to match the observed image. Particle filtering [31] is one of the common top-down approaches for human pose estimation, which used the pose in the current frame and a dynamic model to predict the next pose. Particle Filter (PF) uses multiple predictions, obtained by drawing samples of pose and location prior, and then propagating them using the dynamic model by comparing them with the local image data and calculating the likelihood. The prior is typically quite diffused (because motion can be fast) but the likelihood function may be very peaky, containing multiple local maxima which are hard to account for in detail. Annealed Particle Filter (APF) [36] or local searches are the ways to tackle this problem. APF has been widely used for articulated human pose estimation due to its ability to precisely estimate the statistics of multi-modal and non-Gaussian processes. However, the performance of annealed particle filter drops when the frame rate is lower or the motion is moving fast.

This chapter presents a top-down approach to estimate articulated human pose based on foreground learning by Flexible Mixtures of Parts (FMP) model [6], which has shown strong ability to detect and estimate human motion from still images. In our work, we make use of the sequence to learn and subtract the background, and then jointly track and detect body parts in multiple views. Part models are trained based on PARSE dataset [7]. We evaluate our approach on the HumanEva-II dataset, which is a standard benchmark for 3D pose estimation. Finally, we empirically show the robustness of our approach under challenging conditions for human motion capture such as fast moving and self occlusion.

The rest of this chapter is organized as follows: Section 3.2 describes particle filter for human pose estimation. Section 3.3 introduces foreground modeling by FMP model. Section 3.4 presents the proposed foreground learning based method for pose estimation. Implementation details are presented in Section 3.5. Finally, Section 3.6 draws the conclusion of this chapter.

3.2 Filtering

3.2.1 Particle filter

Particle filter algorithm was developed for tracking objects, using recursive Bayesian estimators derived from Monte Carlo sampling techniques which can handle non-Gaussian processes and multi-modal. In order to make an estimation of the tracked object parameter this algorithm suggests using the importance sampling. Importance sampling is a general technique for estimating the statistics of a random variable. The estimation is based on samples of this random variable generated from other distribution, called proposal distribution, which is easy to sample from [120].

Commonly used in tracking problems, it aims at estimating the posterior density $f(x_t|y_{1:t})$, where $y_{1:t}$ notates the history of observation (x_t is a hidden state vector and y_t is a measurement at time t). The observation process is $f(y_t|x_t)$. The posterior density is represented by a set of weighted particles $\{(x_t^{(0)}, \pi_t^{(0)}) \cdots (x_t^{(N)}, \pi_t^{(N)})\}$, where $\pi_t^{(i)} \propto f(y_t|x_t^{(i)})$. The filtering distribution can be calculated using two steps.

Prediction step:

$$f(x_t|y_{1:t-1}) = \int f(x_t|x_{t-1}) f(x_{t-1}|y_{1:t-1}) dx_{t-1}. \tag{3.1}$$

Filtering step:

$$f(x_t|y_{1:t}) \propto f(y_t|x_t) f(x_t|y_{1:t-1}), \tag{3.2}$$

where $f(y_t|x_t)$ is the likelihood, and $f(x_t|y_{1:t-1})$ predicts the state at time t. Variations of PF: Sequential Importance Sampling (SIS) draws particles from a proposal distribution and then for each particle a proper weight is assigned as follows:

$$\pi_t^{(i)} \propto f(y_t|x_t^{(i)}) f(x_t^{(i)}|x_{t-1}^{(i)}) / q(x_t^{(i)}|x_{t-1}^{(i)}, y_t). \tag{3.3}$$

Consequently, a basic problem is that the distribution $f(y_t|x_t)$ may be very peaky, because $f(y_t|x_t)$ usually detects several local maxima instead of choosing the

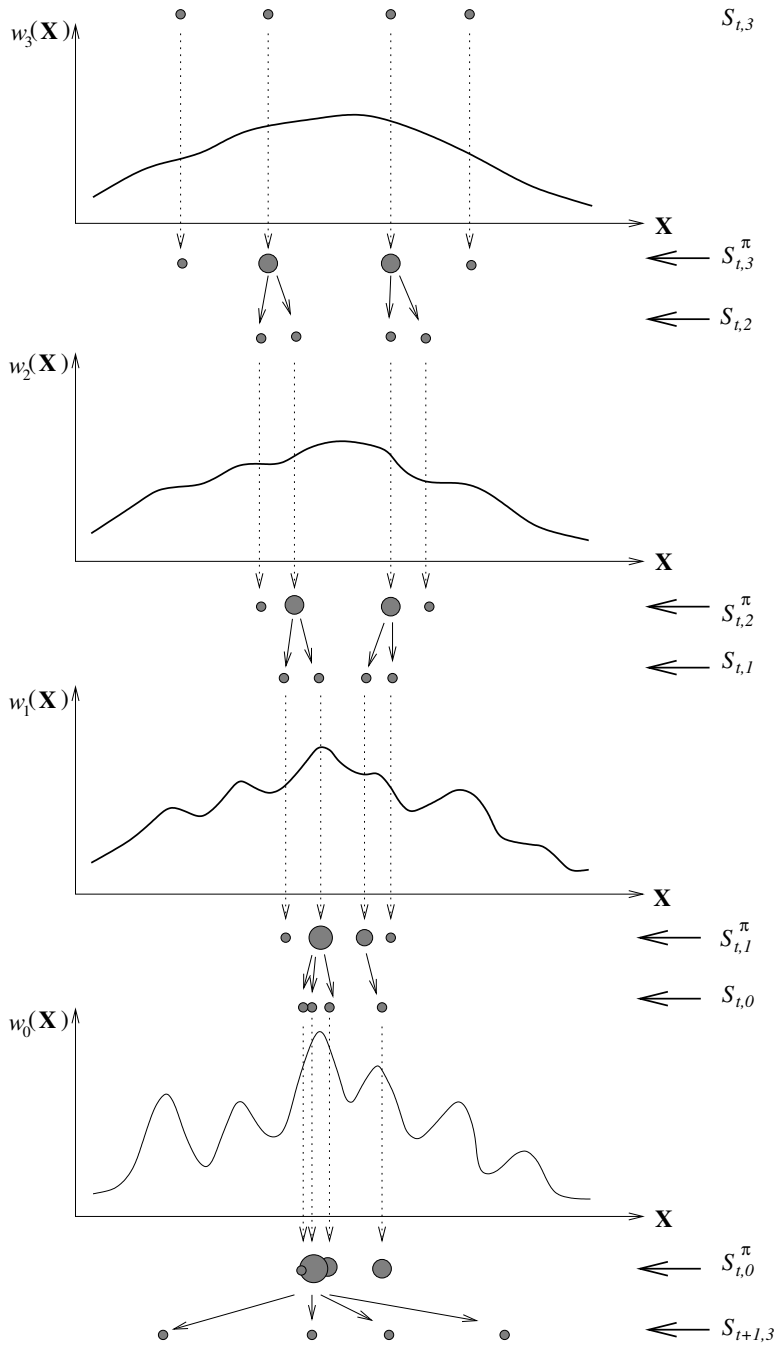


Figure 3.1: Illustration of the annealed particle filter with $M = 3$. A set of sparse particles is started at layer M and gradually towards the global maximum.

global one. This usually happens for the high dimensional problems, like body part tracking. Another factor is the computational cost of calculating $f(y_t|x_t^{(i)})$. Often an intuitive weighting function $w_t^i(y_t, x)$ can be constructed that approximates the probabilistic likelihood, which requires much less computational effort to evaluate [34]. Therefore, the problem becomes to find configuration x_k that maximizes the weighting function $w_t^i(y_t, x)$, move to towards the global maximum of the weighting function.

3.2.2 Annealed particle filter

It has been shown in several works that Sampling Importance Resampling (SIR) Particle Filters [121] are a good approach for tracking in low dimensional spaces, but they become inefficient in high-dimensional problems. Deutscher [34] proposed a variation of the SIR framework by introducing the concept of Annealing particle filter. In body pose tracking problems, the likelihood approximation is often a function with several peaked local maxima [122]. The main idea of APF is to utilize a series of weighting functions($w_0(y_t, x)$ to $w_M(y_t, x)$), where each $w_m(y_t, x)$ differs only slightly from $w_{m-1}(y_t, x)$. The weighting function $w_M(y_t, x)$) is designed to be very smoothed, representing the overall trend of the search space while $w_0(y_t, x)$ might be peaky. This is achieved by using

$$w_m(y_t, x) = (w_0(y_t, x))^{\beta_m}, \tag{3.4}$$

where $1 = \beta_0 > \cdots > \beta_M$ and $w_0(y_t, x)$ is equal to the original weighting function. Therefore, each annealing run includes M layers, and is started at layer M. As illustration in Fig. 3.1, one annealing run is performed at time t. $\mathcal{S}_{t,m}^{\pi}$ denotes a set of weighted particles, while $\mathcal{S}_{t,m}$ is a set of unweighted particles.

3.3 Foreground modeling

3.3.1 Basic pictorial structure model

Pictorial structure [55] model for an object is given by a collection of parts with connections between certain pairs of parts. More specifically, for human body model, the parts can correspond to the head, torso, arms and legs of the human, as shown in Fig. 3.2. Pose parameters are optimized by maximizing the score function which is defined as follows,

$$S(I, L) = \Sigma_{i \in V} \alpha_i \cdot \phi(I, p_i) + \Sigma_{ij \in E} \beta_{ij} \cdot \psi(p_i, p_j), \tag{3.5}$$

where I denote the image, V is a set of nodes and p_i, p_j are locations of part i and j. α_i is unary template for part i, and $\phi(I, p_i)$ is local image features at location p_i in image I; β_{ij} is pairwise springs between part i and part j, and $\psi(p_i, p_j) = [x_i - x_j, (x_i - x_j)^2, y_i - y_j, (y_i - y_j)^2]^T$ is the relative location between part i and part j.

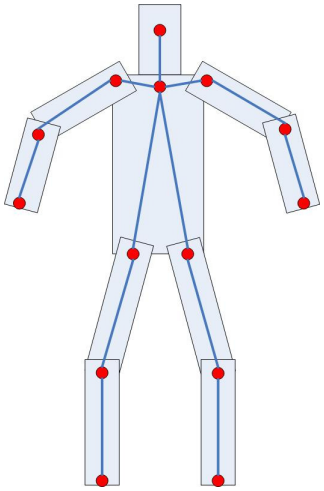


Figure 3.2: Human body model based on pictorial structure: each node and link corresponds to a part and a physical connection between parts.

3.3.2 Flexible mixtures of parts model

Flexible mixtures of parts model is also based on PS framework. As shown in the first row of Fig. 3.3, this model uses smaller body parts rather than the larger one, which is significantly faster than the original model. This section describes FMP model. Taking mixture of parts into account, the new score function can be defined as:

$$\begin{aligned} S(I,L,M) =& \Sigma_{i\in V}\alpha_i^{m_i}\cdot\phi(I,p_i) \\ &+ \Sigma_{ij\in E}\beta_{ij}^{m_i m_j}\cdot\psi(p_i,p_j) + S(M), \end{aligned} \tag{3.6}$$

where m_i is the mixture of part i, $\alpha_i^{m_i}$ is unary template for part i with mixture m_i, and $\beta_{ij}^{m_i m_j}$ is pairwise springs between part i with mixture m_i and part j with mixture m_j. $S(M) = \Sigma_{ij\in E} b_{ij}^{m_i,m_j}$ is a sum of pairwise scores, and the pairwise parameter $b_{ij}^{m_i,m_j}$ favors particular co-occurences between part i with mixture m_i and part j with mixture m_j. E is a set of links each of which connect two parts. As shown in the second row of Fig. 3.3, we can note that this method can be confused by background. Contour-based features have been proposed for articulated pose estimation [71], in an attempt to solve some of the background confusion situations. In our work, we make use of the sequence to learn and subtract the background, and then jointly track and detect body parts in multiple views.

3.4 Tracking with FMP-APF

Based on only the annealing particle filter, one cannot efficiently track fast apparent motions due to low frame rates. On the other hand, FMP model cannot find some

Figure 3.3: Human body parts detection by flexible mixtures of parts model. First row shows the detection results are correct, while the second row shows FMP fails to detect body parts.

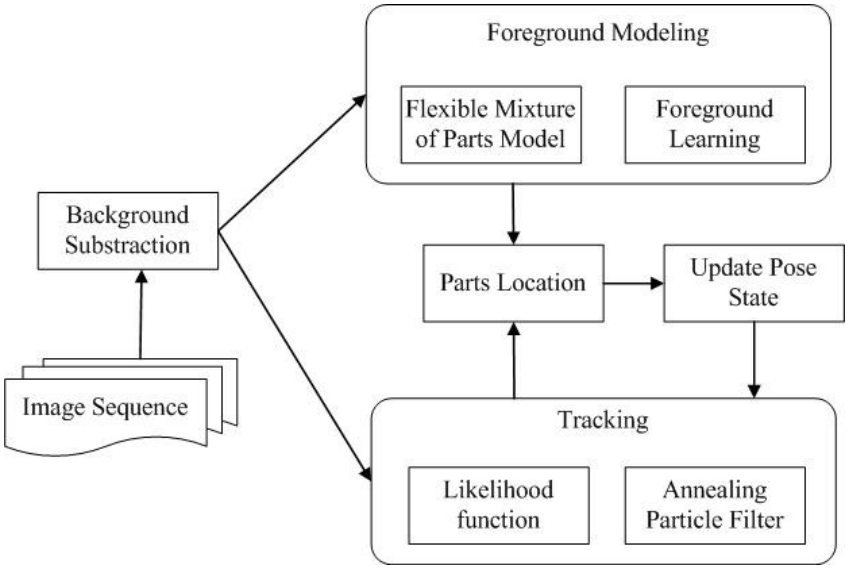


Figure 3.4: Illustration of the proposed method.

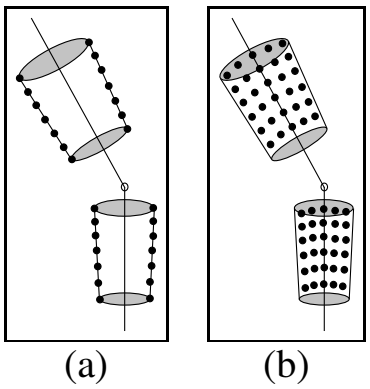


Figure 3.5: Configurations of the pixel map sampling points for the edge-based measurements (a) and the silhouette-based measurements (b).

body parts due to the overlapping and occlusion. For these reasons, we combine these two methods together, and propose a foreground learning-based approach. Fig. 3.4 is the illustration of this proposed FMP-APF scheme.

3.4.1 Modeling the body

As is common in the literature, we build the body model as a 3D kinematic chain with limbs, which consists of 15 segments: pelvis area, torso, head, upper and lower arms and legs, hands and feet. Our objective is to find the pose of the body over time, which is parametrized by a reduced set of 34 parameters comprising the global position and orientation of the pelvis and the relative joint angles between neighboring limbs. The shoulders, hips and thorax are modeled as ball and socket joints with 3 degrees of freedom, the clavicles are allowed 2 degrees of freedom, while the knees, ankles, elbows, wrists and head are assumed to be hinge joints requiring only one degree of freedom [123].

3.4.2 Likelihoods

For each particle in the posterior representation, the likelihood represents how well the projection of a given body pose state fits the observed images. Many image features could be used, including optical flow, color and adaptive appearance regions, however, the most common approaches are based on silhouette and edge information.

Edge-based log-likelihood function is estimated by projecting the pose into the edge map sparse points:

$$-\log f^{e}(y_t|x_t) \propto \frac{1}{k}\sum_{i=1}^{k}(1 - M_i^e(x_t, Y))^2, \tag{3.7}$$

where Y is the image from which the pixel map is derived, and $M_i^e(x_t, Y)$ are the values of the edge pixel map at the K sampling points taken along the model's

silhouette (See Fig. 3.5 (a)).

Silhouette-based log-likelihood function is estimated by projecting the pose into the foreground silhouette map sparse points:

$$-\log f^r(y_t|x_t) \propto \frac{1}{k}\sum_{i=1}^{k}(1 - M_i^r(x_t, Y))^2, \tag{3.8}$$

where $M_i^r(x_t, Y)$ are the values of the foreground silhouette pixel map at the K sampling points taken from the interior of the model (See Fig. 3.5 (b)).

3.4.3 Detection by FMP in multi-view scene

As discussed in Section 3, FMP fails to detect body parts, because of overlapping and occlusion. Multiple views have a powerful ability to solve these problems by combining the detection in each view. So, this thesis extends FMP to the multi-view case:

$$\begin{aligned} S(I, P, M, K) = {} & \Sigma_{i\in V}\alpha_i^{m_i} \cdot \phi(I_k, p_{i,k}) \\ & + \Sigma_{ij\in E}\beta_{ij}^{m_i m_j} \cdot \psi(p_{i,k}, p_{j,k}) + S(M), \end{aligned} \tag{3.9}$$

where I_k denotes the image I in view k, $p_{i,k}$ is the location of part i in view k, and $S(M)$ is a sum of pairwise scores. Let (n, m) denotes a pair of different views from K views. Thus, $p_{i,n}$ and $p_{i,m}$ are the locations of part i in views n, m, which are calculated by Eq.3.9. Nevertheless, sometimes the position p_i is not enough accurate. Epipolar constraint is used between two views to remove false measurements and to achieve more accurate localization. The fundamental matrix F is the representation of epipolar geometry and the epipolar constraint is represented by $p_{i,n}^T F p_{i,m} = 0$. If points $p_{i,n}$ and $p_{i,m}$ are coherent, the $p_{i,n}$ lies on the epipolar line $l = F p_{i,m}$. In this case, the 3D position q_i of part i can be computed by the back-projection of $p_{i,n}$ and $p_{i,m}$ as follows,

$$\begin{cases} L_{i,n}(\lambda) = P^+ p_{i,n} + \lambda C_n, \\ L_{i,m}(\lambda) = P^+ p_{i,m} + \lambda C_m, \end{cases} \tag{3.10}$$

where $L_{i,n}, L_{i,m}$ are two rays, P^+ is the pseudo-inverse of camera matrix P, and C_n, C_m are the camera centers of view m, n. The intersection of the two rays $L_{i,n}, L_{i,m}$ is the 3D position q_i. From all possible (n, m) of the K views, it at least one pair is coherent, then the 3D position is retained and we consider the next body part. Otherwise, an update of the previous of 3D position is performed by APF as detailed in next subsection.

3.4.4 Update the state with APF

As discussed above, some body parts don't have any multi-view correspondence by FMP. To solve this, we introduce APF in FMP framework to realize robust tracking

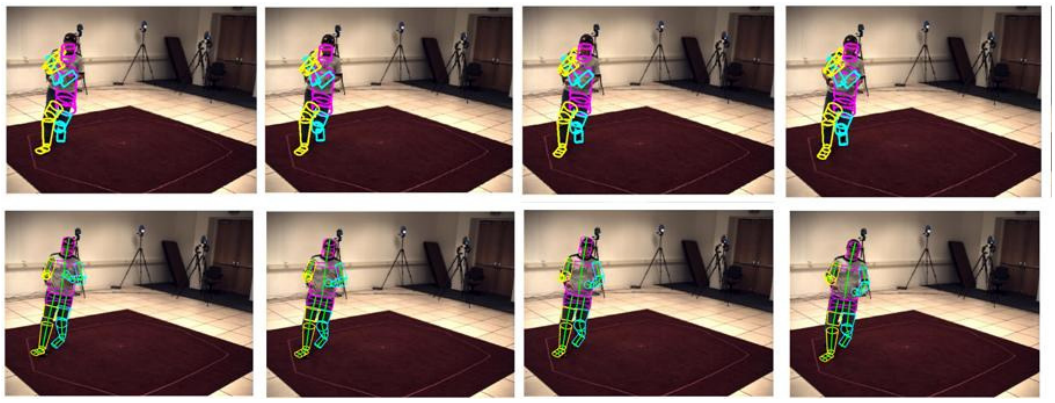


Figure 3.6: Comparison of motion detection. First row shows motion detection by baseline algorithm. Second row shows the detection by FMP-APF.

for all body parts. From APF, the optimal configuration have been computed from the particle set at the bottom layer using:

$$x_{t-1} = \Sigma_{j=1}^{N_p} \pi_{t-1,0}^{(j)} x_{t-1,0}^{(j)}, \tag{3.11}$$

where N_p is the number of particles. Let $x_{t-1} = (X_{t-1,1}, X_{t-1,2} \cdots X_{t-1,S})$, $X_{t-1,i}$ is the parameter vector for part i at time $t-1$, and S is the number of body parts. As discussed in Section 2, after the sample is drawn, the state estimation for each particle becomes:

$$f(x_t|x_{t-1}, y_t) \propto f(y_t|x_t) f(x_t|x_{t-1}). \tag{3.12}$$

APF is not appropriate for estimating high dimensional state parameters, especially for the state parameters of fast move body parts (arms and legs). The main idea of this thesis is to use the detection of body parts to infer a subset of the state parameters. Suppose that the state vector x_t can be decomposed into $(x_{t,1}, x_{t,2})$, where $x_{t,1}$ is to be computed by APF, while the state parameters $x_{t,2}$ have already been computed by multi-view FMP. Therefore, the state estimation for each particle can be rewritten as:

$$\begin{aligned} f(x_{t,1}|x_{t-1,1}, & x_{t,2}, y_t) \propto \\ & f(y_t|x_{t,1}, x_{t,2}) f(x_{t,1}|x_{t-1,1}, x_{t,2}), \end{aligned} \tag{3.13}$$

the above expression combines tracking and detection to perform automatic recovering from body-parts tracking failures. As represented by the term $f(x_{t,1}|x_{t-1,1}, x_{t,2})$, which is used to estimate the state $x_{t,1}$ based on the parameter $x_{t-1,1}$ and $x_{t,2}$. After all particles are computed, the optimal configuration have been computed at the bottom layer as follows:

$$x_{t,1} = \Sigma_{j=1}^{N_p} \pi_{t,1,0}^{(j)} x_{t,1,0}^{(j)}, \tag{3.14}$$

so the new state vector x_t is also computed (see Algorithm 1).

Algorithm 1 APF-FMP.

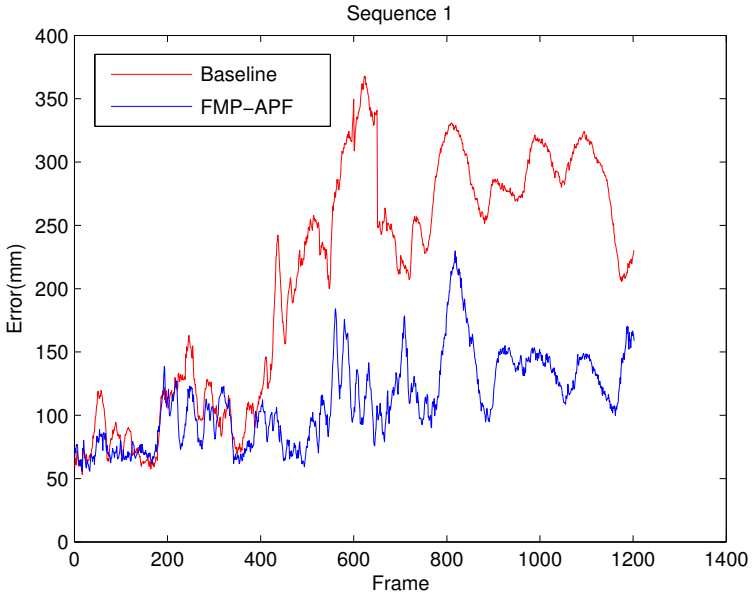
Input: Images $I_{t,k}$ from views k at time t ($k = 1 \cdots K$),
state vector x_{t-1} at time $t-1$.
for $n = 1 \cdots K-1$
  for $m = n+1 \cdots K$
    Compute $F_{n,m}$ between views n,m
  end
end
% Applying multi-view FMP
for $i = 1 \cdots S$ % body part i
  for $n = 1 \cdots K-1$
    for $m = n+1 \cdots K$
      if $p_{i,n}^T F_{n,m} p_{i,m} == 0$
        Compute rays $L_{i,n}(\lambda) = P^+ p_{i,n} + \lambda C$
        $L_{i,m}(\lambda) = P^+ p_{i,m} + \lambda C$
        Compute q_i by the intersection of $L_{i,n}, L_{i,m}$
        Update the parameter vector $X_{t,i}$ with q_i
        $x_{t,2}(i) = X_{t,i}$
      end
    end
  end
end
Set the state vector $x_t = (x_{t,1}, x_{t,2})$
% $x_{t,1}$: non-matching with FMP
% $x_{t,2}$: matching with FMP
Compute $p(x_{t,1}|x_{t-1,1}, x_{t,2}, y_t)$ for each particle
Compute $x_{t,1} = \Sigma_{j=1}^{N_p} \pi_{t,1,0}^{(j)} x_{t,1,0}^{(j)}$
RETURN: x_t

3.5 Implementation details

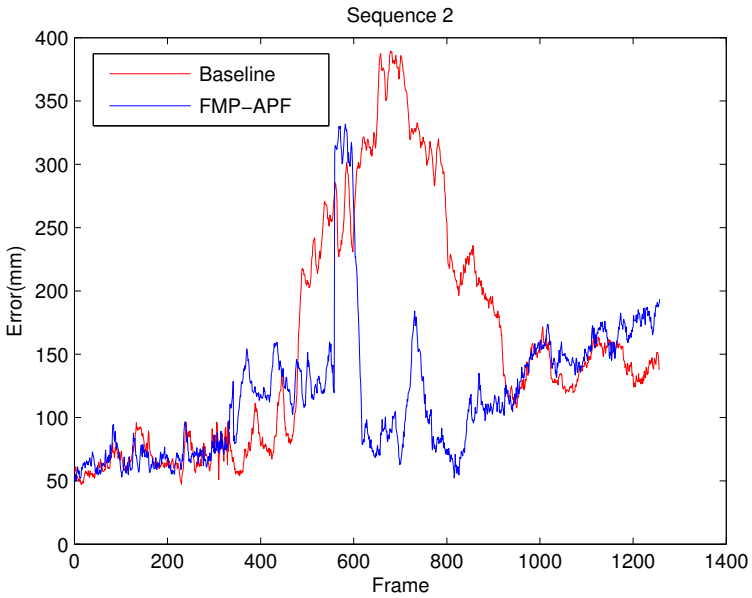
We conducted a series of experiments to measure the effectiveness of our proposed models in real multi-view 3D settings on a variety of sequences from the HumanEva-II dataset.

Datasets. HumanEva [36] is a standard benchmark for 3D human pose estimation in the laboratory setting, which allow quantitative evaluation of performance. The dataset consists of HumanEva-I and HumanEva-II by a set of multi-view sequences. HumanEva-II was captured using a more sophistical hardware system that allowed better quality motion capture data. So we utilize sequences of walking, jogging and balancing from HumanEva-II for our experiments.

Evaluation of our approach. We evaluate the performance of our approach on HumanEva-II by the measure proposed in [36], which computes the 3D errors in millimeters of the locations of the joints and end points of the limbs between 15



(a)



(b)

Figure 3.7: Comparison of errors. The first 400 frames are for walking, and 401-700 frames are for jogging, and the rest for balancing. (a): 3D errors for the first subject by baseline algorithm and FMP-APF. (b): 3D errors for the second subject.

virtual markers on the body and detection results. Then we compare performance against the baseline algorithm based on the methods of Deutscher and Reid [35], which have the same likelihoods and the same number of samples. Balan et al. [124] report APF with edge-based and silhouette-based likelihood function with 5 layers (200 particles per layer). The errors of their work reach 263±60mm for tracking the first 150 frames of the sequence. We applied standard particle filtering with foreground learning and compared our proposed method with baseline in Fig. 3.6 by computing the 3D errors in millimeters of HumanEva II. The performance is clearly improved by our method, especially for jogging, as shown in Fig. 3.7.

3.6 Conclusion

In this thesis we proposed a new framework for human body parts tracking, which is based on flexible mixture of parts model and annealing particle filter. FMP model is used for foreground learning in multiple views, and APF is used for tracking body parts. And then jointly track and detect body parts by estimating and updating the pose state. Experimental results have shown that the proposed method can efficiently track fast apparent motions.

Chapter 4

Pose estimation with multiple mixture parts model based on upper body categories

Contents

4.1 Introduction

In this chapter, we address the problem of building a better model in a pictorial structure framework. It shares some similarities with the Flexible Mixture of Parts model [1] (FMP), but it does not estimate the motions directly. We propose an upper body based Multiple Mixture Parts model (MMP), which is divided into two stages. The first stage includes three steps: upper body detection, model category estimation and model selection. Different categories are proposed for defining upper body models, and each upper body model corresponds to one mixture model. Each mixture model is trained on the dataset in which all the images share the same

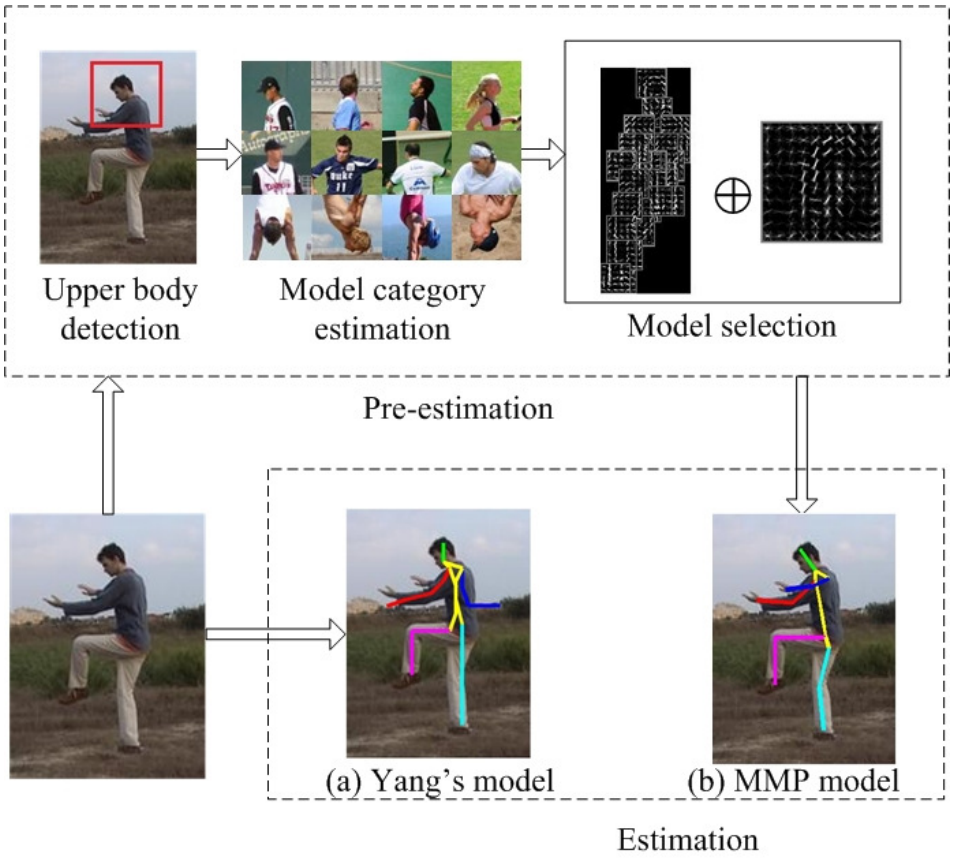


Figure 4.1: The motivation of this thesis in using upper body categories based model: we use the upper body model for pre-estimation, while a multiple mixture part model combined with middle limb models are used to realize more effective human part detection.(a) is the estimation by Yang's model [1], while (b) is based on MMP model.

upper body category. As for model selection, there are two-level models. One is the local part model and the other is the combined model. The combined models are defined between pairs of joints, while local part models are built between each joint and middle point. Two-level models are used to achieve a more accuracy estimation. The first stage only categorizes poses, so it is referred to as the pre-estimation stage, while the other is the estimation stage. In the second stage, the MMP model is proposed to join different mixture models. Each mixture model in MMP corresponds to one upper body category. Different mixture models not only have different kinematic constraints, but also have different numbers of part models. As illustrated in Fig. 1, the MMP model is compared with the method in Ref. 1. Figure 4.1 (a) is the estimation by Yang's model while figure 4.1 (b) is based on the proposed model. This framework has shown that the performance of human pose estimation can be improved in each stage. In the pre-estimation stage, the main task is to find a more effective and discriminative model to categorize poses. In the estimation stage, the main task is to detect each body part using different part based models. Moreover, this framework is an effective solution for more complex poses and can be easily extended to more MMP categories or any other pose category estimation methods. The upper body is chosen to distinguish different models, since the appearance models of the upper body are discriminative in different categories.

Concerning experiments, first, the proposed upper body model is tested on the Buffy stickmen dataset [125] which is used for the state-of-the-art comparison. Then, the MMP model is trained with different numbers of samples on the LSP and LSPET datasets (Leeds Sports Pose Extended Training Dataset) [126]. In addition, the results are compared using different annotations on Leeds Sport Dataset (LSP) [127], and the performance of the proposed model on the LSP and UIUC people datasets [128] is evaluated.

The rest of the chapter is organized as follows. Section 4.2 presents related work and background on human pose estimation. Section 4.3 describes the proposed MMP model for pose estimation. The experiments and the results are provided in Section 4.4. Finally, Section 4.5 presents the conclusion of this chapter.

4.2 Background

This subsection explains why the upper body is chosen to categorize poses. Then we introduce the flexible mixtures-of-parts model which is the original model.

4.2.1 Support vector machine

4.2.1.1 Support vector machines for binary classification

The original Support Vector Machines (SVM) was invented by Vapnik and Lerner [129]. SVM provides a powerful tool for learning models that generalize well even in sparse, high dimension settings [130]. SVM minimizes the structural risk, the probability of misclassifying patterns for a fixed but unknown probability distribution of

the data [131]. In addition to performing linear classification, SVM can efficiently handle nonlinear classification problems using kernel trick [132–134]. Considering the problem of separating the set of training data $\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_n, y_n)\}$, a feature vector $\boldsymbol{x_i} \in \mathbb{R}^d$ belong to two separate classes $y_i = \pm 1$. In linear classification, the data are separated by a maximum-margin hyperplane,

$$\boldsymbol{w}^\top \boldsymbol{x}_i + \rho = 0, \tag{4.1}$$

where $\boldsymbol{w}$ denotes a vector, ρ is a constant. The decision function for each datum $\boldsymbol{x}$ can be defined as:

$$\varphi(\boldsymbol{x}) = \mathbf{sgn}(\boldsymbol{w}^\top \boldsymbol{x}_i + \rho). \tag{4.2}$$

Assuming the minimization distance of the data to the final separating plane is 1, one has:

$$\begin{cases} \boldsymbol{w}^\top \boldsymbol{x}_i + \rho \geq +1, y_i = +1, \\ \boldsymbol{w}^\top \boldsymbol{x}_i + \rho \leq -1, y_i = -1. \end{cases} \tag{4.3}$$

The above two equations can be rewritten as:

$$y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + \rho) \geq 1. \tag{4.4}$$

The distance of each vector $\boldsymbol{x_i}$ to the decision plane can be defined as:

$$d(\boldsymbol{x}) = \frac{y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + \rho)}{\|\boldsymbol{w}\|} \geq \frac{1}{\|\boldsymbol{w}\|}, \tag{4.5}$$

Maximizing the margin becomes minimizing $\|\boldsymbol{w}\|$ under constraints. The constrained optimization problem is solved by introducing Lagrange multipliers α_i. Thus, the corresponding Lagrangian is,

$$L(\boldsymbol{w}, \rho, \boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{n} \alpha_i (y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + \rho) - 1). \tag{4.6}$$

where the vector $\boldsymbol{\alpha}$ is composed by α_i. Taking the derivatives of function (4.6) with respect to $\boldsymbol{w}$ and ρ, we have:

$$\frac{\partial L}{\partial \boldsymbol{w}} = 0 \qquad \Rightarrow \boldsymbol{w}^\top = \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i^\top, \tag{4.7}$$

$$\frac{\partial L}{\rho} = 0 \qquad \Rightarrow \sum_{i=1}^{n} y_i \alpha_i = 0. \tag{4.8}$$

Replace (4.7) (4.8) into (4.6), the optimization problem can be obtained as follows:

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j, \tag{4.9}$$

$$\text{subject to: } \sum_{i=1}^{n} \alpha_i y_i = 0, \alpha_i \geq 0. \tag{4.10}$$

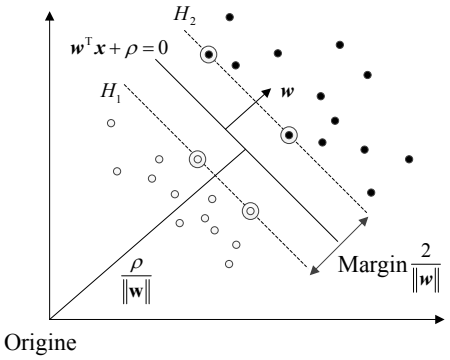


Figure 4.2: Principle of support vector machines for two classes classification. The support vectors are labeled by circle.

This problem can be addressed by standard quadratical program method. Only few a small proportion of the Lagrange multipliers α_i are not 0, these corresponding training samples are called support vector (SV). Once the $\boldsymbol{\alpha}$ are measured, the optimal hyperplane can be defined as:

$$\boldsymbol{w} = \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i, \tag{4.11}$$

$$\rho = -\frac{1}{2}\boldsymbol{w}^\top (\boldsymbol{x}_r + \boldsymbol{x}_s), \tag{4.12}$$

where $\boldsymbol{x}_r$ and $\boldsymbol{x}_s$ are any support vectors from each class. These vectors satisfy the following the equation:

$$\alpha_r, \alpha_s > 0, y_r = -1, y_s = +1. \tag{4.13}$$

As presented in Fig.4.2, the circle samples in supplementary hyperplane H_1 and H_2 are support vectors. The hard margin classifier is defined as,

$$\varphi(\boldsymbol{x}) = \mathbf{sgn}\ (\boldsymbol{w}^\top \boldsymbol{x} + \rho) = \mathbf{sgn}\ (\sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i^\top \boldsymbol{x} + \rho). \tag{4.14}$$

Usually, the data is not linearly separable. To deal with this case, soft margin SVM is used to tolerate mislabeled data points. The degree of misclassification of sample $\boldsymbol{x}_i$ is quantified by slack variable ξ_i, $\xi_i \geq 0$. The optimization problem becomes:

$$\min\ \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{n} \xi_i, \tag{4.15}$$

$$\text{subject to: } y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + \rho) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, \ldots, n. \tag{4.16}$$

As the case in hard margin classifier, we have:

$$\max \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j, \tag{4.17}$$

$$\text{subject to: } \sum_{i=1}^{n} \alpha_i y_i = 0, \quad 0 < \alpha_i \leq C, \quad i = 1, 2, \ldots, n. \tag{4.18}$$

The standard quadratical program is used to address this soft margin problem. If the training examples are nonlinearly separable, linear SVMs need to extended to nonlinear SVMs with a kernel function. If an kernel κ is given, the decision function becomes:

$$\varphi(\boldsymbol{x}) = \mathbf{sgn}(\sum_{i=1}^{n} \alpha_i y_i \kappa(\boldsymbol{x}_i, \boldsymbol{x}) + \rho). \tag{4.19}$$

4.2.1.2 Structured support vector machines

The structured support vector machine [135] is a discriminative method for structured learning that generalizes the Support Vector Machine (SVM) classifier. The traditional SVM supports binary and multiclass classification, while the structured SVM allows the training samples with structured labels. Moreover, structured SVM can be used to predict complex objects like trees or sequences, and it provides state-of-art prediction accuracies in many area of computer vision.

Inference: Given a sample $\boldsymbol{x} \in \mathcal{X}$, a learning function $f : \mathcal{X} \to \mathcal{Y}$ maps this given sample to a predicted label from the label space $\mathcal{Y}$. A linearly-parametrized structural predictor produces a label of the form

$$f(\boldsymbol{x}) = \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \quad \boldsymbol{w} \cdot \Psi(\boldsymbol{x}, y) \tag{4.20}$$

where $\boldsymbol{w}$ denotes a parameter vector obtained from training. Ψ is a feature function extracting the combined feature representation from a given sample and label.

Learning: Given a set of n training instances $(\boldsymbol{x}_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \ldots, n$ from a sample space $\mathcal{X}$ and label space $\mathcal{Y}$, the standard structured SVM is given as follows. for each sample, each representing the value of the maximum. The standard structured SVM primal formulation is given as follows.

$$\begin{aligned} &\min_{\boldsymbol{w}, \boldsymbol{\xi}} \|\boldsymbol{w}\|^2 + C \sum_{i=1}^{n} \xi_i \\ &\text{subject to:} \boldsymbol{w} \cdot \Psi(\boldsymbol{x}_i, y_i) - \boldsymbol{w} \cdot \Psi(\boldsymbol{x}_i, y) \geq \Delta(y_i, y) - \xi_i, \\ &\quad i = 1, \ldots, n, \forall y \in \mathcal{Y} \backslash y_i \end{aligned} \tag{4.21}$$

where $\Delta(y_i, y)$ is an arbitrary loss function measuring a distance in label space. The high loss $\Delta(y_i, y)$ increases the required margin. ξ_i is the slack variable.

4.2.2 Upper body based pose category

The upper body based pose category is a fundamental idea in this thesis and is the main task in the first stage of the proposed framework. Here, we present it in detail. Different poses can share similar upper body appearances. Nevertheless, there are too many pose categories, which makes it difficult to categorize poses directly. In this thesis, it is proposed to use the upper body to categorize pose. This means that human poses are categorized based on the viewpoints of upper body. Different poses can share the same view-based upper body model, and each upper body model corresponds to different poses that have similar upper bodies. Moreover, there are several reasons why the upper body is chosen to categorize poses. First, the detection of the upper body is more accurate than any other body part. Many previous studies [1, 64, 66, 75, 80] have shown that the PCP (Percentage of Correct Parts) results of the torso and head are higher than those of legs and arms. The second reason is that the upper body has different shapes in different viewpoints, which makes it possible to distinguish different upper body categories. Third, an upper body category based pose detector has the advantage that their kinematic prior is specific to each category and the part models are tuned to each view.

4.2.3 Flexible mixtures of parts model.

Here, it is proposed to introduce the Flexible Mixtures of Parts (FMP) model that is also based on a pictorial structure framework [64]. In this thesis, FMP is the basic model that we build on and serves as a baseline for comparison. This model shows excellent results in human body detection and pose estimation. Taking the mixture of parts into account, the score function of FMP can be defined as:

$$S(I,P,M) = \sum_{i\in V} \omega_i^{m_i} \cdot \phi(I,p_i) + \sum_{ij\in E} \omega_{ij}^{m_i m_j} \cdot \psi(p_i,p_j) + S(M), \tag{4.22}$$

where I denotes the image, V is a set of nodes and p_i, p_j are positions of part i and j. $\phi(I,p_i)$ is the local image feature at position p_i in image I, and $\psi(p_i,p_j) = [x_i - x_j, (x_i - x_j)^2, y_i - y_j, (y_i - y_j)^2]^T$ is the relative position between part i and part j. E is a set of links between two different parts. m_i is the mixture of part i, $\omega_i^{m_i}$ is a template for part i with mixture m_i, and $\omega_{ij}^{m_i m_j}$ are pairwise springs between part i and part j. $S(M)$ is a compatibility term. The parent j collects the messages from all its children and passes the messages to its parent recursively towards a root node. After all the messages have been passed to the root node, the root pose parameters are determined, maximizing the objective in Eq.(1).

4.3 Proposed MMP pose estimation method

The proposed method is composed of two stages: one is upper body detection and categorization, the other is human pose estimation. These two stages are described in the following subsection.

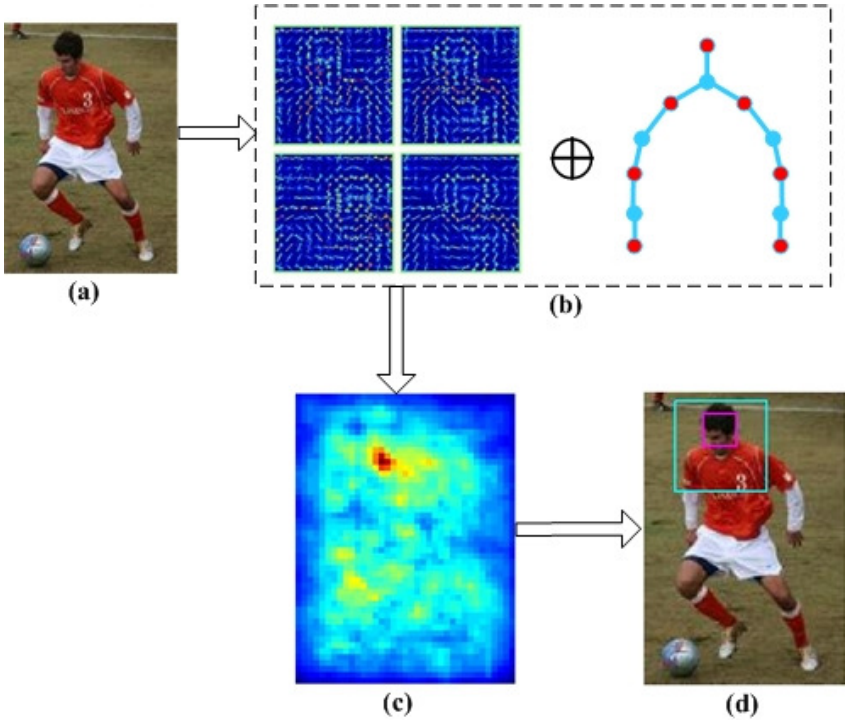


Figure 4.3: Upper body detector. (a) The input image. (b) The hierarchical upper body model: the first model is the global level upper body model, while the second one is the part based tree model (red nodes denote different joints, and head is the root node). (c) The combined score maps of upper body filters and part filters. (d) The result of upper body detection: the cyan box denotes the location of upper body.

4.3.1 Upper body detection and categorization (Pre-estimation stage)

This subsection gives a brief introduction to the upper body detection and presents the proposed hierarchical upper body model, and the proposed approach for estimating the categories of the upper body.

4.3.1.1 Hierarchical upper body model

Our upper body model is based on previous approaches for rigid object detection [61, 62, 80]. All these detectors use a sliding window mechanism followed by non-maximum suppression. An upper body detector was proposed in Ref. 80, which combined HOG templates with a face detector [136]. This model performs well on video frames from movies and TV shows. However, the results are poor when this method is used in some more challenging datasets (e.g. Leeds Sports Pose dataset (LSP) [127]). Thus, a hierarchical upper body model is proposed (Fig. 4.3), which

is also based on face and upper body detection. The main contribution of our work is to add the pairwise term between parts in the upper body instead of the face detector in Ref. 136. There are two-level part models (Fig. 4.3(b)): the global level upper body model and the local level part based model. For each level model, there are several mixture components similar to FMP. For the local level model, it is defined by 12 small parts including 4 parts for each arm, 1 for each shoulder and 2 for the head, which are used to compute the pairwise term. Then, the score function can be defined as:

$$\begin{aligned} S(I,P,M) =& \omega^m_{upper} \cdot \phi(I, p_{upper}) + \omega^m_{head} \cdot \phi(I, p_{head}) \\ &+ \sum_{i,j} \omega_{ij}^{m_i m_j} \cdot \psi(p_i, p_j) + S(M), \end{aligned} \tag{4.23}$$

which can be divided into three different terms: appearance, deformable and compatibility.

Appearance term: the first two terms in Eq.(2) are an appearance model that includes two-level local scores: upper body and head. ω^m_{upper} is the HOG template for the upper body with mixture type m, while ω^m_{head} is the template for the head.

Deformable term: $\Sigma_{i,j}\omega_{ij}^{m_i m_j} \cdot \psi(p_i, p_j)$ is the deformable term, where i, j denote different parts in upper body. It is also described as pairwise term which can be interpreted as attaching a spring between the two parts. This term can be computed by the distance transform from the leaf node to the root node.

Compatibility term: the last term $S(M)$ denotes whether two types are compatible in the training set. Together with the deformable term, it specifies an image-independent prior over part locations and types. Thus, let us study the optimal upper body location maximizing the following score function:

$$p^*_{upper} = \arg\max_p S(I, P, M), \tag{4.24}$$

where p^*_{upper} is the upper body location from the detection.

4.3.1.2 Estimation of upper body categories

After the detection of the upper body, it is proposed to estimate the categories of the upper body. The purpose of this step is to classify a variety of input images into different categories depending on the upper body. As illustrated in Fig. 1, three different upper-body categories are defined: left-right side view, near front-back view and handstand view. This step could be extended to more categories or other more discriminative body part features.

Let us investigate two main strategies for estimating upper body categories. In the first strategy, different sets of models are proposed to detect the upper body and estimate the upper body categories in one step. In the second strategy, we jointly detect and categorize poses for estimating upper body categories. The second strategy contains two separate steps. In the following, these two strategies are described specifically.

Strategy 1. In this case, only one step is used to estimate the upper body category using different sets of models. Let us rewrite the Eq. (2) associated with a configuration of upper-body categories:

$$\begin{aligned} S(I,P,M,C) =& \omega_{upper}^{m,c} \cdot \phi(I, p_{upper}) + \omega_{head}^{m,c} \cdot \phi(I, p_{head}) \\ &+ \sum_{i,j} \omega_{ij}^{m_i m_j, c} \cdot \psi(p_i, p_j) + S(M, C), \end{aligned} \tag{4.25}$$

where c denotes the category index of the upper body. By calculating the maximum scores in each upper body model, one can determine the category c, which is also used to select the model in MMP. Immediately, c is defined in the equation below:

$$c_1^* = \arg\max_c S(I, P, M, C), \tag{4.26}$$

where c_1^* is the selected category index.

Strategy 2. In this case, two steps are needed. The first is to detect the upper body using Eq.(2). The second is to estimate upper body categories based on the detection p_{upper}^*. Then, the score is defined as:

$$S(I, M, C) = \omega_{upper}^{m,c} \cdot \phi(I, p_{upper}^*), \tag{4.27}$$

where $\phi(I, p_{upper}^*)$ is the upper body feature with the location p_{upper}^*. The category index c_2^* is computed by:

$$c_2^* = \arg\max_c S(I, M, C). \tag{4.28}$$

Comparison of the two strategies. By analyzing these two strategies, we demonstrate which one works effectively. First, these two strategies are tested on the TUD Multiview Pedestrians dataset [137] with different pedestrians in the front-back view and right-left side view. As illustrated in Fig. 4.4, average scores are calculated with 95% confidence intervals based on the images in each view. Both strategies perform well on this dataset, which is due to the restricted poses in the TUD Multiview Pedestrians dataset. Moreover, let us evaluate the proposed method on the LSP dataset. Images are manually selected from the LSP dataset and divided into three subsets: side-view, front-view and handstand. Then the method is tested on these three subsets. As shown in Fig. 4.5, the average scores in strategy 1 are more distinguishable than those in strategy 2. Thus, strategy 1 performs better than strategy 2 on the LSP dataset.

4.3.2 Multiple mixture parts model (Estimation stage)

In this subsection, we firstly describe the general multiple mixture-part model composed of different separate mixture-part models (See Fig. 4.6). Each individual model corresponds to one category in the training data. The proposed two-stage MMP model has two advantages that the kinematic prior is specific to each category, and different models can be joined together to improve the performance. In this

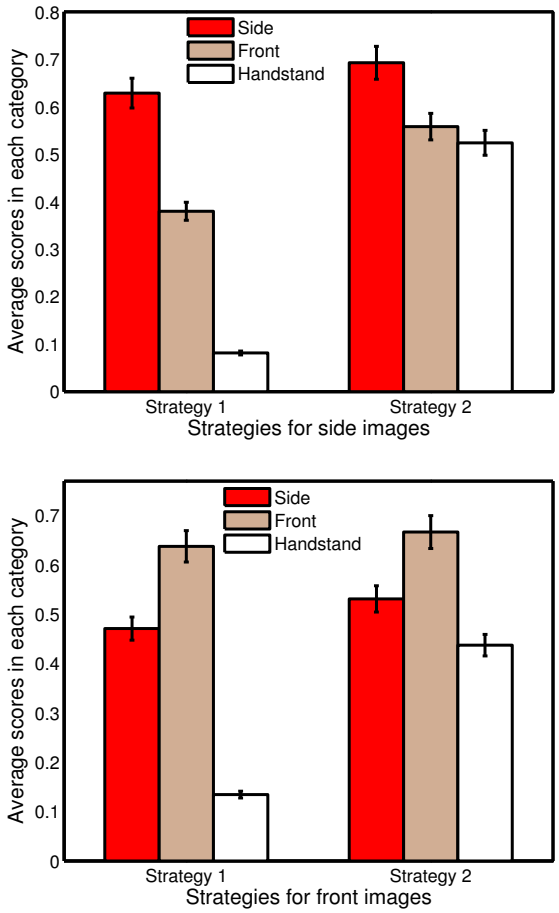


Figure 4.4: Comparison of two strategies on the TUD Multiview Pedestrians dataset: the bar charts are created with 95% confidence intervals. A successful strategy produces scores that significantly separates the three categories (Side, Front, Handstand).

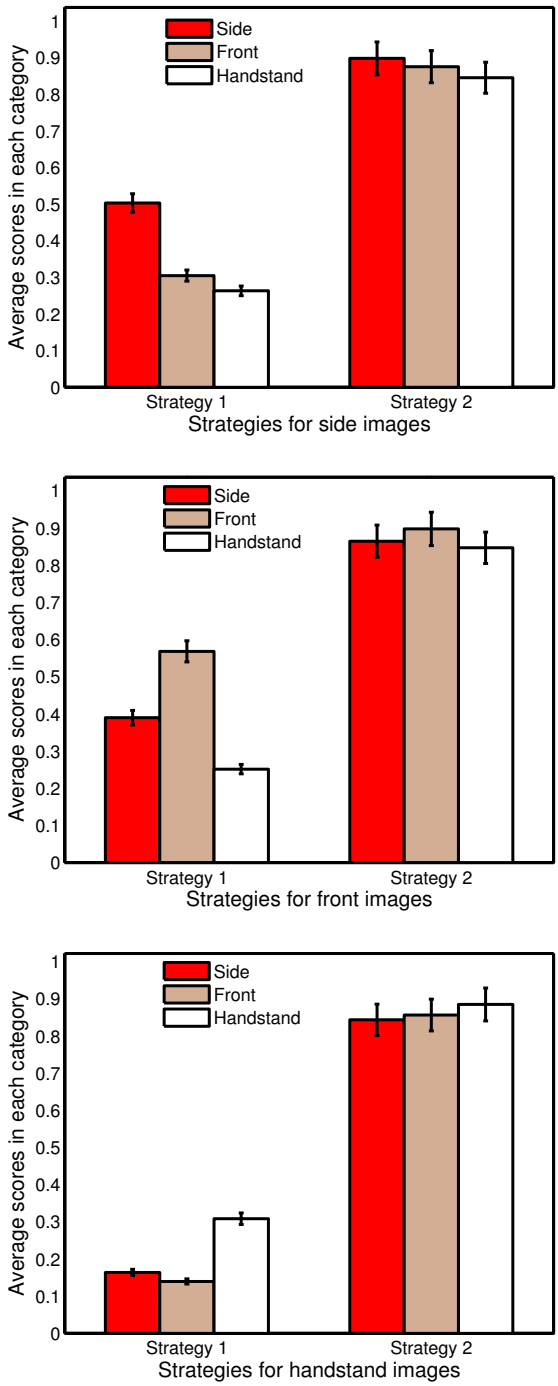


Figure 4.5: Comparison of two strategies on the LSP dataset. It includes three bar charts for each strategy, and each chart corresponds to one view. The data clearly shows that Strategy 1 better distinguishes the three categories (Side, Front, Handstand) as compared to Strategy 2.

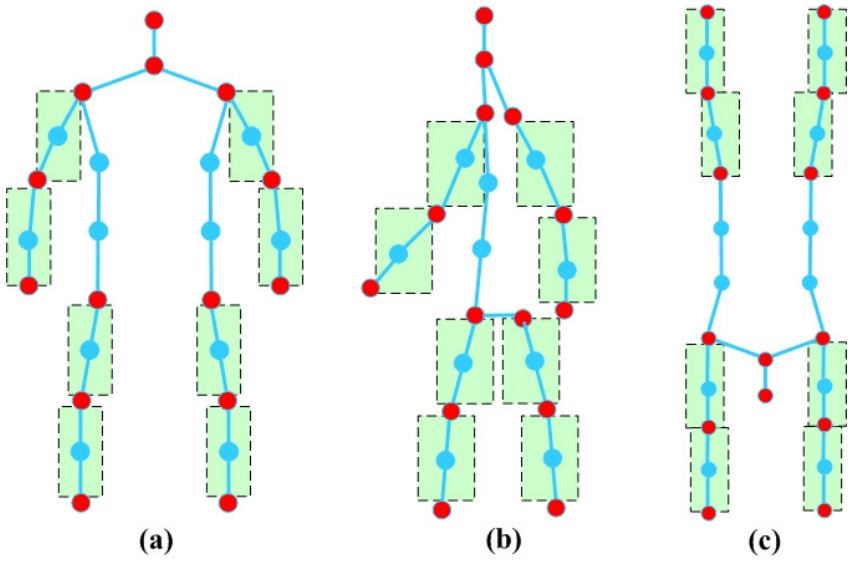


Figure 4.6: Multiple mixture parts model. Our MMP model is composed of a three category mixture parts model. The red nodes denote joints, and the cyan nodes denote middle points between two joints. The green boxes denote combined model in MMP. (a) The mixture parts model for near front-back view poses is composed of 26 parts. (b) The mixture parts model for right-left side view poses includes 24 parts. (c) The mixture parts model for handstand view poses has 26 parts.

thesis, three categories are proposed in the MMP model. The first category is for near front-back view poses (See Fig. 4.6(a)). This model has 14 joints and 26 parts. The second is for right-left side view poses (See Fig. 4.6(b)). It should be noted that the proposed model has 24 parts instead of 26 parts. In the 26-part model, there are 6 parts for the torso which are able to overlap in side view scenes. Nevertheless, the proposed 24-part model can not only match the human body kinematic constraints, but also reduce double-counting. The third category is for handstand-style poses (head down, feet up). It should be noted that the upper body in this category is more discriminative, and makes the pose categorization easier. For this model, we still adopt a 26-part model. In each category, there are 8 green boxes which denote 8 combined models. Since the limbs are more complex and the appearances change significantly, adding 8 combined models in MMP is proposed to give more context. The combined models share the joints with local part models. Therefore, the combined models are defined to connect two limb joints, and each limb joint has two-level models (a combined model and a local part model). The size of the combined model is larger than that of the local part model.

Similar to the upper body model, let us define the proposed MMP model as three terms: appearance, deformable and compatibility. It should be noted that the combined model is used in each term.

Appearance term: This term includes two levels of templates. One is for

combined models and the other is for local part models. The appearance score can be written as:

$$S_a(I,P,C) = \sum_{i\in V} \omega_i^{c,m_i} \cdot \phi(I,p_i) + \lambda \sum_{k\in V_l} \omega_k^{c,m_k} \cdot \phi(I,p_k), \tag{4.29}$$

where c denotes the category index of MMP, V is a set of local parts and V_l is a set of combined parts. p_i is the position of local part i while p_k is the position of combined part k. ω_k^{c,m_k} denotes a HOG template for combined part k with category index c and mixture type m_k, while ω_i^{c,m_i} is for local part level. These two-level appearance models are combined by the parameter λ, which controls the trade-off between two terms and is tuned manually.

Deformable term:

$$S_d(I,P,C) = \sum_{i,j\in E} \omega_{i,j}^{c,m_i m_j} \cdot \psi(p_i,p_j) + \lambda \sum_{f,g\in E_l} \omega_{f,g}^{c,m_f m_g} \cdot \psi(p_f,p_g), \tag{4.30}$$

where part i,j and part f,g are pairwise connected in E, E_l. E is a set of links between two parts in local part model and E_l is for the combined model.

Compatibility term:

$$\begin{aligned} S(M,C) =& \Sigma_{i,j\in E} b_{i,j}^{c,m_i,m_j} + \lambda \Sigma_{f,g\in E_l} b_{f,g}^{c,m_f,m_g} \\ &+ \Sigma_{i\in V} b_i^{c,m_i} + \lambda \Sigma_{k\in V_l} b_k^{c,m_k}, \end{aligned} \tag{4.31}$$

similar to the upper body model, $S(M,C)$ is defined as a co-occurences model. The parameter b_{ij}^{c,m_i,m_j} favors particular co-occurences between part i with mixture m_i and part j with mixture m_j in category c. For example, if part types correspond to orientations and part i and j are on the same rigid limb, b_{ij}^{c,m_i,m_j} would favor consistent orientation assignments. The parent j collects the messages from its all children and passes the messages to its parent recursively towards a root node.

Inference. Inference corresponds to maximizing the full score function:

$$p^* = \arg\max_p S_a(I,P,C) + S_d(I,P,C) + S(M,C), \tag{4.32}$$

where p^* denotes the locations of body parts. The root scores are used to generate multiple detection in image I by thresholding them and applying non-maximum suppression (NMS). Then, a backtracking is used to find the location and the type of each part in each maximal configuration.

Learning. During training, we have access to training images with ground truth joint/body locations and category numbers $\{I_t, P_t, M_t, C_t\}_{t=1}^T$. T is the number of training images. I_t denotes the image with the ground truth value P_t, the mixture index M_t and the manual category index C_t. It should be noted that P_t is not treated as hidden variable. To illustrate learning, let us write $Z_t = (P_t, M_t, C_t)$. The score function can be expressed in terms of a dot product, $S(I_t, Z_t) = \beta \cdot \Phi(I_t, Z_t)$, between a vector of model parameters β and a feature vector $\Phi(I_t, Z_t)$,

$$\begin{aligned} \beta =& (\omega_1^{c,m_1}, \cdots, \omega_N^{c,m_N}, \omega_{N+1}^{c,m_{N+1}}, \cdots, \omega_{N+K}^{c,m_{N+K}}, b_1^{c,m_1}, \cdots, b_N^{c,m_N}, \\ & b_{N+1}^{c,m_{N+1}}, \cdots, b_{N+K}^{c,m_{N+K}}, \cdots, \omega_{i,j}^{c,m_i,m_j}, \cdots, \cdots, \omega_{f,g}^{c,m_f,m_g}, \cdots, \\ & \cdots, b_{i,j}^{c,m_i,m_j}, \cdots, \cdots, b_{f,g}^{c,m_f,m_g}, \cdots), \quad (i,j \in E; f,g \in E_l) \end{aligned} \tag{4.33}$$

$$\Phi(I_t, Z_t) = (\phi(I, p_1), \cdots, \phi(I, p_N), \lambda\phi(I, p_{N+1}), \cdots, \lambda\phi(I, p_{N+K}), 1, \cdots, 1, \lambda, \cdots, \lambda, \tag{4.34}$$
$$\cdots, \psi(p_i, p_j), \cdots, \cdots, \lambda\psi(p_f, p_g), \cdots, \cdots, 1, \cdots, \cdots, \lambda, \cdots), \quad (i, j \in E; f, g \in E_l)$$

where N is the number of local part models and K is the number of combined models. Thus, let us define a large-margin learning objective function similar to the work of Ref. 63, 138:

$$\begin{aligned} &\arg \min_{\beta, \xi_t \geq 0} \frac{1}{2}\|\beta\|^2 + \mathcal{C}\sum_t \xi_t, \\ s.t. \forall t \in pos \quad &\langle \beta, \Phi(I_t, Z_t)\rangle \geq 1 - \xi_t, \\ \forall t \in neg \quad &\langle \beta, \Phi(I_t, Z)\rangle \leq -1 + \xi_t, \end{aligned} \tag{4.35}$$

where $\mathcal{C}$ controls the relative weight of the regularization term and ξ_t denotes the slack variables of the objective function. The constraint states that positive examples should score better than 1, while negative examples less than -1. It should be noted that the standard structured SVM do not require an explicit negative training set, and instead generate negative data from positive examples with mis-estimated labels Z. This leads the trained model to score a ground-truth pose highly and alternates pose poorly. The trained model with the above constraints is more robust and works well for human pose estimation.

4.4 Experiment results

The experiments are presented in this section. There are several state-of-the-art datasets in the human pose estimation, e.g. the Buffy dataset [125], the PASCAL Stickmen dataset [56]. Nevertheless, these datasets are only for upper-body pose estimation, which cannot meet the proposed full-body pose estimation. In this work, we evaluate the performance on the Leeds Sport Dataset (LSP) [127], the UIUC people dataset [128] and the Buffy stickmen dataset [125]. First, let us demonstrate that the proposed method performs well in upper body detection. In addition, an evaluation of the MMP model's performance is proposed.

4.4.1 Evaluation for upper-body detection

In the upper body detection experiments, there are two different level models: the global level upper body model and the local level part models. Each model is trained by using the first 1000 images in LSP dataset (1000 images for training and 1000 images for testing). This dataset has a large variety of pose changes. The training images are manually partitioned into three disjoint subsets: side view, near front-back view, handstand view. For the side view and near front-back training data, the images and labels are manually flipped to increase the size of our training set. The additional rotated images are used for handstand views, since the number of handstand view images is rather small in the training set. For each category, 10 mixtures are used in each level model.

Figure 4.7: Upper body estimation. We show the green bounding boxes for different types of upper body in the LSP dataset. The first row denotes the near front-back view, the second row is for the side view, and the third row shows the results of handstand view.

Dataset	Method	Detection rate
Buffy	Eichner et al. [80]	89.01
	Ferrari et al. [125]	88
	Niebles & Fei-Fei [139]	73
	Ours	**93.56**

Table 4.1: Comparison of upper body detectors on the Buffy dataset.

Figure 4.7 displays the results for upper body estimation including three different categories. The proposed model is tested on the 1000 testing images in LSP dataset. A detection is considered correct if 50% of the ground truth upper body is in the bounding boxes. The detection accuracies of the proposed upper body model achieve 92%. Compared with the upper body detector in Ref. 80, the proposed model is more robust. Testing their model on the Buffy data [125], the detection rate is 90% with 10% false positives. The results of applying Ref. 80 on LSP dataset are not satisfactory since LSP is more challenging and the face detector in Ref. 136 is not effective on this dataset.

To make a fair comparison, let us test the proposed upper body model on the Buffy dataset [125] by comparing it with the model in Ref. 80, 125, 139. For each method, the global upper body model is trained on the Buffy episodes 3 and 4, while Buffy episodes 2, 5 and 6 are used for testing (276 images in total). As illustrated

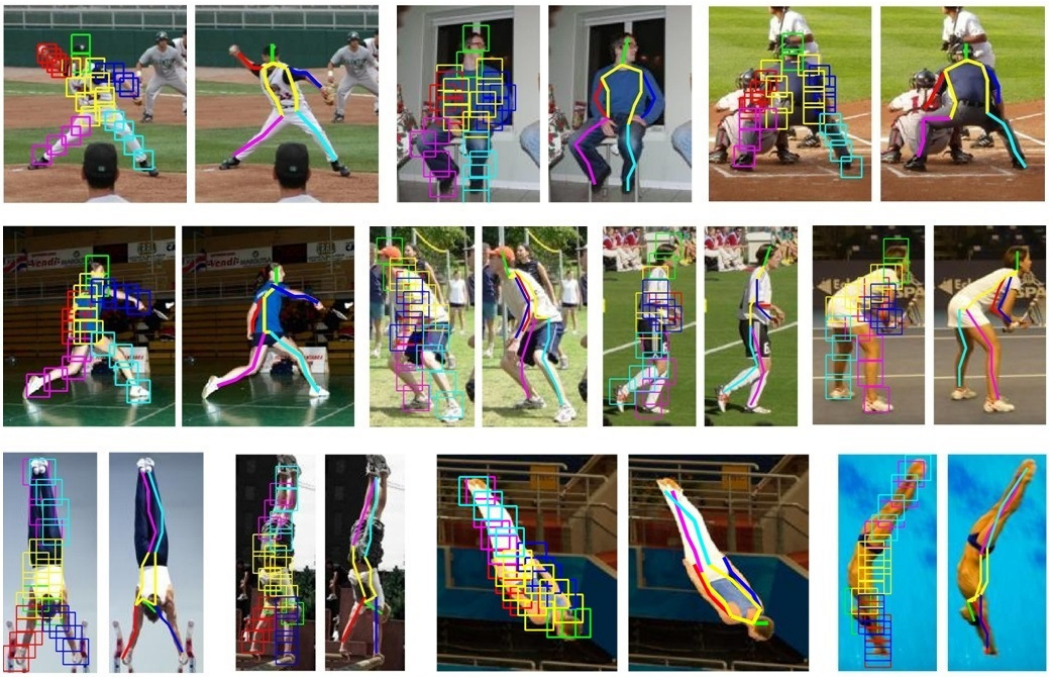


Figure 4.8: Successful example of human pose estimation on the LSP dataset. We show the bounding boxes for each body part (left) as well as the skeletons computed from bounding boxes (right). The first row denotes the detection in near front-back view with a 26 part model. The second row is the results for the side view with a 24 part model. The third row shows the results of the handstand view with a 26 part model.

in table 4.1, the performance of the proposed method is better as the detection accuracy rate is 4.55% higher than that in Ref. 80. Therefore, it can be concluded that our pairwise based upper body model is more effective and robust than the previous model in Ref. 80.

4.4.2 Evaluation for pose estimation

A comprehensive evaluation of the MMP model for human pose estimation is presented in this subsection. Let us describe the datasets for training and testing: the LSP dataset and the UIUC people dataset. The LSP dataset contains 2000 images collected from various human activities. Three subsets from 1000 training images are used for three categories of MMP models. The performance of the proposed model is tested with different number of mixtures, which helps to find the most effective one. The UIUC people dataset has 346 images for training and 247 images for testing. In this dataset, only two categories are used: side-view and near front-back view.

Results on LSP. The qualitative results of the MMP model on LSP datasets

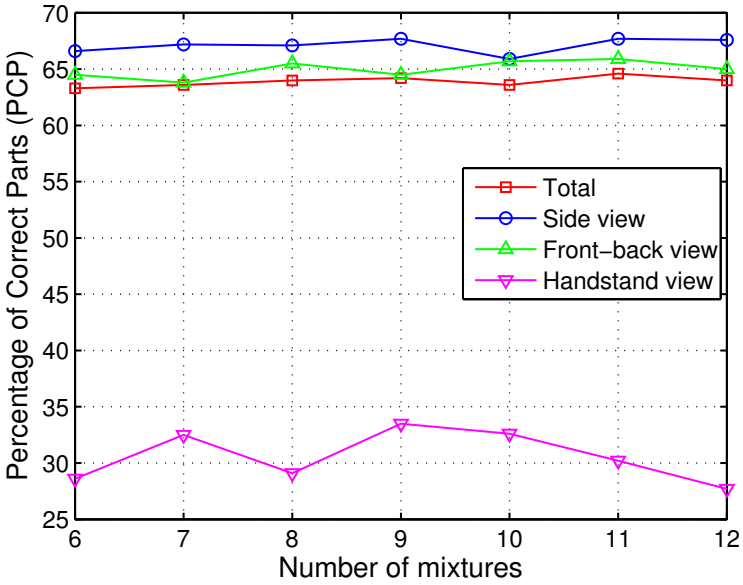


Figure 4.9: Comparison of the PCP performance for pose estimation with different numbers of mixtures. We test it from the 6 mixtures to 12 mixtures. The detection results include total, side view, near front-back view and handstand view. The total denotes the results composed of these three views.

are illustrated in Fig. 4.8, which displays the results in pairs. The left image denotes the detection of different models, while the right one shows the skeleton. The results are also shown in different categories: detection of side view, near front-back view and handstand view.

Evaluation measure. Different evaluation measures have been reported in Ref. 1, 67, 125, 140. Among these measures, Percentage of Correct Parts (PCP) is broadly-adopted evaluation protocol with two different definitions. The first declares a part as detected if the distance between the average of the predicted endpoints and the average of the ground truth endpoints is within 50% of the length of corresponding ground truth limb. The second is a stricter definition declaring that the distance between both of the predicted endpoints and the ground truth endpoints is within 50% of the length of corresponding ground truth limb. For fair comparison, all the evaluation criteria remain the same as in Ref. 125 which uses the second definition.

Different numbers of mixtures for each category. The MMP model was tested on the LSP dataset with different numbers of mixture parts. Its performance is illustrated in Fig. 4.9, it starts from 6 mixture parts to 12 mixture parts for total evaluation and each view based evaluation. Overall, the PCP values of side view and near front-back view are slightly higher than total results, while the PCP values of handstand view are lower than that of total. This can be explained by the fact

Dataset	Train			Test	PCP						
	Front	Side	Hand-stand	-	Torso	Head	U. Leg	L. Leg	U. Arm	L. Arm	Total
Subset 1	500	400	100	1000	94.5	86.9	72.05	62.45	57.95	39.75	64.6
Subset 2	1000	700	300	1000	91.0	85.2	73.6	63.2	61.55	42.1	65.7
Subset 3	1400	1000	600	1000	92.9	86.4	74.6	64.1	61.05	42.85	66.4
Subset 4	1800	1300	900	1000	92.3	86.7	74.5	64.0	62.05	44.05	66.8
Subset 5	2200	1600	1200	1000	91.8	85.6	74.7	63.6	62.95	44.8	66.9

Table 4.2: Performance on the LSP dataset with different numbers of training samples (Person-Centric annotations).

that the training dataset does not have many images in handstand view and the kinematic constraints are more complex. The PCP results ascend with fluctuation and reach a peak at 11 mixtures for the side view, near front-back view and total results, then decrease slightly. For the handstand view, the PCP reaches a peak at 9 mixtures. By analyzing the trend of this figure, we can conclude that PCP is not monotonously increasing with the growth of mixtures. Finding the peak for each model improves the performance.

Different numbers of training samples. The proposed MMP model is trained with different numbers of samples in the LSP dataset and the LSPET dataset (Leeds Sports Pose Extended Training Dataset) [126]. Both the training and test data use Person-Centric (PC) annotations. The total training samples include 1000 images from the LSP training dataset and 4000 images from the LSPET dataset. These images are divided into 5 subsets: 1∼1000, 1∼2000, 1∼3000, 1∼4000, 1∼5000. All the LSP training images are used in each subset. Then each subset is manually divided into three categories: near front-back view, right-left side view and handstand view. The MMP model is trained on each subset and tested on the LSP dataset (1000 test images), as shown in table 4.2. Compared to subset 1, the PCP results of subset 2 increase significantly, while the results of subset 5 are only slightly higher than those of subset 4.

Comparison with the state of the art. Let us compare the detection accuracy of the MMP model with the state-of-the-art model on the LSP dataset and the UIUC people dataset. Table 4.3 summarizes the evaluation results and highlights the highest scores. Compared with Yang & Ramanan [1], our results are better. The detection accuracies of the proposed model on Upper Arm (57.95%), Low Arm

Dataset	Method	Torso	Head	U. Leg	L. Leg	U. Arm	L. Arm	Total
LSP	Yang & Ramanan [1]	92.6	87.4	66.4	57.7	50.0	30.4	58.9
	Johnson & Everingham [127]	78.1	62.9	65.8	58.8	47.4	32.9	55.1
	Tian et al. [69]	**95.8**	**87.8**	69.9	60.0	51.9	32.8	61.3
	Johnson & Everingham [126]	88.1	74.6	**74.5**	66.5	53.7	37.5	62.7
	Pishchulin et al. [75]	88.7	85.1	63.6	58.4	46.0	35.2	58.0
	Fang & Yi [66]	91.9	86.0	74.0	**69.8**	48.9	32.2	62.8
	Ours	94.5	86.9	72.05	62.45	**57.95**	**39.75**	**64.6**
UIUC	Wang et al. [64]	86.6	68.8	56.3	50.2	30.8	20.3	47.0
	Tian et al. [69]	**98.8**	**96.8**	**78.7**	64.2	62.2	39.5	68.5
	Ours	97.57	95.95	78.34	**64.98**	**66.19**	**49.19**	**71.1**

Table 4.3: Performance on the LSP dataset (Person-Centric annotations) and the UIUC people dataset. The first 7 rows are PCP results of different algorithms where the training and the testing are both from the LSP dataset, while the last 3 rows show PCP results on UIUC people dataset.

Dataset	Method	Torso	Head	U. Leg	L. Leg	U. Arm	L. Arm	Total
LSP	Yang & Ramanan [1]	84.1	77.1	69.5	65.6	52.5	35.9	60.8
	Kiefel & Gehler [141]	84.4	78.4	74.4	67.1	53.3	27.4	60.7
	Tian et al. [69]	86.2	80.1	74.3	69.3	56.5	37.4	64.3
	Ramakrishna et al. [76]	88.1	80.9	78.9	73.4	62.3	39.1	67.6
	Pishchulin et al. [75]	87.5	78.1	75.7	68.0	54.2	33.9	62.9
	Fang & Yi [66]	90.9	84.6	79.2	71.3	61.9	35.0	67.0
	Ours	**93.90**	**88.10**	**82.30**	**74.70**	**69.55**	**49.35**	**73.4**

Table 4.4: Training and testing on the LSP dataset with Observer-Centric(OC) annotations.

(39.75%), and total (64.6%) are higher than other models. Compared with Fang & Yi [66], the proposed method outperforms it in four out of six joints, and the total accuracy is 1.8% higher. On the UIUC people dataset, the total accuracy of the MMP model is 2.6% higher than the method of Tian et al. [69]. Therefore, it can be concluded that the proposed MMP model is more effective.

The LSP dataset has different annotations: Person-Centric (PC) and Observer-Centric(OC). Usually, the methods using OC annotations are better than those based on PC annotations. As shown in table 5.1, we compared the MMP model with others using OC annotations. The proposed method marginally outperformed the models in Ref. 76, and the total accuracy is 5.8% higher. Moreover, we trained and tested the proposed MMP model with OC annotations and PC annotations respectively, and compared it with the method in Ref. 66. As shown in table 4.5, the results demonstrate that the MMP model performs better with OC annotations.

Qualitative evaluation. Successful results of the proposed model are shown in Fig. 4.8, the MMP model already achieves good results as it is able to cope with highly variable part appearances. We also show some examples of failures in Fig. 4.10. Figure 4.10 (a) is caused by double counting, which is a typical failure in model based methods. Figure 4.10 (b) presents the failure caused by occlusion, while Fig. 4.10 (c) is possibly caused by the failure selection of the MMP model. Therefore, more effective selection will be helpful for further improving the performance. Even with these failures, the MMP model outperformed the state-of-the-art methods in Ref. **?**, **?**, 66, 75, 76.

Computation. The computation complexity is presented by using the proposed model on the LSP dataset. There are 1000 images for testing. The running time for testing is approximately 5.4s per image on a Linux 64 bit OS using Core i7

Dataset	Train	Test	Method	Torso	Head	U. Leg	L. Leg	U. Arm	L. Arm	Total
LSP	PC	PC	Fang & Yi [66]	91.9	86.0	74.0	69.8	48.9	32.2	62.8
			Ours	**94.5**	86.9	72.05	62.45	57.95	39.75	64.6
	PC	OC	Fang & Yi [66]	92.9	85.4	70.6	59.1	53.1	37.9	62.0
			Ours	94.40	86.80	72.10	63.10	60.15	40.80	65.3
	OC	OC	Fang & Yi [66]	90.9	84.6	79.2	71.3	61.9	35.0	67.0
			Ours	93.90	**88.1**	**82.3**	**74.7**	**69.6**	**49.4**	**73.4**
	OC	PC	Fang & Yi [66]	90.9	84.6	67.2	57.6	46.5	26.8	57.2
			Ours	93.80	87.20	69.10	58.40	52.75	37.90	61.7

Table 4.5: Performance on the LSP dataset with PC annotations and OC annotations respectively.

(a) (b) (c)

Figure 4.10: Failure examples of the MMP model in the LSP dataset.

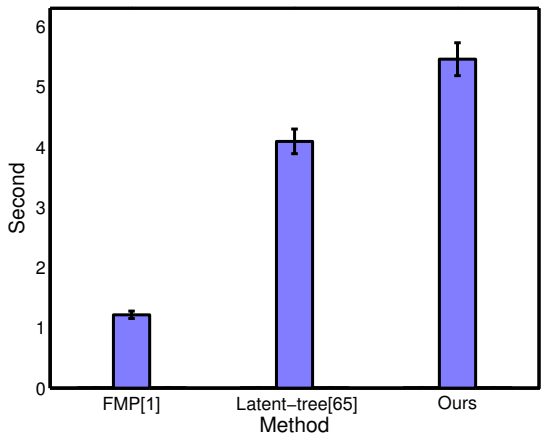


Figure 4.11: Comparison of the computation complexity on the LSP dataset. The proposed model has higher computation times.

2.6G CPU. The implementation is a mix of C++ and non-optimized Matlab code. We also compared the MMP model with the methods in Ref. 1 and Ref. 66. As illustrated in Fig. 4.11, the proposed model has higher computation times. This is due to the categorization of upper body and combined models in the MMP model.

4.5 Conclusion

In this thesis, we addressed the problems in articulated human pose estimation using pictorial structure models, and proposed a new two-stage estimation framework. The proposed framework divides the problem into several subproblems: the first is finding more discriminative features to distinguish different activities; the second subproblem is building a different model for each pose category; the third is comparing the performance of models in each category with different numbers of mixtures and show the most challenging pose. In other words, the performance of pose estimation can be improved in each subproblem. We trained the proposed MMP model with different numbers of samples, and tested it on different datasets. Empirical results suggest that the upper body based MMP model is more effective and outperforms the state of the art in human pose estimation.

Chapter 5

Pose estimation with deep convolutional neural network

Contents

5.1 Introduction

Over the last few years, deep learning techniques have made tremendous progress, especially in the field of computer vision. Deep learning is part of a broader family of machine learning that uses deep architectures to learn high-level feature representation. The deep architecture means that the network has more than one hidden layer. The essence of deep learning is attempt to compute hierarchical features or representations of the observational data, where the higher-level features and concepts are defined in terms of lower-level ones.

Active researchers in this area include those at University of Toronto, New York University, University of Montreal, University of Oxford, Stanford University, UC Berkeley, Google, Microsoft Research, Facebook, just to name a few. These researchers have demonstrated the successes of deep learning in diverse applications of computer vision, object detection and recognition, voice and image search, speech and image feature coding, robotics, and so on.

Convolutional Neural Network (CNN) is very important method in the family of deep learning. A Convolutional Neural Network (CNN) is comprised of one or more convolutional layers and then followed by one or more fully connected layers. This architecture allows CNNs to take advantage of the 2D structure of an input image (or other 2D input such as a speech signal). This is achieved with local connections and tied weights followed by some form of pooling which provides translation invariant features. Another benefit of CNNs is that they are easier to train and have much fewer connections and parameters due to the local-connectivity and shared-filter architecture in convolutional layers.

5.2 Deep convolutional neural network

The concept of deep convolutional neural network originated from artificial neural network research. The history of artificial neural network is filled with individuals from many different fields, including psychologists and physicists. Neural networks experienced different periods of hypes in the 1940s and 1980s/90s. In 1943 [142], McCulloch and Pitts created a computational model for neural networks based on mathematics and algorithms. This model paved the way for neural network research to split into two distinct approaches. In the late 1940s Hebb [143] created a hypothesis of learning based on the mechanism of neural plasticity that is now known as Hebbian learning. The first practical application of artificial neural network came with the invention of the perceptron network and associated learning rule by Rosenblatt in the late 1950s [144]. During the 1980s research in neural networks increased dramatically. Firstly, in 1982, Hopfield [145] used statistical mechanics to explain the operation of neural networks. Another key advance that came later was the backpropagation algorithm, a generalized form of the delta rule, for training multi-layer perceptron networks. Rumelhart and McClelland [146] provided a full exposition of the use of connectionism in computers to simulate neural processes and showed that it is effective for the class of semi-linear activation functions. These

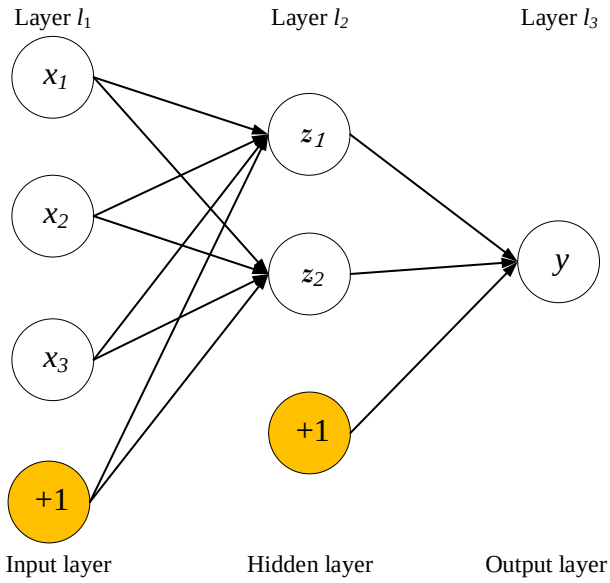


Figure 5.1: A simple feed-forward neural network.

new development reinvigorated the field of neural networks. For the last decade, neural networks have been celebrating a comeback under the term deep learning, taking advantage of many hidden layers in order to build more powerful machine learning algorithms [85]. In this section, I provides a detail introduction of both neural networks as well as Convolutional Neural Network (CNN).

5.2.1 Feed-forward neural networks

Feed-forward neural networks are the simplest type of neural networks, including an input layer, one or more hidden layers and an output layer(as shown in Fig. 5.1). Consider a supervised learning problem where we have access to labeled examples $(x^{(i)}, y^{(i)})$. Here, x^i denotes the feature and y^i denotes the label of input example i. Neural networks give a way of defining a complex, non-linear function $h_W(x)$. This function is parameterized by a weight matrix W. The network in Fig. 5.1 consists three units or neurons in the input layer (not counting the bias uint labeled by '+1'), denoted x_1, x_2, x_3, and two z_1, z_2 in the hidden layer. Aside from the neurons in the input layer, each neuron in the current layer that takes the values of the neurons in the preceding layer as input. As shown in Fig. 5.1, the inputs to the neuron z_1 is x_1, x_2, x_3 and the input to y is z_1 and z_2. Given its inputs, we can first compute:

$$a_j^{(l_2)} = \Sigma_{i=1}^{3} \omega_{ji}^{(l_1)} x_i + b_j^{(l_1)}, \tag{5.1}$$

where $a_j^{(l_2)}$ denote the total weighted sum of inputs to unit j in layer l_2, including the bias value $b_j^{(l_1)}$ of unit j. $\omega_{ji}^{(l_1)}$ is a parameter describing the interaction between neuron z_j in the layer l_2 and the input neuron x_i in the layer l_1. While a nonlinear

activation function is used to $a_j^{(l_2)}$, the activation or value of the unit j in layer l_2 is defined to be:

$$z_j^{l_2} = f(a_j^{(l_2)}) = f(\Sigma_{i=1}^{3}\omega_{ji}^{(l_1)}x_i + b_j^{(l_1)}), \tag{5.2}$$

where $f(\cdot)$ denotes the activation function. Here, the sigmoid function is use as the activation function in the network,

$$f(a_j^{(l_2)}) = \frac{1}{1+\exp(-a_j^{(l_2)})} \tag{5.3}$$

The activation of the output layer can be defined as:

$$y = h_{\omega,b}(x) = z_j^{l_3} = f(\Sigma_{i=1}^{2}\omega_{ji}^{(l_2)}z_i^{l_2} + b_j^{(l_2)}) \tag{5.4}$$

It should be noted that this network has only one unit in the output layer ($j = 1$). Given a set of the input variables x and the parameters W, b, we can compute the activation of each neuron in the hidden or output layers by the above steps. Since the activation of each neuron depends only upon the values of neurons in preceding layers, we compute the activations starting from the first hidden layer and proceed it through the network. Thus, this process is called the forward-propagation step. A set of outputs y can be used as the classification results of the input x.

Given a fixed training set $(x^{(i)}, y^{(i)})$, the objective is to learn the parameters ω, b in the neural network by minimizing some objective or cost function. There are different cost functions, such as the least squares or cross-entropy cost function, described in [147]. The latter one has been reported in [148], which is able to generalize better and speed up learning.

5.2.2 Deep Convolutional neural network

Convolutional neural networks (CNNs) are a type of feed-forward artificial neural network. At the most basic level, the CNN is a multilayer, hierarchical neural network. They are made up of neurons that have learnable weights and biases. Each neuron receives some inputs, performs a dot product and optionally follows it with a non-linearity. CNNs take advantage of the fact that the input consists of images and they constrain the architecture in a more sensible way. Specifically, unlike a classic feed-forward neural network, the layers of a CNN have neurons arranged in 3 dimensions: width, height, depth. It should be noted that the 'depth' refers to the third dimension of an activation volume. As shown in Fig. 5.2, the input image is an input volume of the activation, and the volume has dimensions 40x40x3 (Here 3 is a value of the depth). There are three main types of layers to build DCNN architectures: Convolutional Layer, Pooling Layer, and Fully-Connected Layer. In the following, we will introduce these layers detailedly.

5.2.2.1 Convolutional Layer

The Convolutional layer is the core building block of a DCNN. The parameters of convolutional layer consist of a set of learnable filters (or kernels). Every filter have

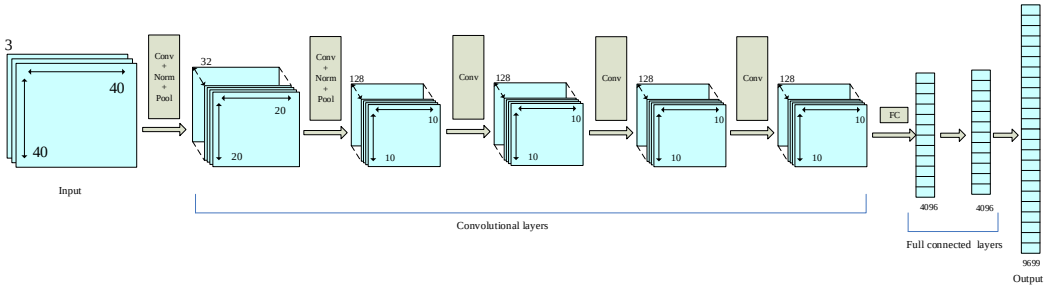


Figure 5.2: Deep Convolutional neural network.

a small receptive field, but extend through the full depth of the input volume. In the forward pass, each filter is slid across the width and height of the input volume, computing the dot product between the entries of the filter and the corresponding input and producing a feature map of that filter. As a result, the network learns filters that activate when they match a specific feature at a given spatial position of the input.

In the CNN architecture, it exploits spatially-local correlation by constraining each neuron to depend on a local subset of the neurons in the previous layer. In other words, each neuron in layer l is connected to only a local region of in layer $l-1$. The spatial extent of this connectivity is a hyperparameter called the receptive field of the neuron. In addition, in CNNs, each filter is replicated across the entire visual field. Thus, the weights are shared across multiple neurons in a hidden layer by evaluating the same filter over multiple subwindows of the input images. As show in Fig. 5.3, the input size is $w_i * h_i * d_i = 7*7*3$ and is padded with 1. The number of feature maps to be learned d_0 is 2. The size of filter is $w_f * h_f * d_f * d_0 = 3*3*3*2$. The strides when filter is slid along width and height are both 2. Thus, the output has size of $3*3*2$. The number of parameters to be learned is $(3*3*3+1)*2 = 56$.

5.2.2.2 Pooling Layer

Another important concept of CNNs is pooling or subsampling. Its function is to progressively reduce the dimensionality of the convolutional responses and provide a form of translation invariance into the model. In spatial pooling [149], the convolutional response map is first divided into a set of $m * n$ blocks. Then a pooling function is used to evaluate the response in each block. In the case of max pooling, the maximum values in each block are the responses for the blocks. As shown in Fig. 5.3, the convolutional response map is a $4*4$ grid and the max pooling is used over four $2*2$ blocks. The pooled responses are taken to be the maximum of the values in each block.

5.2.2.3 Fully-Connected Layer

After several convolutional and max pooling layers, the intuitive reasoning behind these layers in the neural network is done via fully connected layers. Neurons in a

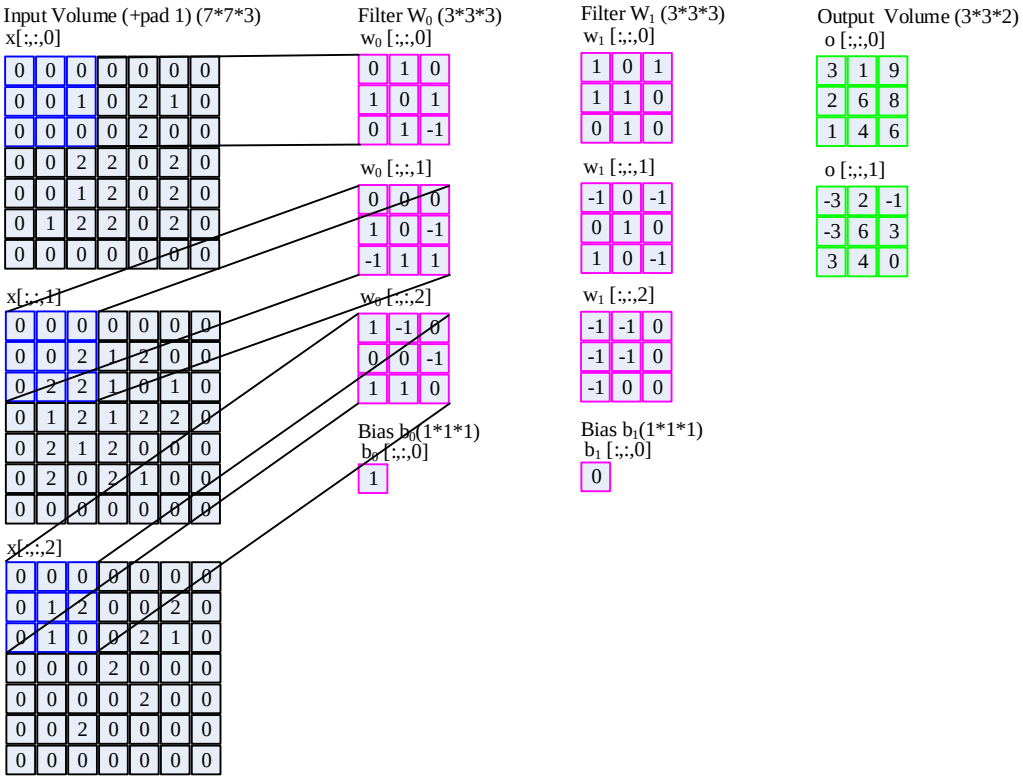


Figure 5.3: Convolutional Layer.

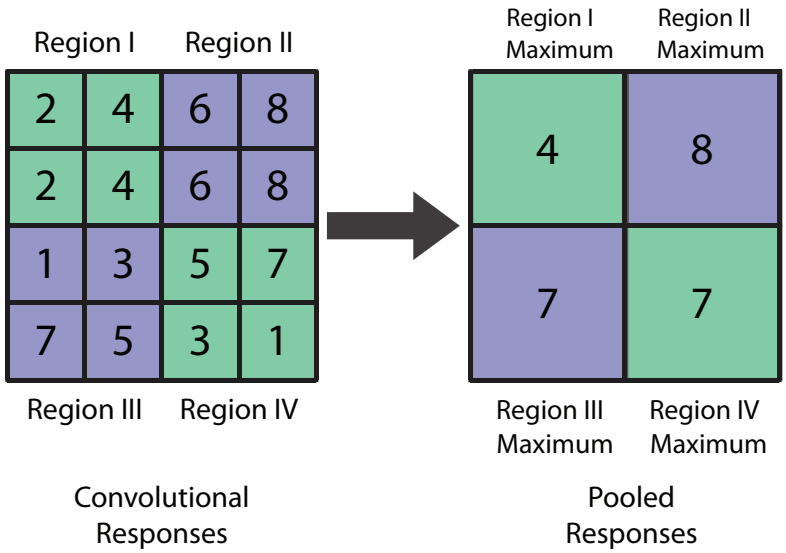


Figure 5.4: Max pooling Layer in a convolutional neural network.

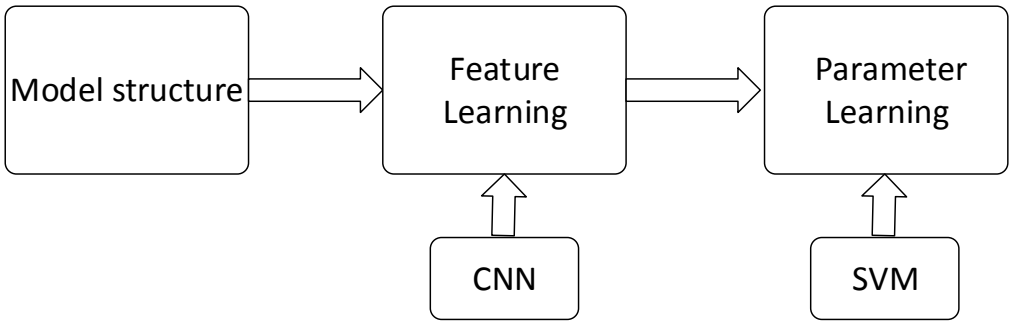


Figure 5.5: Framework for estimating human poses.

fully connected layer have full connections with all neurons in the previous layer, as seen in Fig. 5.2.

5.3 Model

The articulated human pose estimation can be divided into tree main parts, including model structure, feature learning, and parameter learning (see the Fig. 5.5). we will first introduce the proposed graphical model and learning procedure of our model.

5.3.1 Graphical model

Appearance model. The proposed human model is also based the pictorial structure model that has show its performance in many works. Here, we simplify the Multiple Mixture Part models (MMP) to a single mixture part model by using more filters of each body part. As shown in Fig. 5.6, it contains 18 joint parts and 8 limb parts. Limb parts contain two upper arms, two lower arms, two upper legs and two lower legs. Different size of body parts are merged together to efficiently represent human pose. Generally, the limb parts are larger than joint parts, and have more context information. Thus, we can rewrite the appearance term in MMP model as:

$$S_a(I,P) = \sum_{i^* \in V^*} \omega_{i^*}^{m_{i^*}} \cdot \phi(I, p_{i^*}) + \sum_{k^* \in V_l^*} \omega_{k^*}^{m_{k^*}} \cdot \phi(I, p_{k^*}), \tag{5.5}$$

where $V^* = \{1, \ldots, N\}$ is a set of joint parts and $V_l^* = \{1, \ldots, K\}$ is a set of limbs parts. It should be noted that V^* and V_l^* is subset of V, where $V = V^* \cup V_l^* = \{1, \ldots, K+N\}$. $K+N$ is the total number of human body part in the proposed model. p_{i^*} is the position of joint part i^* while p_{k^*} is the position of limb part k^*. $\omega_{k^*}^{m_{k^*}}$ denotes the parameter for limb part k^* with mixture type m_{k^*}, while $\omega_i^{m_i}$ is for the joint part.

Deformable model. The deformable model is used to predict the relative spatial positions between each pair of body parts in the graphical model. These pairs

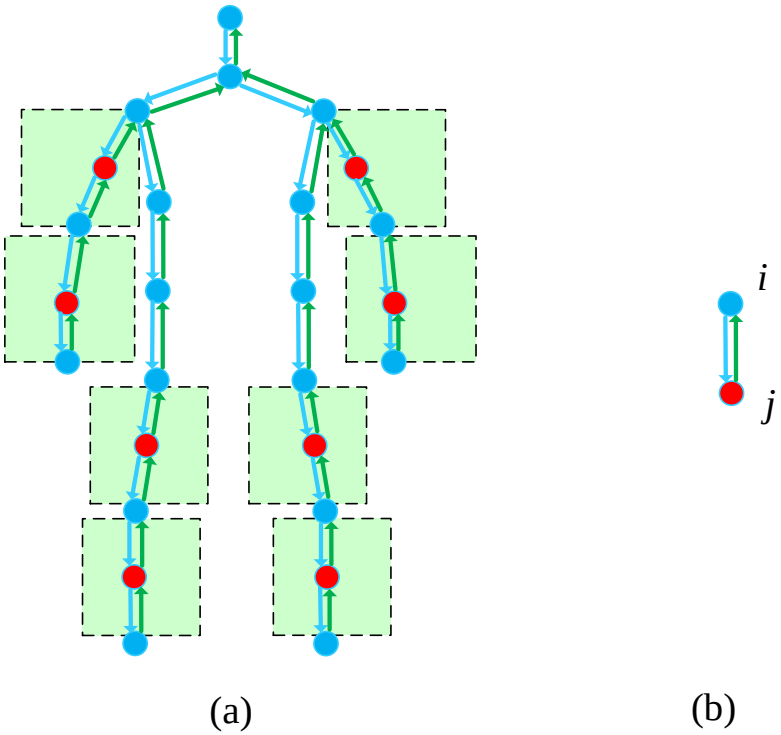


Figure 5.6: Graphical Model for estimating human poses. (a) is the graphical model: the red nodes denote centers of limb parts, and the cyan nodes denote centers of joint parts. The green boxes show the large regions of the limb parts. Each pair connected parts have the relative deformable information. (b) is a pair connected relationship between part i and part j. The cyan edge with an arrow denotes the relative mixture m_{ij}, while the green one denote the relative mixture m_{ji}.

of parts mix joint parts and limb parts. Here, the *relative spatial information* is defined by each pair of connected parts. As shown in Fig. 5.6 (b), two edges denotes the relative deformable information between part i and part j. For example, the neighboring parts of a left knee are left upper leg and lower leg, and the left upper leg is parent part of the left knee while the lower leg is the child part. Similar to MMP, the proposed model also contains mixture information. Thus, this mixture information is merged into the relative deformable information forming the *Relative Mixture Deformable Model* (RMDM). In order to make use of the mixture information of each pair of parts in RMDM, the pairwise locations between part $i \in V$ and part $j \in V$ are disctretized into a relative mixture $m_{ij} \in \{1, \ldots, M_{ij}\}$ and a relative mixture $m_{ji} \in \{1, \ldots, M_{ji}\}$, where $V = \{1, \ldots, K+N\}$ is a set of all parts including joint parts and limb parts. Both relative mixtures are corresponding to a mean relative location $r_{ij}^{m_{ij}}$ or $r_{ji}^{m_{ji}}$ that is computed from the all training data. Thus, the deformable term can be rewrite as:

$$\begin{aligned} S_d(I,P) = & \sum_{i,j \in E} \omega_{ij}^{m_{ij}} \cdot \psi(p_i + r_{ij}^{m_{ij}} - p_j) + \sum_{i,j \in E} \omega_{ij} \phi(I, p_i, m_{ij}) \\ & + \sum_{j,i \in E} \omega_{ji}^{m_{ji}} \cdot \psi(p_j + r_{ji}^{m_{ji}} - p_i) + \sum_{j,i \in E} \omega_{ji} \phi(I, p_j, m_{ji}), \end{aligned} \tag{5.6}$$

where part i, j are pairwise connected in E. E is a set of links between two of 26 parts (8 limb parts and 18 joints parts). $\omega_{ij}^{m_{ij}}, \omega_{ij}, \omega_{ji}^{m_{ji}}, \omega_{ji}$ are the weight parameters. $\phi(I, p_i, m_{ij})$ denotes the feature for the part i with the relative mixture (type) m_{ij}, while $\phi(I, p_j, m_{ji})$ is for the part j with the mixture m_{ji}. $\psi(p_i + r_{ij}^{m_{ij}} - p_j)$ is the standard quadratic deformation feature, where the relative location of part i with respect to part j. Thus, $\psi(p_j + r_{ji}^{m_{ji}} - p_i)$ is the deformation feature for the relative location of part j with respect to part i.

5.4 Deep hierarchical model based on local multi-resolution convolutional neural network

5.4.1 Local multi-resolution convolutional neural network

The standard convolutional network has shown its performance in many fields. A DCNN mainly consists of two parts: convolutional layers, and fully-connected layers. The convolutional layers operate in a sliding-window manner and output features maps which represent the spatial arrangement of the activations. In fact, convolutional layers do not have the limitation of a fixed image sizes and can generate feature maps of any sizes. On the other hand, the fully-connected layers need to have fixed-size input by their definition. Thus, the input image patches need to keep in the same size. This leads to the problem that how to determine the input patch sizes of each body part during the training phrase. There is a common method to solve this problem by resizing all patches of the same body part from all training data. This method does not take account of the fact that the image patch sizes of single body part are different in each training example. On the other

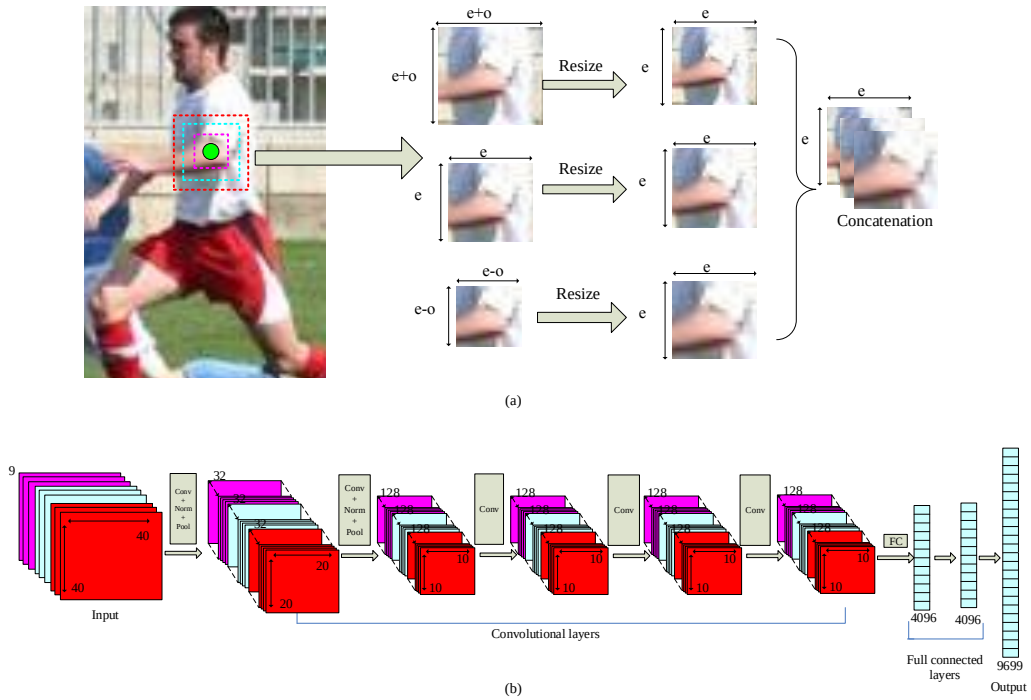


Figure 5.7: Local multi-resolution convolutional neural network. (a) shows the concatenation of different scales of a boy part to increase the channel of input data. (b) The concatenations are used as input data and processed by convolutional layers and full connected layers (Input scale $e = 40$).

hand, the single patch size cannot provide enough context information. Motivated by these, a Local Multi-Resolution Convolutional Neural Network (LMR-CNN) is proposed. As shown in Fig. 5.7 (a), it contains three-level scales of the elbow joint $((e + o) * (e + o), e * e, (e - o) * (e - o))$. o is defined as the offset in this multiple scales. e is the center scale of the input joint parts in LMR-CNN. We first crop these three-level patch from the given image, and resize each patch into the same size $e * e$. Then, these patches are concatenated together. Due to an RGB image has 3 channels, the concatenation of three patches has 9 channels. It should be noted that these three-level scales can be extend to more levels. Fig. 5.7 (b) presents the pipeline of the proposed LMR-CNN that takes the concatenation of patches as the input data. There are 5 convolutional layers to output three-level features that have different levels of the context. The input of fully-connected layer is all the convolutional features. Thus, the proposed LMR-CNN not only provide enough context information, but also capture the local feature.

5.4.2 Deep hierarchical limb model for pose estimation

In traditional models, all body parts share one part scale. Due to the small scale of joint parts, it does not have to be semantically meaningful and can not capture

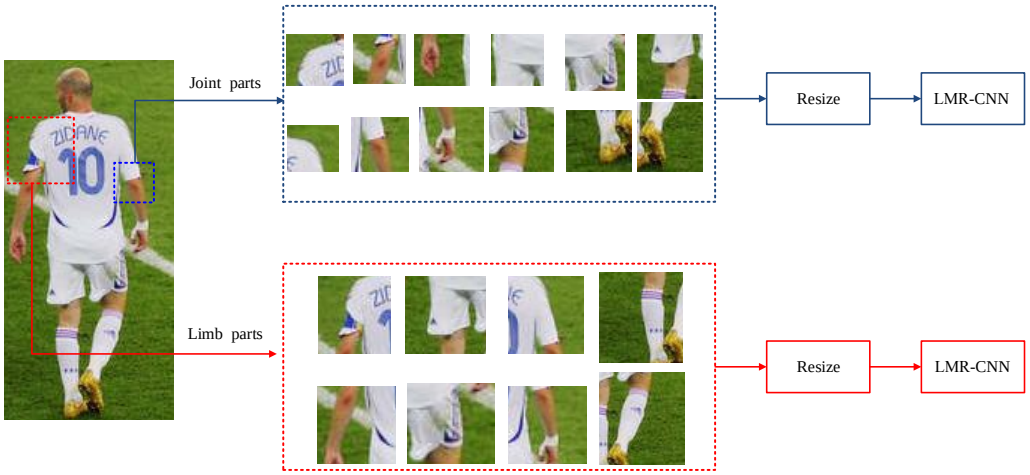


Figure 5.8: Deep hierarchical limb model. The blue box shows joint parts and the red one shows limb parts.

different granularity of details. The poselet models have the limitation of building detectors of non-rigid limbs inaccurate as they are variable in appearance. A hierarchical model can be seen as bridging the gap between these two popular approaches for pose estimation. Comparing to a traditional hierarchical model, the proposed model focuses on the limbs that are more complex in the structure and appearance. Here, the hierarchical limb means that the variant sizes of appearance model between limb parts and joint parts. Moreover, the deep learning architecture is introduced in hierarchical model for pose estimation. Thus, we call this method as deep hierarchical limb model. As shown in Fig. 5.8, it contains two levels of parts: joint parts and limb parts. Concerning joint parts, the parameters of the LMR-CNN are similar to Fig. 5.7. Nevertheless, the limb parts are larger than joints parts and cannot share one network together with joint parts. Thus, an extra network is used to training limb parts and is called *Limbnet*. The input of *Limbnet* also contains three-level scales of the elbow joint: $(e'+o)*(e'+o), e'*e', (e'-o)*(e'-o)$. e' is the center scale of limb parts in LMR-CNN.Three-level patches from the given image are resized into the same size $e'*e'$. The joint parts and limb parts share the offset o.

In the proposed deep model, the LMR-CNN is used to learn the feature instead of using HOG filters. Thus, the appearance model is based on the local image patch $I(p_{i^*})$ at the location p_{i^*} of part i. The feature $\phi(I, p_{i^*}), \phi(I, p_{k^*})$ in Eq. A.24 can be defined as:

$$S_a(I,P) = \sum_{i^* \in V^*} \omega_{i^*}^{m_{i^*}} \cdot f(i^*|I, p_{i^*}; \theta) + \sum_{k^* \in V_l^*} \omega_{k^*}^{m_{k^*}} \cdot f(k^*|I, p_{k^*}; \theta'), \tag{5.7}$$

where f is the conditional probability distribution of parts i^*, k^* learned by LMR-CNN. θ denote the parameters learned based on joint parts, while θ' are the parameters of limb network. For the deformable model, the Eq. A.25 can be rewritten

as:

$$S_d(I,P) = \sum_{i,j\in E} \omega_{ij}^{m_{ij}} \cdot \psi(p_i + r_{ij}^{m_{ij}} - p_j) + \sum_{i,j\in E} \lambda_{ij} f(m_{ij}|I,p_i,i;\theta,\theta') \\ + \sum_{j,i\in E} \omega_{ji}^{m_{ji}} \cdot \psi(p_j + r_{ji}^{m_{ji}} - p_i) + \sum_{j,i\in E} \lambda_{ji} f(m_{ji}|I,p_j,j;\theta,\theta'), \quad (5.8)$$

where $f(m_{ij}|I,p_i,i;\theta,\theta')$ denotes the distribution for the part i with the relative mixture m_{ij} depended on the parameters θ,θ', while $f(m_{ji}|I,p_j,j;\theta,\theta')$ for part j. λ_{ij} is the weight parameter.

5.5 Inference and learning

5.5.1 Inference

Given an image I, the inference problem is to find the optimal pose. Here, the optimal configuration is defined by maximizing the full score function:

$$p^* = \arg\max_p S_a(I,P) + S_d(I,P) + b, \quad (5.9)$$

where p^* denotes the locations of body parts, and b is the bias term for the root filter. The root scores are used to generate multiple detection in image I by thresholding them and applying non-maximum suppression (NMS). Then, a backtracking is used to find the location and the type of each part in each maximal configuration.

5.5.2 Learning

We consider the problem of learning the model parameters from images labeled with part positions. This is the type of data available in the pose estimation datasets [56, 125, 127, 128]. Each dataset contains thousands of images and each image has part location annotations.

The proposed model consists of three sets of parameters. First is the mean relative locations r that is learned by the K-means algorithm. Second is the parameter θ,θ' of the appearance term learned by LMR-CNN. Third is weight parameters ω learned by structured SVM.

During training, we have access to training images with ground truth joint/body locations $\{I^t,P^t,M^t\}_{t=1}^T$. T is the number of training images. I^t denotes the image with the ground truth value P^t, the mixture index M^t.

Parameters for LMR-CNN: In the training of a LMR-CNN, each local image patch $I(p_i^t)$ centered at an annotated part location p_i in the image example t. m_{ij}^t is the relative mixture between part i and part j in the image I^t and denotes the relationship between part i and its neighbor part j in the deformation model. Due to the tree structure, the part i could have more than one neighbor part. As shown in Fig. 5.9, joints like the wrist and ankle have one neighbor, joints like the elbows have two neighbors, and joints like shoulders have three neighbors. As for training

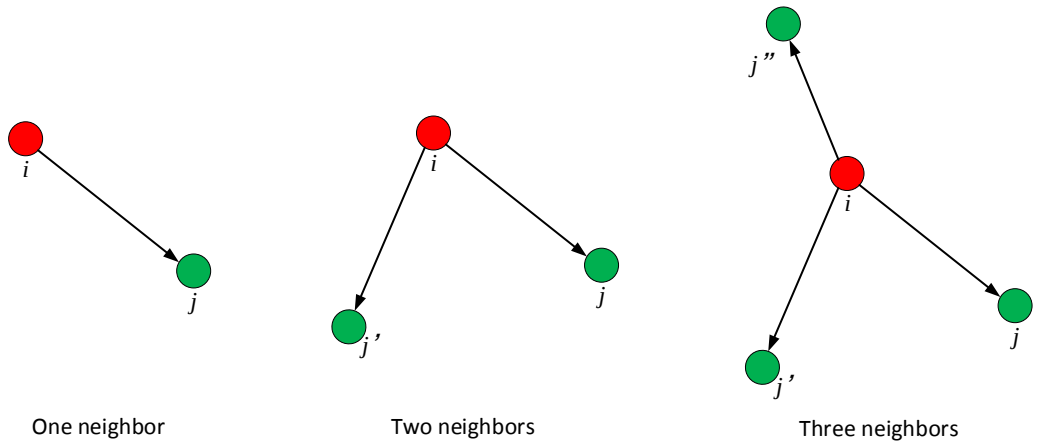


Figure 5.9: The different numbers of neighbors of the center joint. The red nodes denote the center joint, while the green nodes are the neighbor joints.

CNNs, we need to divide the types of part i depended on all the neighbor parts of the current part. This can ba defined as:

$$\begin{cases} \mathbb{C}_{i\mathbb{N}(i)} = \{1, \cdots, M_{ij}\}, & \mathbb{N}(i) = \{j\} \\ \mathbb{C}_{i\mathbb{N}(i)} = \{1, \cdots, M_{ij}\} \times \{1, \cdots, M_{ij'}\}, & \mathbb{N}(i) = \{j, j'\} \\ \mathbb{C}_{i\mathbb{N}(i)} = \{1, \cdots, M_{ij}\} \times \{1, \cdots, M_{ij'}\} \times \{1, \cdots, & M_{ij''}\}, \\ & \mathbb{N}(i) = \{j, j', j''\} \end{cases} \tag{5.10}$$

where $\mathbb{N}(i)$ is a set of neighbors of part i, $\mathbb{C}_{i\mathbb{N}(i)}$ denote a set of categories of part i depended on all its neighbors. In fact, $\mathbb{C}_{i\mathbb{N}(i)}$ are the combination of all the pairwise relative mixtures. $M_{ij}, M_{ij'}, M_{ij''}$ are the number of relative mixtures in each pairwise connection. j, j', j'' denote different neighbors of part i, if it has three neighbors.

As mentioned above, the part number is described by $i^t \in \{1, \cdots, K+N\}$. Thus each image has $K+N$ parts or patches. $c^t_{i^t\mathbb{N}(i^t)} \in \mathbb{C}^t_{i^t\mathbb{N}(i^t)}$ is the neighbor-based relative mixture of part i in the example t. In this way, we have access to training images with ground truth body locations that are labeled as a set of patches $\mathbb{P} = \{I(p^t_i), i^t, c^t_{i^t\mathbb{N}(i^t)}\}^{i=K+N,t=T}_{i=1,t=1}$. It should be noted that this set $\mathbb{P}$ can be divided into two subsets $\mathbb{P}_1, \mathbb{P}_2$ for training the LMR-CNN models of joint parts and limb parts respectively. Similar to other methods of the CNN training, the mixture numbers of the background patches from negative examples are labeled as 0. We performed standard batch stochastic gradient descent to train this multi-class LMR-CNN.

Weight parameter: To illustrate learning weight parameters in the proposed model, let us write $Z_t = (P_t, M_t)$. The score function can be expressed in terms of a dot product, $S(I_t, Z_t) = \beta \cdot \Phi(I_t, Z_t)$, between a vector of model parameters β and

a feature vector $\Phi(I_t, Z_t)$,

$$\beta = (\omega_1^{m_1}, \cdots, \omega_N^{m_N}, \omega_{N+1}^{m_{N+1}}, \cdots, \omega_{N+K}^{m_{N+K}}, \cdots, \omega_{ij}^{m_{ij}}, \cdots, \tag{5.11}$$
$$\cdots, \lambda_{ij}, \cdots, \cdots, \omega_{ji}^{m_{ji}}, \cdots, \cdots, \lambda_{ji}, \cdots, b), \quad (i, j \in E)$$
$$\Phi(I_t, Z_t) = (f(1|I, p_1; \theta), \cdots, f(N|I, p_N; \theta), f(N+1|I, p_{N+1}; \theta'), \cdots, \tag{5.12}$$
$$f(N+K|I, p_{N+K}; \theta'), \cdots, \psi(p_i + r_{ij}^{m_{ij}} - p_j), \cdots, \cdots, f(m_{ij}|I, p_i, i; \theta, \theta'), \cdots,$$
$$\cdots, \psi(p_j + r_{ji}^{m_{ji}} - p_i), \cdots, \cdots, f(m_{ji}|I, p_j, j; \theta, \theta'), \cdots, 1), \quad (i, j \in E)$$

where N is the number of joint parts and K is the number of limb parts. Thus, let us define a large-margin learning objective function similar to the work of Ref. 63, 138:

$$\arg \min_{\beta, \xi_t \geq 0} \frac{1}{2}\|\beta\|^2 + \mathcal{C} \sum_t \xi_t, \tag{5.13}$$
$$s.t. \forall t \in pos \quad \langle \beta, \Phi(I_t, Z_t) \rangle \geq 1 - \xi_t,$$
$$\forall t \in neg \quad \langle \beta, \Phi(I_t, Z_t) \rangle \leq -1 + \xi_t,$$

where $\mathcal{C}$ controls the relative weight of the regularization term and ξ_t denotes the slack variables of the objective function.

5.6 Experiment results

In this section, we report experimental results to evaluate the proposed approach in human pose estimation. We first introduce the benchmark datasets and the evaluation metrics used in this thesis, followed by the implementation details of our model. Then the results of our approach and comparisons with the state-of-the-art models are presented LMR-CNN.

5.6.1 Setup

5.6.1.1 Datasets

There are several state-of-the-art datasets in the human pose estimation. In this chapter, we evaluate the performance on the Leeds Sport Dataset (LSP) [127], Frames Labeled In Cinema (FLIC) [67]. Both datasets have large number of training examples that are sufficient to train a large model such as the proposed.

The first dataset we use is LSP dataset that contains 2000 images: 1000 training images and 1000 test images. These images are collected from various human activities and are quite challenging in terms of appearance and especially articulations. In this dataset, all images are used for the full body human pose estimation and are labeled with total 14 joints.

The second dataset we use is FLIC dataset that consists of 4000 training and 1000 test images. These images are obtained from popular Hollywood movies, and contain people in various poses. In this datase, each person is labeled with 10 upper body joints.

To train the proposed model, the images from the Inria Person dataset [61] are used as the negative training examples.

5.6.1.2 Evaluation Metrics

In order to be able to compare with published results we will use a widely accepted evaluation metric. Different evaluation measures have been reported in Ref. 1, 67, 125, 140. Among these measures, Percentage of Correct Parts (PCP) is broadly-adopted evaluation protocol with two different definitions. The first declares a part as detected if the distance between the average of the predicted endpoints and the average of the ground truth endpoints is within 50% of the length of corresponding ground truth limb. The second is a stricter definition declaring that the distance between both of the predicted endpoints and the ground truth endpoints is within 50% of the length of corresponding ground truth limb. For fair comparison, all the evaluation criteria remain the same as in Ref. 125 which uses the second definition.

On the FLIC dataset, Percentage of Detected Joints (PDJ) is used to measure the performance of pose estimation. In the PDJ metrics, a predicted joint is considered detected if the distance between the predicted and the ground truth joint is within a certain fraction of the torso diameter. By varying this fraction, the performance are measured using a curve of the percentage of correctly predicted joints for varying thresholds of localization precision.

5.6.1.3 Implementation detail

Our LMR-CNN has millions of parameters, while only several thousand of training examples are available in each dataset. Thus, it is proposed to do data augmentation by rotating the positive examples and horizontally flipping each image to double image samples. The random examples from the positive training example that are used as a validation set for the LMR-CNN. LMR-CNN is trained using local multi-resolution part patches. In the LSP dataset, the multiple scales of each joint part are $\{40*40, 48*48, 56*56\}$ pixels that are resized into the center scale $48*48$ pixels. Thus, the input patch size of each joint part in this multi-resolution network is $48*48$ pixels. For limb parts, the center scale is $e' = 60$. In the FLIC dataset, the multiple scales of each joint part are changed to $\{48*48, 56*56, 64*64\}$ pixels and centered at $56*56$ pixels. For limb parts, the center scale is $e' = 68$. The similar LMR-CNN architectures are used on both datasets except the input layer. The LMR-CNN is implemented within the Caffe [150] framework by the custom GPU implementations. Training the LMR-CNN takes approx. 4 days, the graphical model approx. 3 days. Our total pipeline requires approximately 17 seconds to process an image.

Each pairwise connection has the same number of relative mixture. Here, we set it as $M_{ij} = M_{ji} = 13$ on all datasets. Thus, the parts with one neighbor have 13 types, while the parts with two neighbors have 13^2 types. The proposed graphic model has 26 body parts on the LSP dataset and 18 body parts on the FLIC dataset.

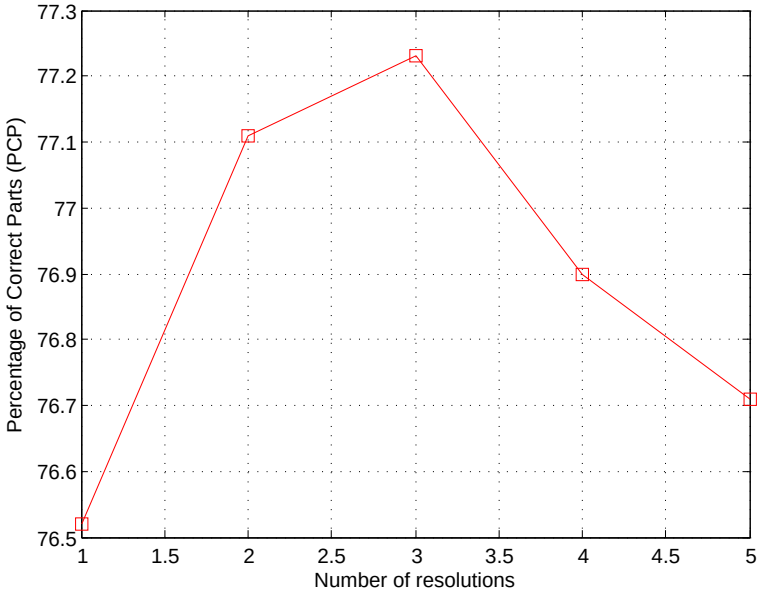


Figure 5.10: Comparison of the PCP performance for pose estimation with different numbers of resolutions.

.

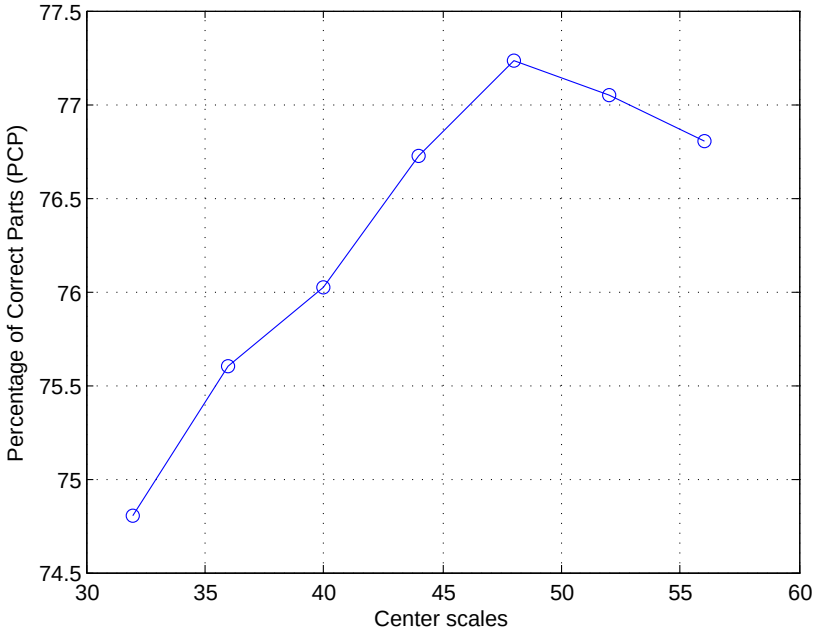


Figure 5.11: Comparison of the PCP performance for pose estimation with different center scales.

.

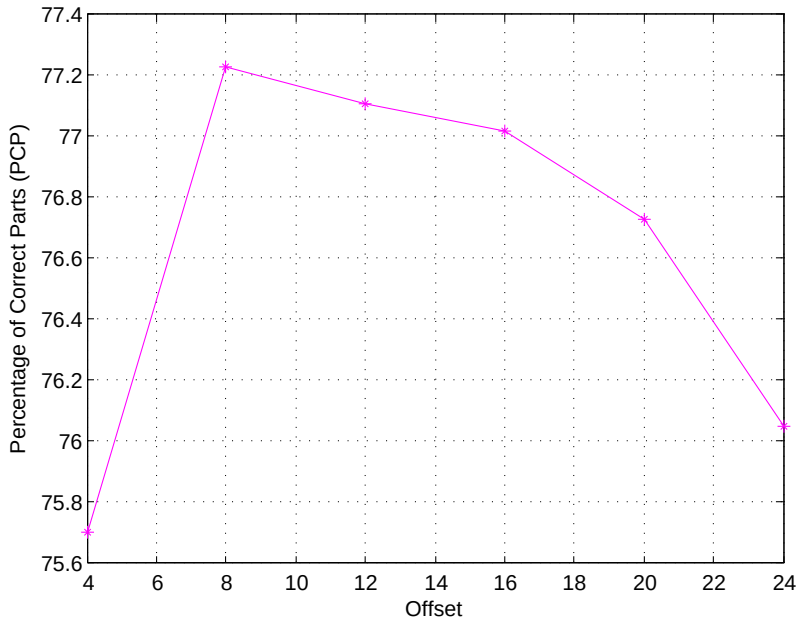


Figure 5.12: Comparison of the PCP performance for pose estimation with different offsets.

.

5.6.2 Diagnostic experiments for LMR-CNN

Here, we analyze how the network structure influences the model performance on the LSP datasets. There are three main factors: the number of resolutions of each local part, the center scale e and the offset o. As illustrated in Fig. 5.10, the model performance reaches a peak at 3 resolutions. In this case, the offset o and the center scale e are constants. We can conclude that PCP is not monotonously increasing with the growth of number of resolutions. Then, the effect of varying center scales e on the accuracy of pose estimation is considered in Fig. 5.11. It shows that the PCP results reach its peak at a center scale $e = 48$. It should be noted that the center scale e' is constant and larger than e. In this experiments, e' is defined as $e' = e + 12$ pixels. Finally, the offset o is analyzed in Fig. 5.12. Its performance at $o = 8$ is better than it at other offsets. This result demonstrate that the larger offset o can not enhance the performance of pose estimation in the proposed LMR-CNN.

5.6.3 Results and discussion

5.6.3.1 Comparisons

To demonstrate the effectiveness of the proposed LMR-CNN based model, we present comparative results to other approaches on the LSP dataset and the FLIC dataset. First the strict PCP metric is used for comparison on the LSP dataset. As presented in Table 5.1, the proposed method clearly outperform all other approaches,

Method	Torso	Head	U. Leg	L. Leg	U. Arm	L. Arm	Avg. Limb-s	Total
Yang & Ramanan [1]	84.1	77.1	69.5	65.6	52.5	35.9	55.9	60.8
Kiefel & Gehler [141]	84.4	78.4	74.4	67.1	53.3	27.4	55.6	60.7
Eichner & Ferrari [151]	86.2	80.1	74.3	69.3	56.5	37.4	59.4	64.3
Pishchulin et al. [152]	88.7	85.6	78.8	73.4	61.5	44.9	64.7	69.2
Ramakrishna et al. [76]	88.1	80.9	78.9	73.4	62.3	39.1	63.4	67.6
Pishchulin et al. [75]	87.5	78.1	75.7	68.0	54.2	33.9	58.0	62.9
Fang & Yi [66]	90.9	84.6	79.2	71.3	61.9	35.0	61.9	67.0
Chen & Yuille [110]	92.7	**87.8**	82.9	77.0	69.2	55.4	71.1	75.0
Fu et al. [153]	85.4	77.7	75.0	72.0	62.0	48.0	64.2	67.7
Ours	**93.5**	85.4	**84.9**	**78.6**	**72.2**	**60.9**	**74.2**	**77.3**

Table 5.1: Percentage of Correct Parts (PCP) comparison on the LSP dataset with Observer-Centric(OC) annotations.

especially achieving better estimation for arms. For lower arms we obtain 60.9 up from 55.4 for the next best performing method. The detection accuracies of the proposed model on all limbs are higher than other models. Compared with the state-of-the-art method in [110], the proposed method outperforms it in five out of six joints, and average limbs accuracy is 3.1% higher.

On the FLIC dataset, the PDJ metric is used to evaluate the performance of pose estimation. Fig. 5.13 and Fig. 5.14 present PDJ curves of elbows and wrists comparing against additional four methods. At normalized precision threshold $0.1, 0.15, 0.2$, the proposed model outperforms state of the art methods by a significant margin.

5.6.3.2 Example poses

The qualitative results of the proposed model on LSP datasets are illustrated in Fig. 5.15. The first row shows the detection for persons with different viewpoints. The second row presents the results of headstand objects. The last row is for more challenging poses. These results demonstrate the effectiveness and robustness of our model. Fig. 5.16 shows successful examples of human pose estimation on the FLIC dataset.

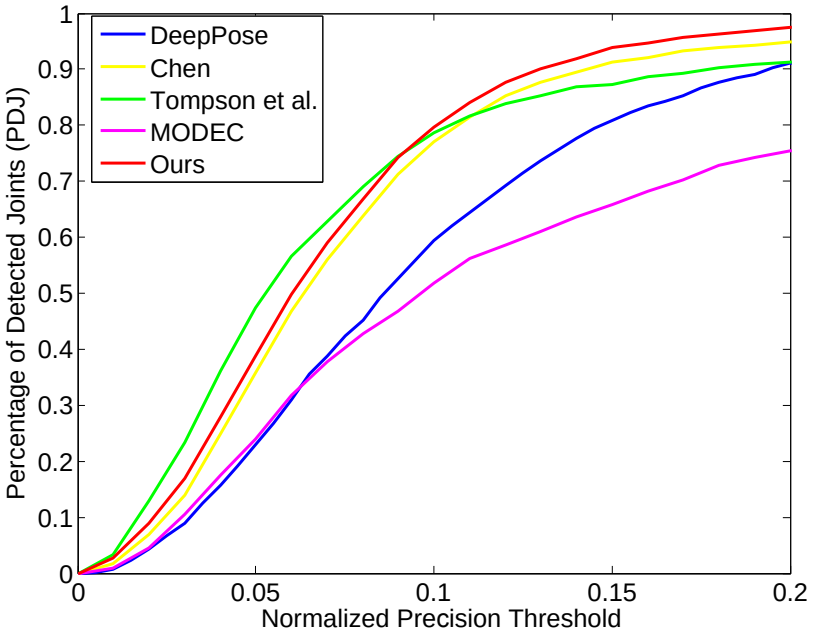


Figure 5.13: Comparison of the PDJ curves of elbows on the FLIC dataset.

.

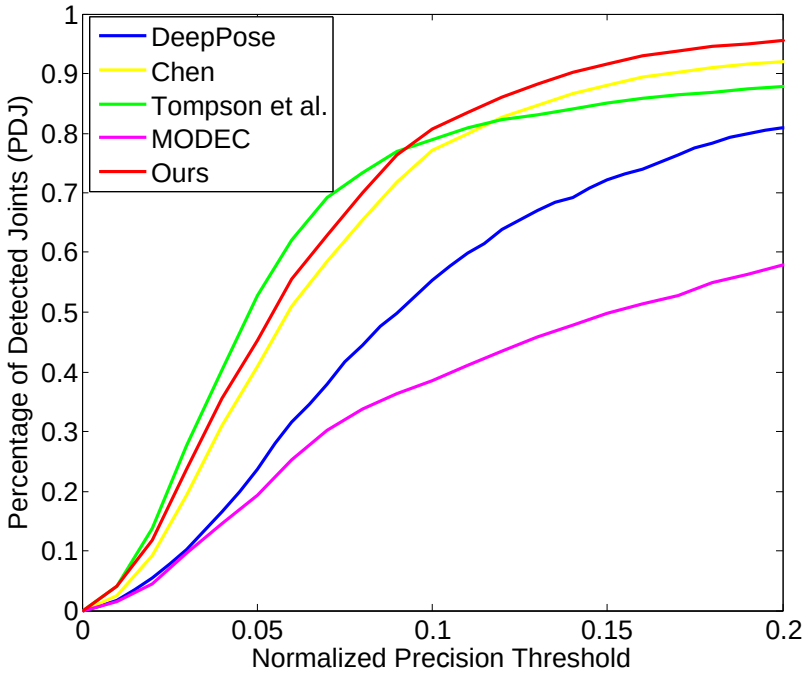


Figure 5.14: Comparison of the PDJ curves of wrists on the FLIC dataset.

.

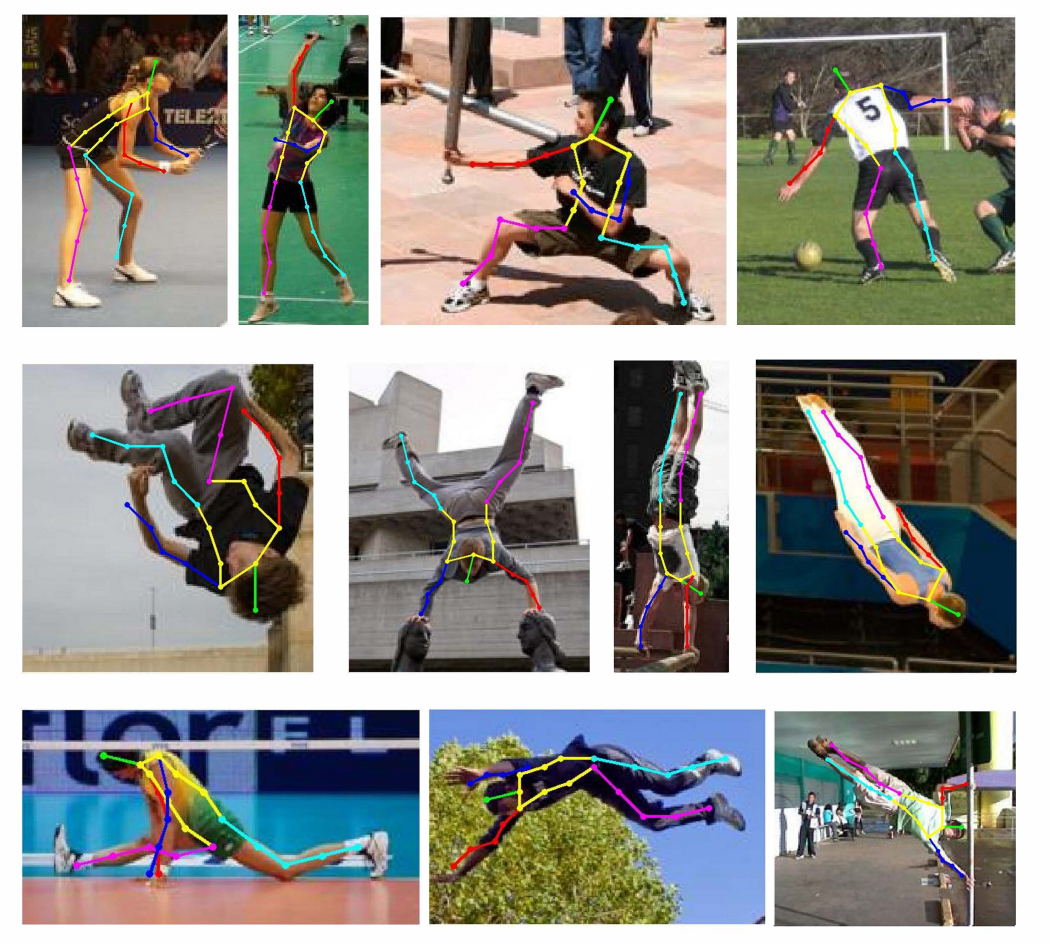


Figure 5.15: Results on the LSP dataset.

Figure 5.16: Results on the FLIC dataset.

5.7 Conclusion

This chapter presents human pose estimation with deep convolutional neural network. The local multi-resolution convolutional neural network (LMR-CNN) is proposed to learn the representation of each body part. The deep learning architecture is introduced in hierarchical model for pose estimation. Comparing to a traditional hierarchical model, the proposed model focuses on the limbs that are more complex in the structure and appearance. In this case, a Limbnet is defined based on LMR-CNN to learn the presence of limb parts. Empirical results suggest that the proposed model is more effective and outperforms the state of the art in human pose estimation.

Chapter 6

Conclusions and perspectives

Contents

6.1 Conclusions

Human pose estimation is a fundamental problem in computer vision. In this thesis, our contributions are summarized as follows. Firstly, the Flexible Mixtures of Parts (FMP) model is introduced to cooperate with Annealed Particle Filter (APF) for tracking body parts. The tracking and detection are combined to estimate and update the pose state. Secondly, an upper body based multiple mixture parts model (MMP) is proposed, which divides pose estimation into two stages: pre-estimation and estimation. In the pre-estimation stage, the main task is to find a more effective and discriminative model to categorize poses. In the estimation stage, the main task is to detect each body part using different part based models. Thirdly, we seek to promote the performance of pose estimation within deep learning framework. A Local Multi-Resolution Convolutional Neural Network (LMR-CNN) is proposed to learn the representation of each body part. Then the proposed LMR-CNN is introduced in a hierarchical model. Comparing to a traditional hierarchical model, the proposed model focuses on the limbs that are more complex in the structure and appearance. In this case, a Limbnet is defined to learn the presence of limb parts. Empirical results suggest that the proposed models are more effective and outperform the state of the art in human pose estimation.

6.2 Perspectives

This section discusses some potential future directions for pose estimation in this thesis.

Training data:

One of the main limitations for current human pose estimation research is the lack of large training datasets, such as ImageNet for object detection. This datasets would contain both single and multiple labelled objects in varying scenes (e.g. partk, schools etc.).

Temporal information: In videos, the temporal information can provide rich cues for estimating the upper-body or full-body poses. The optical flow encodes a type of temporal information between frames. In the CNN architecture, the temporal feature can used as input data that cooperate with image data to enhance the pose estimation.

Multiple objects: This thesis is focus on a single object. Nevertheless, many images or video sequences contains more than one object. This would be a challenge research in the future work.

Appendix A

Résumé de la thèse en Français

A.1 Introduction

L'estimation de la pose humaine est un problème fondamental dans la vision par ordinateur avec de nombreuses applications potentielles telles que le sport, la reconnaissance de l'action et de l'interaction homme-machine. L'estimation de la pose humaine est formulée comme suit: on se donne une image qui contient un corps humain et un modèle d'articulation (un modèle du corps) et on se propose de décrire la configuration du corps en termes d'un ensemble de membres et d'articulations de rotation qui les relient dans une structure arborescente (Fig. A.1). Au cours des dernières décennies, nous avons assisté à l'évolution des méthodes de l'estimation de la pose articulée d'une personne dans des environnements souvent intérieures contrôlées. Malgré de nombreuses années de recherche, l'estimation de la pose reste un problème difficile. Il partage toutes les difficultés de la détection d'objets, tels que la diversité des apparences, les changements de scène, de l'éclairage et la pose de la caméra, le changement de fond, et l'occlusion.

A.2 Estimation de la pose avec filtre à particules

A.2.1 Filtrage

A.2.1.1 Filtre à particules

L'algorithme de filtre à particules a été développé pour le suivi d'objets, en utilisant des estimateurs bayésiens récursifs se basant sur les techniques de Monte Carlo qui peuvent gérer les processus non gaussiens et multi-modaux. Afin de faire une estimation du paramètre, cet algorithme utilise l'échantillonnage d'importance. L'échantillonnage d'importance est une technique générale pour estimer les statistiques d'une variable aléatoire. L'estimation est basée sur des échantillons de cette variable aléatoire généré à partir d'une autre distribution, appelé la loi de proposition, qui est facile à échantillonner [120].

Couramment utilisé dans les problèmes de suivi, l'approche bayésienne vise à estimer la densité a posteriori $p(x_t|y_{1:t})$, où$y_{1:t}$ désigne l'historique des observations (x_t est un vecteur d'état caché et y_t est une mesure à l'instant t). Le processus d'observation est $p(y_t|x_t)$. La densité a posteriori est représenté par un ensemble de particules pondérés $\{(x_t^{(0)}, \pi_t^{(0)}) \cdots (x_t^{(N)}, \pi_t^{(N)})\}$, où $\pi_t^{(i)} \propto p(y_t|x_t^{(i)})$. La répartition de filtrage peut être calculée en utilisant deux étapes.

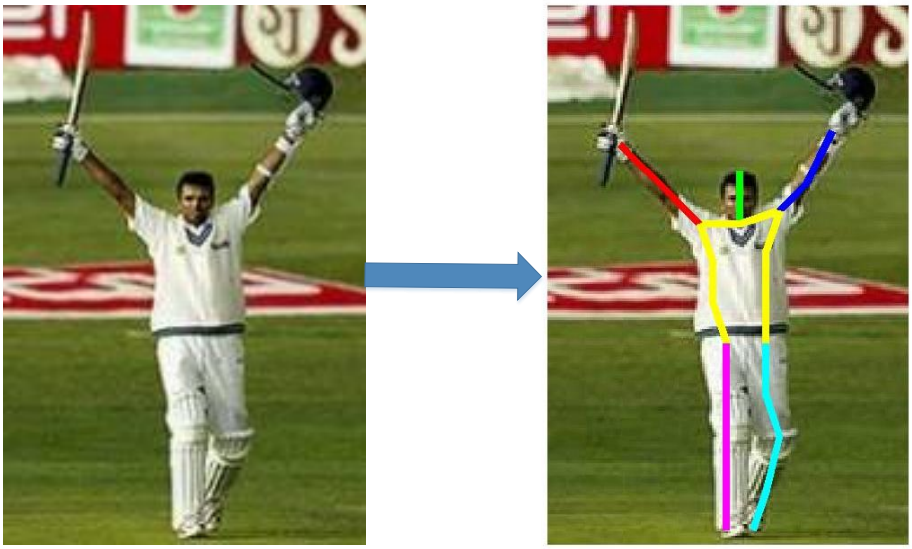


Figure A.1: Estimation de la pose.

Etape de prédiction :

$$p(x_t|y_{1:t-1}) = \int p(x_t|x_{t-1})p(x_{t-1}|y_{1:t-1})dx_{t-1}. \tag{A.1}$$

Etape de filtrage:

$$p(x_t|y_{1:t}) \propto p(y_t|x_t)p(x_t|y_{1:t-1}), \tag{A.2}$$

où $p(y_t|x_t)$ est la probabilité, et $p(x_t|y_{1:t-1})$ prédit l'état à l'instant t. Variations de PF: Importance échantillonnage séquentiel (SIS) attire des particules d'une distribution de proposition, puis pour chaque particule d'un poids approprié est affecté comme suit:

$$\pi_t^{(i)} \propto p(y_t|x_t^{(i)})p(x_t^{(i)}|x_{t-1}^{(i)})/q(x_t^{(i)}|x_{t-1}^{(i)}, y_t). \tag{A.3}$$

Par conséquent, un problème fondamental est que la distribution $p(y_n|x_n)$ peut être très piquée, parce que $p(y_n|x_n)$ détecte habituellement plusieurs maxima locaux au lieu de choisir celle qui est globale. Cela se produit généralement pour les problèmes de grande dimension, comme le suivi des parties du corps. Un autre facteur est le coût de calcul de calcul $p(y_n|x_n^{(i)})$. Souvent, une fonction de pondération intuitive $w_n^i(y_n, x)$ peut être construite, ce qui nécessite beaucoup moins d'effort de calcul pour évaluer [34]. Par conséquent, le problème revient à trouver la configuration x_k qui maximise la fonction de pondération $w_n^i(y_n, x)$, à déplacer vers le maximum global de la fonction de pondération.

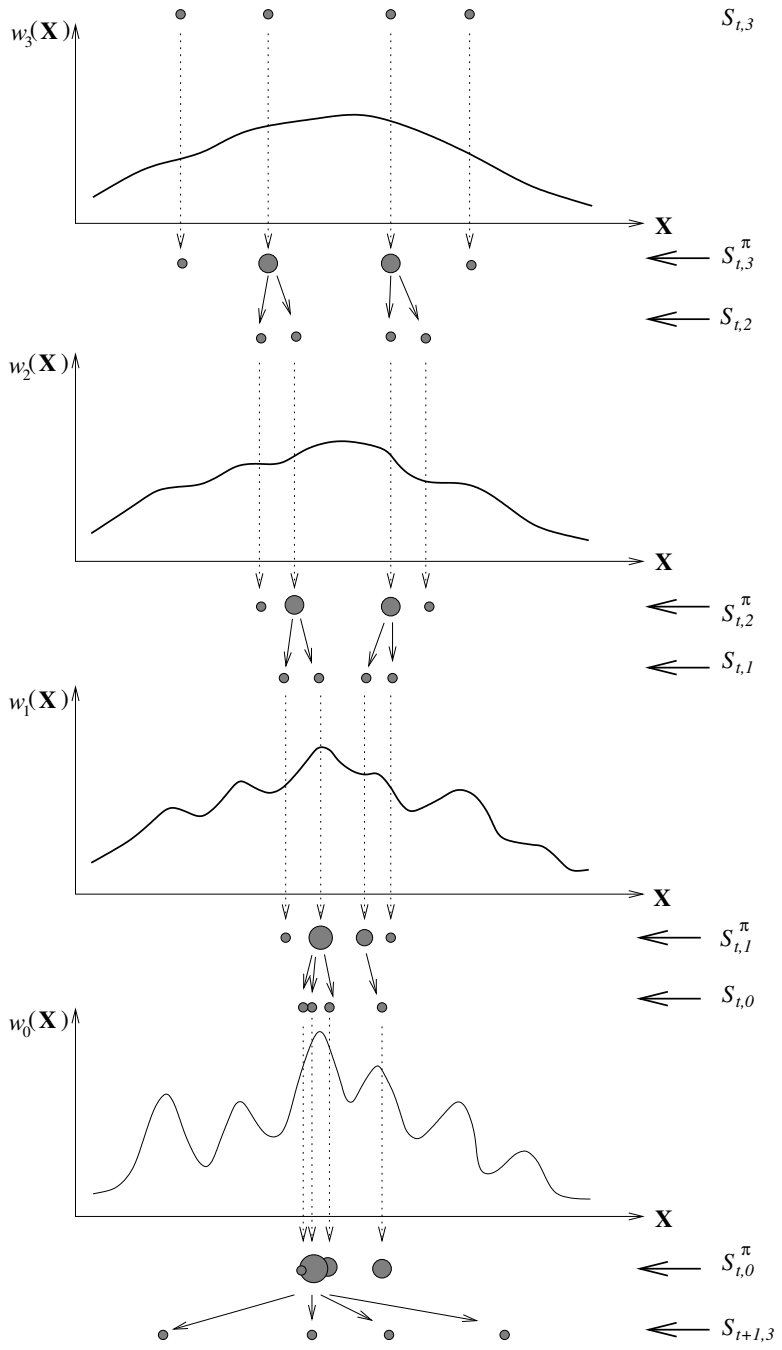


Figure A.2: Illustration du filtre à particules recuit avec $M = 3$. Un ensemble de particules creuses est lancé à la couche M et peu à peu vers le maximum global.

A.2.1.2 Filtre à particules en recuit simulé

Il a été démontré dans plusieurs ouvrages que l'échantillonnage SIR [121] sont une bonne approche pour le suivi dans les espaces de faible dimension, mais ils deviennent inefficaces dans les problèmes de grande dimension. Deutscher [34] a proposé une variante du cadre SIR en introduisant la notion de filtre recuit des particules. L'idée principale du CSA consiste à utiliser une série de fonctions de pondération ($w_0(y_t, x)$ à $w_M(y_t, x)$), où chaque$w_m(y_t, x)$ ne diffère que légèrement de $w_{m-1}(y_t, x)$. La fonction de pondération $w_M(y_t, x)$) est conçue pour être très lisse, représentant la tendance globale de l'espace de recherche. Ce résultat est obtenu en utilisant:

$$w_m(y_t, x) = (w_0(y_t, x))^{\beta_m}, \tag{A.4}$$

où $1 = \beta_0 > \cdots > \beta_M$ et $w_0(y_t, x)$ est égale à la fonction d'origine de pondération. Par conséquent, chaque opération de recuit comprend M couches. Comme illustré sur la Fig. A.2, un seul passage de recuit est effectuée à l'instant t. $\mathcal{S}^{\pi}_{t,m}$ désigne un ensemble de particules pondérés, tandis que $\mathcal{S}_{t,m}$ est un ensemble de particules non pondérés.

A.2.2 La modélisation du premier plan

A.2.2.1 Le modèle de base de la structure graphique

Le modèle de structure picturale [55] d'un objet est donné par un ensemble de pièces avec des connexions entre certaines paires de pièces. Plus précisément, pour le modèle de corps humain, les éléments peuvent correspondre à la tête, le torse, les bras et les jambes de l'être humain, comme le montre la Fig. A.3.

Les paramètres de la pose sont optimisés en maximisant la fonction de partition qui est définie comme suit,

$$S(I, L) = \Sigma_{i \in V} \alpha_i \cdot \phi(I, p_i) + \Sigma_{ij \in E} \beta_{ij} \cdot \psi(p_i, p_j), \tag{A.5}$$

où I désigne l'image, V est un ensemble de nœuds et p_i, p_j sont des positions des parties i et j, α_i est un modèle unaire pour une partie i, et $\phi(I, p_i)$ représente les caractéristiques locales de l'image à l'emplacement p_i dans l'image I; β_{ij} est le ressort par paire entre la partie i et la partie j, et $\psi(p_i, p_j) = [x_i - x_j, (x_i - x_j)^2, y_i - y_j, (y_i - y_j)^2]^T$ est la position relative entre la partie i et la partie j.

A.2.2.2 Mélanges flexibles de modèle de membres

Les mélanges flexibles de modèles de membres sont également basés sur le système PS. Ce modèle utilise des petites parties du corps, plutôt que le plus grand, ce qui est nettement plus rapide que le modèle original. Cette section décrit le modèle FMP. En prenant le mélange de parties en compte, la nouvelle fonction de score peut être définie comme:

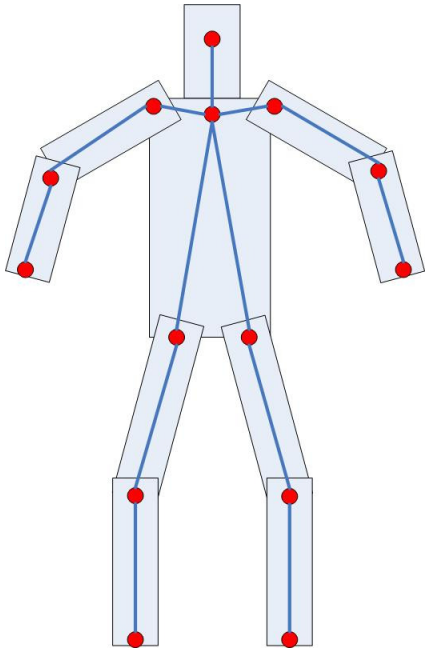


Figure A.3: Modèle du corps humain sur la base de la structure picturale: chaque nœud et le lien correspond à une partie et une connexion physique entre les parties.

$$\begin{aligned} S(I,L,M) =& \Sigma_{i\in V}\alpha_i^{m_i}\cdot\phi(I,p_i) \\ &+ \Sigma_{ij\in E}\beta_{ij}^{m_i m_j}\cdot\psi(p_i,p_j) + S(M), \end{aligned} \tag{A.6}$$

où m_i est le mélange de la partie i, $\alpha_i^{m_i}$ est modèle unaire pour la partie$i$ avec le mélange m_i, et $\beta_{ij}^{m_i m_j}$ est ressorts par paire entre la partie i avec le mélange m_i et la partie j avec le mélange m_j. $S(M) = \Sigma_{ij\in E} b_{ij}^{m_i,m_j}$ est une somme des scores par paires, deux à deux et le paramètre $b_{ij}^{m_i,m_j}$ favorise notamment les co-occurrences entre la partie $i$avec le mélange m_i et la partie j avec le mélange m_j. E est un ensemble de liens entre les parties.

A.2.3 Suivi avec FMP-APF

Basé uniquement sur le filtre à particules, on ne peut pas suivre efficacement les mouvements apparents rapides. D'autre part, le modèle FMP ne peut pas trouver certaines parties du corps en raison du chevauchement et de l'occlusion. Pour ces raisons, nous combinons ces deux méthodes ensemble, et proposons une approche basée sur l'apprentissage de premier plan.

A.2.3.1 Détection par FMP en multi-vue

Comme indiqué au Section 3, FMP ne parvient pas à détecter les parties du corps, en raison du chevauchement et de l'occlusion. Fusionner plusieurs vues permet de

résoudre ces problèmes en combinant la détection dans chaque vue. Cette thèse étend le FMP au cas multi-vues:

$$S(I,P,M,K) = \Sigma_{i\in V}\alpha_i^{m_i}\cdot\phi(I_k,p_{i,k}) + \Sigma_{ij\in E}\beta_{ij}^{m_i m_j}\cdot\psi(p_{i,k},p_{j,k}) + S(M), \tag{A.7}$$

où I_k désigne l'image I en vue k, $p_{i,k}$ est l'emplacement de la partie i en vue k, et $S(M)$ est une somme des scores par paires. Soit (n,m) désigne une paire de points de vue différents de vues K. Donc $p_{i,n}$ et $p_{i,m}$ sont les endroits de la partie i en vue n,m, qui sont calculées par eq.8. Cependant, parfois, la position p_i est pas assez précis. contrainte épipolaire est utilisé entre deux vues pour éliminer les erreurs de mesure et de permettre une localisation plus précise. La matrice fondamentale F est la représentation de la géométrie épipolaire et la contrainte épipolaire est représenté par $p_{i,n}^T F p_{i,m} = 0$. Si les points $p_{i,n}$ et $p_{i,m}$ sont cohérentes, $p_{i,n}$ se trouve sur la ligne épipolaire $l = Fp_{i,m}$. Dans ce cas, la 3D la position q_i de la partie i peut être calculée par l'arrière-projection de $p_{i,n}$ et $p_{i,m}$ comme suit,

$$\begin{cases} L_{i,n}(\lambda) = P^+ p_{i,n} + \lambda C, \\ L_{i,m}(\lambda) = P^+ p_{i,m} + \lambda C, \end{cases} \tag{A.8}$$

où $L_{i,n}, L_{i,m}$ sont deux rayons, P^+ est la pseudo-inverse de la matrice de la caméra P, et C est le centre de la caméra. L'intersection des deux rayons $L_{i,n}, L_{i,m}$ est la position 3D q_i. De tout (n,m) possible des vues K, au moins une paire est cohérente, alors la position 3D est conservée et nous considérons la prochaine partie du corps. Sinon, une mise à jour de la position 3D précédente est réalisée par APF comme détaillé dans le prochain paragraphe.

A.2.3.2 Mise à jour de l'état avec l'APF

Comme indiqué plus haut, certaines parties du corps n'ont pas des vues multiples. Pour résoudre ce problème, nous introduisons APF dans le cadre du FMP pour réaliser le suivi robuste pour toutes les parties du corps. La configuration optimale a été calculé à partir de la particule fixée à la couche de fond à l'aide de:

$$x_{t-1} = \Sigma_{j=1}^{N_p}\pi_{t-1,0}^{(j)} x_{t-1,0}^{(j)}, \tag{A.9}$$

où N_p est le nombre de particules. Soit $x_{t-1} = (X_{t-1,1}, X_{t-1,2}\cdots X_{t-1,S})$, $X_{t-1,i}$ est le vecteur de paramètres pour la partie i à l'instant $t-1$, et S est le nombre de parties du corps. Comme indiqué au Section 2, après que l'échantillon soit tiré, l'estimation d'état pour chaque particule devient:

$$p(x_t|x_{t-1},y_t) \propto p(y_t|x_t)p(x_t|x_{t-1}). \tag{A.10}$$

APF est pas approprié pour l'estimation des paramètres d'état de grande dimension, en particulier pour les paramètres d'état de mouvement rapide des parties du corps

(bras et jambes). L'idée principale de cette thèse est d'utiliser la détection de parties du corps pour en déduire un sous-ensemble des paramètres d'état. Supposons que le vecteur d'état x_t peuvent être décomposées en $(x_{t,1}, x_{t,2})$, où $x_{t,1}$ doit être calculée par l'APF, tandis que les paramètres de l'Etat $x_{t,2}$ ont déjà été calculée par le multi-vue FMP. Par conséquent, l'estimation d'état pour chaque particule peut être réécrite comme:

$$p(x_{t,1}|x_{t-1,1},x_{t,2}, y_t) \propto \\ p(y_t|x_{t,1}, x_{t,2})p(x_{t,1}|x_{t-1,1}, x_{t,2}), \tag{A.11}$$

l'expression ci-dessus combine le suivi et la détection automatique pour effectuer la récupération des parties du corps. Représentée par le terme $p(x_{t,1}|x_{t-1,1}, x_{t,2})$, qui est utilisé pour estimer l'état $x_{t,1}$ basé sur le paramètre $x_{t-1,1}$ et $x_{t,2}$. Après toutes les particules sont calculés à la couche inférieure de la façon suivante:

$$x_{t,1} = \Sigma_{j=1}^{N_p} \pi_{t,1,0}^{(j)} x_{t,1,0}^{(j)}, \tag{A.12}$$

de sorte que le nouveau vecteur d'état x_t est également calculé.

A.3 Estimation de la pose avec pré-traitement de la partie supérieure

A.3.1 Proposé méthode d'estimation

La méthode proposée est composée de deux phases: l'une est la détection du haut du corps et sa classification, l'autre est l'estimation de pose humaine. Ces deux étapes sont décrites dans le paragraphe suivant.

A.3.1.1 Détection du haut du corps et catégorisation

Cette section donne une brève introduction à la détection du haut du corps et présente le modèle proposé hiérarchique du haut du corps, et l'approche proposée pour estimer les catégories de la partie supérieure du corps.

Modèle hiérarchique du haut du corps Notre modèle de partie supérieure du corps est basé sur les approches précédentes pour la détection d'objet rigide [61, 62, 80]. Tous ces détecteurs utilisent un mécanisme de fenêtre glissante suivie d'une suppression non maximale. Un détecteur de corps supérieur a été proposé dans la Ref. 80, qui a combiné des modèles de HOG avec un détecteur de visage [136]. Ce modèle se comporte bien sur des images vidéo à partir de films et émissions de télévision. Cependant, les résultats sont médiocres lorsque cette méthode est utilisée dans certains jeux de données plus difficiles(e.g. Leeds Sports Pose dataset (LSP) [127]). Ainsi, un modèle hiérarchique du haut du corps est proposé (Fig. A.4), qui est également basé sur le visage et la détection du haut du corps. La principale contribution de notre travail est d'ajouter le terme par paires entre les

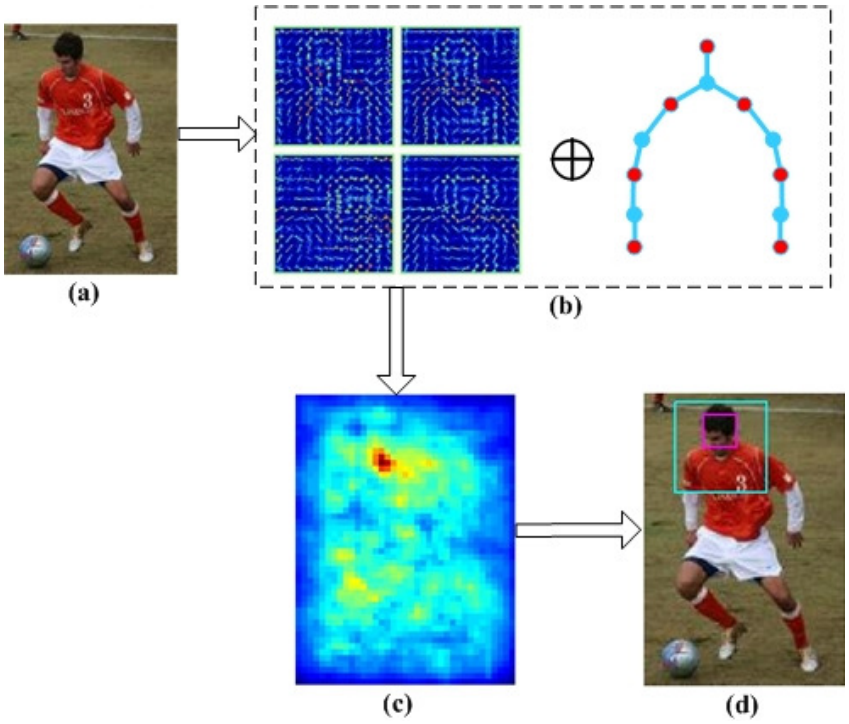


Figure A.4: Modèle du haut du corps. (a) L'image d'entrée. (b) Le modèle hiérarchique du haut du corps: le premier modèle est le modèle du haut du corps au niveau mondial, tandis que le second est le modèle d'arbre sur la base de la partie (nœuds rouges désignent différentes articulations, et la tête est le nœud racine). (c) Le score cartes combinées de filtres du haut du corps et les filtres de la pièce. (d) Le résultat de la détection du haut du corps: la boîte de cyan indique l'emplacement du haut du corps.

parties dans le haut du corps au lieu de la détection de visage dans la Ref. 136. Il existe des modèles de pièces à deux niveaux (Fig. A.4(b)): Le modèle du haut du corps au niveau mondial et le niveau local based model partie. Pour chaque modèle de niveau, il y a plusieurs composants du mélange similaires à FMP. Pour le modèle de niveau local, il est défini par 12 petites pièces dont 4 pièces pour chaque bras, 1 pour chaque épaule et 2 pour la tête, qui sont utilisés pour calculer le terme par paires. Ensuite, la fonction de score peut être définie comme:

$$\begin{aligned} S(I,P,M) =& \omega^m_{upper} \cdot \phi(I, p_{upper}) + \omega^m_{head} \cdot \phi(I, p_{head}) \\ &+ \sum_{i,j} \omega^{m_i m_j}_{ij} \cdot \psi(p_i, p_j) + S(M), \end{aligned} \tag{A.13}$$

qui peut être divisé en trois termes différents: terme aspect, terme de déformation et terme de de compatibilité.

Terme aspect:. les deux premiers termes de l'Eq.(2) sont un modèle d'apparence qui comprend deux niveaux scores locaux: le haut du corps et la tête. ω^m_{upper} est le modèle de HOG pour le haut du corps avec le type de mélange m, tandis ω^m_{head} est le modèle de la tête.

Terme de déformation: $\Sigma_{i,j}\omega^{m_i m_j}_{ij} \cdot \psi(p_i, p_j)$ est le terme déformable, où i, j désigner différentes parties dans le haut du corps. Il est aussi décrit comme terme par paire qui peut être interprété comme la fixation d'un ressort entre les deux parties. Ce terme peut être calculée par la distance à partir de la transformation de nœud feuille au nœud racine.

Terme de compatibilité: le dernier terme $S(M)$ indique si deux types sont compatibles dans l'ensemble de la formation. Avec le terme déformable, il spécifie une image-indépendant avant sur une partie des emplacements et des types. Ainsi, l'emplacement du haut du corps est obtenu en maximisant la fonction de score suivant:

$$p^*_{upper} = \arg\max_p S(I, P, M), \tag{A.14}$$

où p^*_{upper} est l'emplacement du haut du corps de la détection.

Estimation des catégories du haut du corps. Après la détection de la partie supérieure du corps, il est proposé d'estimer les catégories de la partie supérieure du corps. Le but de cette étape est de classer une série d'images d'entrée en différentes catégories en fonction de la partie supérieure du corps. Comme illustré sur la Fig. 1, trois différentes catégories du haut du corps sont définis: vue de gauche à droite côté, près de vue avant-arrière et vue ATR. Cette étape pourrait être étendu à d'autres catégories ou d'autres plus discriminantes caractéristiques de la partie du corps.

Examinons les stratégies pour estimer catégories du haut du corps. Les résultats expérimentaux montrent que le procédé suivant est préférable. Différents ensembles de modèles sont proposés pour détecter la partie supérieure du corps et évaluer les catégories de la partie supérieure du corps en une seule étape. Dans ce cas, nous réécrivons l'Eq. (2) associé à la configuration de catégories du haut du corps:

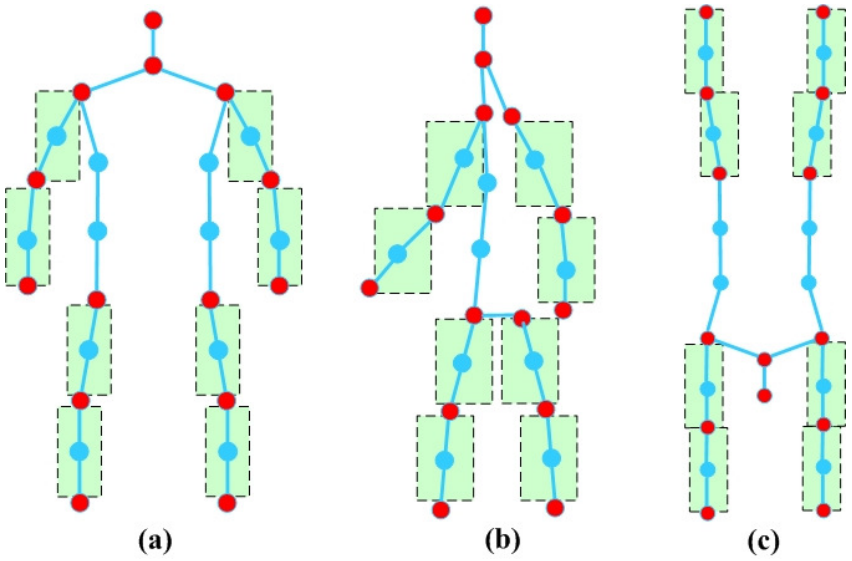


Figure A.5: Multiple modèle de pièces de mélange. Notre modèle de MMP est composé d'un modèle en trois catégories de composants du mélange. Les nœuds rouges indiquent les articulations, et les nœuds cyan désignent des points intermédiaires entre deux articulations. Les boîtes vertes dénotent combinés modèle MMP. (a) Le modèle de pièces de mélange pour une vue avant-arrière près de poses est composé de 26 parties. (b) Le modèle de pièces de mélange pour afficher latérales droite-gauche poses comprend 24 parties. (c) Le modèle de pièces de mélange pour afficher handstand poses a 26 parties.

$$\begin{aligned} S(I,P,M,C) =& \omega_{upper}^{m,c} \cdot \phi(I, p_{upper}) + \omega_{head}^{m,c} \cdot \phi(I, p_{head}) \\ &+ \sum_{i,j} \omega_{ij}^{m_i m_j, c} \cdot \psi(p_i, p_j) + S(M, C), \end{aligned} \tag{A.15}$$

où c désigne l'indice de la catégorie de la partie supérieure du corps. En calculant les notes maximales dans chaque modèle du haut du corps, on peut déterminer la catégorie c, qui est également utilisé pour sélectionner le modèle de MMP. Immédiatement, c est défini dans l'équation ci-dessous:

$$c_1^* = \arg\max_c S(I, P, M, C), \tag{A.16}$$

où c_1^* est l'indice de la catégorie choisie.

A.3.1.2 Modèle de mélange multiple(Etape de l'estimation)

Dans ce paragraphe, nous décrivons d'abord le modèle multiple composée de différents modèles distincts (voir Fig. A.5). Chaque modèle correspond à une catégorie dans les données de formation. Le modèle de MMP en deux temps proposée présente

plusieurs avantages. Dans cette thèse, trois catégories sont proposées dans le modèle de RPM. La première catégorie est pour prèsenter des poses de vue avant-arrière (voir Fig. A.5(a)). Ce modèle dispose de 14 articulations et 26 parties. La deuxième est de droite-gauche poses vue de côté (voir Fig. A.5(b)). Il convient de noter que le modèle proposé a 24 parties au lieu de 26 parties. Dans le modèle 26-parties, il y a 6 parties pour le torse qui sont capables de chevaucher en vue de côté coulisses. Néanmoins, le modèle 24-partie proposée peut ne pas correspondre seulement le corps des contraintes cinématiques humaines, mais aussi de réduire le double comptage. La troisième catégorie est pour les poses ATR de style (la tête en bas, les pieds). Il convient de noter que la partie supérieure du corps de cette catégorie est plus discriminante, et rend la pose facile de catégorisation. Pour ce modèle, nous adoptons toujours un modèle 26-partie. Dans chaque catégorie, il y a 8 boîtes vertes qui dénotent 8 modèles combinés. Étant donné que les membres sont plus complexes et les apparences changent de manière significative, en ajoutant 8 modèles combinés de MMP est proposé de donner plus de contexte. Les modèles combinés partagent les joints avec des modèles de pièces locales. Par conséquent, les modèles combinés sont définis pour connecter deux articulations des membres, et chaque articulation d'un membre a deux niveau modèles (un modèle combiné et un modèle de partie locale). La taille du modèle combiné est supérieur à celui du modèle de pièce locale.

D'une manière similaire au modèle du haut du corps, nous définissons le modèle de MMP comprenant trois termes: l'apparence, la déformation et la compatibilité. Il convient de noter que le modèle combiné est à chaque fois utilisé.

Terme apparence: Ce terme comprend deux niveaux de modèles. L'un est pour les modèles combinés et l'autre est pour les modèles de pièces locales. Le score de l'apparence peut être écrite comme:

$$S_a(I,P,C) = \sum_{i \in V} \omega_i^{c,m_i} \cdot \phi(I,p_i) + \lambda \sum_{k \in V_l} \omega_k^{c,m_k} \cdot \phi(I,p_k), \tag{A.17}$$

où c désigne l'indice de catégorie de MMP, V est un ensemble de pièces locales et V_l est un ensemble de pièces combinés. p_i est la position de la partie locale i, tandis que p_k est la position du combiné partie k. ω_k^{c,m_k} désigne un modèle HOG pour partie combinée k avec la catégorie index c et le type de mélange m_k, tandis que ω_i^{c,m_i} est pour le niveau de la partie locale. Ces modèles d'apparence à deux niveaux sont combinés par le paramètre λ, qui contrôle le compromis entre deux termes et est tourné manuellement.

Terme déformation:

$$S_d(I,P,C) = \sum_{i,j \in E} \omega_{i,j}^{c,m_i m_j} \cdot \psi(p_i,p_j) + \lambda \sum_{f,g \in E_l} \omega_{f,g}^{c,m_f m_g} \cdot \psi(p_f,p_g), \tag{A.18}$$

où la partie i,j et la partie f,g sont deux à deux reliés en E, E_l. E est un ensemble de liens entre les deux parties dans le modèle de partie locale et E_l est pour le modèle combiné.

Terme de compatibilité:

$$\begin{aligned} S(M,C) =& \Sigma_{i,j\in E} b_{i,j}^{c,m_i,m_j} + \lambda \Sigma_{f,g\in E_l} b_{f,g}^{c,m_f,m_g} \\ &+ \Sigma_{i\in V} b_i^{c,m_i} + \lambda \Sigma_{k\in V_l} b_k^{c,m_k}, \end{aligned} \tag{A.19}$$

semblable au modèle du haut du corps, $S(M,C)$ est défini comme étant un modèle de co-occurrence. Le paramètre b_{ij}^{c,m_i,m_j} favorise notamment co-occurrences entre une partie i avec un mélange m_i et une partie j avec un mélange m_j dans la catégorie c. Par exemple, si les types de pièces correspondent aux orientations et la partie i et j sont sur le même membre rigide, b_{ij}^{c,m_i,m_j} serais en faveur de missions d'orientation cohérents. Le parent j recueille les messages de ses tous les enfants et transmet les messages à son parent de manière récursive vers un nœud racine.

L'inférence correspond à maximiser la fonction complète de partition:

$$p^* = \arg\max_p S_a(I,P,C) + S_d(I,P,C) + S(M,C), \tag{A.20}$$

où p^* désigne l'emplacement des parties du corps. Les scores sont utilisés pour générer la détection multiple dans l'image I les achète un seuil et en appliquant la suppression non-maximale (NMS). Un retour en arrière est utilisé pour trouver l'emplacement et le type de chaque partie dans chaque configuration maximale.

Apprentissage Pendant l'apprentissage, nous avons accès à des images annotées avec la réalité de terrain conjointe / endroits du corps et de numéros de catégorie $\{I_t, P_t, M_t, C_t\}_{t=1}^T$. T est le nombre d'images d'apprentissage. I_t désigne l'image avec la valeur de vérité terrain P_t, l'indice de mélange M_t et le manuel category C_t. Il convient de noter que P_t est pas traité comme variable cachée. Pour illustrer l'apprentissage, écrivons $Z_t = (P_t, M_t, C_t)$. La fonction de partition peut être exprimée en termes de produit scalaire, $S(I_t, Z_t) = \beta \cdot \Phi(I_t, Z_t)$, entre un vecteur de paramètres du modèle β et un vecteur de caractéristiques $\Phi(I_t, Z_t)$,

$$\begin{aligned} \beta =& (\omega_1^{c,m_1}, \cdots, \omega_N^{c,m_N}, \omega_{N+1}^{c,m_{N+1}}, \cdots, \omega_{N+K}^{c,m_{N+K}}, b_1^{c,m_1}, \cdots, b_N^{c,m_N}, \\ & b_{N+1}^{c,m_{N+1}}, \cdots, b_{N+K}^{c,m_{N+K}}, \cdots, \omega_{i,j}^{c,m_i,m_j}, \cdots, \cdots, \omega_{f,g}^{c,m_f,m_g}, \cdots, \\ & \cdots, b_{i,j}^{c,m_i,m_j}, \cdots, \cdots, b_{f,g}^{c,m_f,m_g}, \cdots), \quad (i,j \in E; f,g \in E_l) \end{aligned} \tag{A.21}$$

$$\Phi(I_t, Z_t) = (\phi(I,p_1), \cdots, \phi(I,p_N), \lambda\phi(I,p_{N+1}), \cdots, \lambda\phi(I,p_{N+K}), 1, \cdots, 1, \lambda, \cdots, \lambda, \tag{A.22}$$

$$\cdots, \psi(p_i,p_j), \cdots, \cdots, \lambda\psi(p_f,p_g), \cdots, \cdots, 1, \cdots, \cdots, \lambda, \cdots), \quad (i,j \in E; f,g \in E_l)$$

où N est le nombre de modèles de pièces locales et K est le nombre de modèles combinés. Ainsi, nous définissons une grande marge apprentissage fonction objectif semblable au travail de Ref. 63, 138:

$$\begin{aligned} &\arg\min_{\beta,\xi_t\geq 0} \frac{1}{2}\|\beta\|^2 + \mathcal{C}\sum_t \xi_t, \\ s.t. &\forall t \in pos \quad \langle \beta, \Phi(I_t, Z_t)\rangle \geq 1 - \xi_t, \\ &\forall t \in neg \quad \langle \beta, \Phi(I_t, Z)\rangle \leq -1 + \xi_t, \end{aligned} \tag{A.23}$$

Figure A.6: Cadre pour l'estimation des poses humaines.

où $\mathcal{C}$ contrôle le poids relatif du terme de régularisation et ξ_t représente les variables d'écart de la fonction objectif. La contrainte indique que des exemples positifs devraient marquer mieux que 1, tandis que des exemples négatifs de moins de -1. Il est à noter que la norme structuré SVM ne nécessitent pas un ensemble de formation négative explicite, et au lieu de générer des données négatives de la part des exemples positifs avec MIS-estimés étiquettes Z. Le modèle formé avec les contraintes ci-dessus est plus robuste et fonctionne bien pour l'estimation de la pose.

A.4 Estimation de la pose avec un réseau profond de neurones à convolution

A.4.1 Modèle

L'humain articulé estimation de pose peut être divisé en parties principales d'arbres, y compris la structure du modèle, extraction de caractéristiques, et le paramètre d'apprentissage (voir la Fig. A.6). Nous allons d'abord introduire le modèle et l'apprentissage procédure graphique proposée de notre modèle.

A.4.1.1 Modèle graphique

Modèle d'apparence. Le modèle proposé humaine se fonde également le modèle de structure picturale qui a montrer ses performances dans de nombreux ouvrages. Ici, nous simplifions les multiples modèles de pièces de mélange (MMP) à un seul modèle de pièce de mélange à l'aide de plusieurs filtres de chaque partie du corps. Comme on le voit sur la Fig. A.7, il contient 18 parties communes et 8 parties de branche. Pièces de membre contiennent deux bras, deux bras inférieurs, deux jambes supérieures et deux jambes. Taille différente des parties du corps sont fusionnés ensemble pour représenter efficacement pose humaine. En général, les parties de branche sont plus grandes que les parties d'articulation, et obtenir plus d'informations de contexte. Ainsi, on peut réécrire le terme d'apparence dans le modèle MMP comme:

$$S_a(I,P) = \sum_{i^* \in V^*} \omega_{i^*}^{m_{i^*}} \cdot \phi(I, p_{i^*}) + \sum_{k^* \in V_l^*} \omega_{k^*}^{m_{k^*}} \cdot \phi(I, p_{k^*}), \tag{A.24}$$

où $V^* = \{1, \ldots, N\}$ est un ensemble de pièces communes et $V_l^* = \{1, \ldots, K\}$ est un ensemble de pièces de membres. Il convient de noter que V^* et V_l^* est sous-ensemble

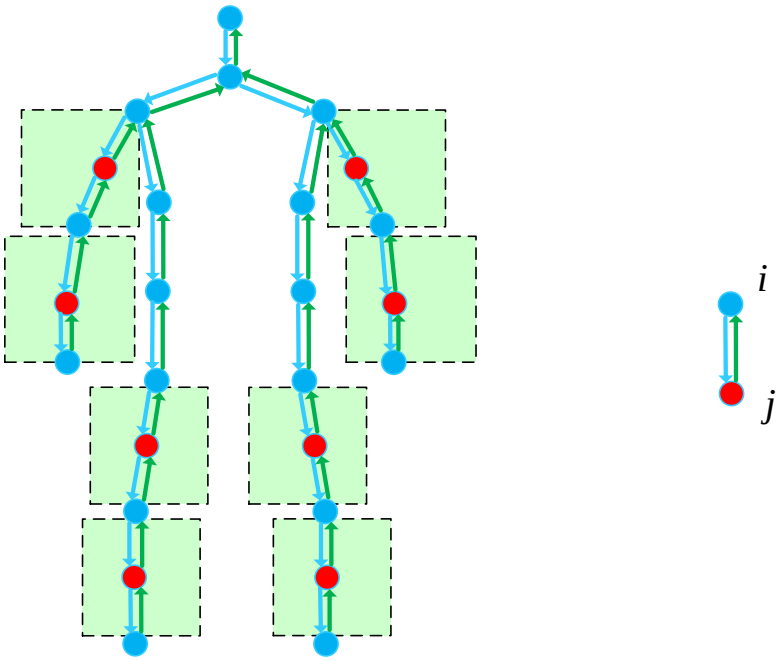


Figure A.7: Modèle graphique pour estimer poses humaines. (a) est le modèle graphique: les noeuds rouges désignent les centres de pièces de membre, et les noeuds cyan désignent des centres de pièces communes. Les cases vertes indiquent les grandes régions des parties de branche. Chaque pièces de paire connectée disposent de l'information déformable relative. (b) est une relation connectée paire entre la partie i et une partie j. Le bord cyan avec une flèche indique le rapport mélange m_{ij}, tandis que le vert représentent le mélange par rapport m_{ji}.

.

de V, où $V = V^* \cup V_l^* = \{1, \ldots, K+N\}$. $K+N$ est le nombre total de parties du corps humain dans le modèle proposé. p_{i^*} est la position de la partie commune i^*, tandis que p_{k^*} est la position du membre partie k^*. $\omega_{k^*}^{m_{k^*}}$ désigne la fonction de membre partie k^* avec le type de mélange m_{k^*}, alors $\omega_i^{m_i}$ est pour le niveau de la pièce commune.

Modèle déformable. Le modèle déformable est utilisé pour prédire les positions spatiales relatives entre chaque paire des parties du corps dans le modèle graphique. Ces paires de pièces mélangées parties communes et parties de branche. Par exemple, les parties voisines d'un genou gauche sont laissés cuisse et de la jambe et la cuisse gauche est parent partie du genou gauche tandis que le bas de la jambe fait partie des enfants. Afin de rendre l'utilisation de l'information de mélange de chaque paire de pièces, les emplacements par paire entre une partie $i \in V$ et une partie $j \in V$ sont discrétisé en mélanges relatifs $m_{ij} \in \{1, \ldots, M_{ij}\}$, où $V = \{1, \ldots, K+N\}$ est un ensemble de toutes les parties communes et parties

de branche. Pour chacun des mélanges relatifs correspondent à une position relative moyenne $r_{ij}^{m_{ij}}$ qui est calculée à partir des toutes les données de formation. Ainsi, le terme déformable peut être réécrit comme suit:

$$S_d(I,P) = \sum_{i,j\in E} \omega_{ij}^{m_{ij}} \cdot \psi(p_i + r_{ij}^{m_{ij}} - p_j) + \sum_{i,j\in E} \omega_{ij}\phi(I,p_i,m_{ij}) + \sum_{j,i\in E} \omega_{ji}^{m_{ji}} \cdot \psi(p_j + r_{ji}^{m_{ji}} - p_i) + \sum_{j,i\in E} \omega_{ji}\phi(I,p_j,m_{ji}), \quad \text{(A.25)}$$

où la partie i,j sont deux à deux reliés en E. E est un ensemble de liens entre les deux de 26 parties (8 parties de branche et 18 joints parties). $\omega_{ij}^{m_{ij}}, \omega_{ij}, \omega_{ji}^{m_{ji}}, \omega_{ji}$ sont les paramètres de poids. $\phi(I,p_i,m_{ij})$ désigne la fonction pour la partie i avec le mélange relative (type) m_{ij}, tandis $\phi(I,p_j,m_{ji})$ est pour la partie j avec le mélange m_{ji}. $\psi(p_i + r_{ij}^{m_{ij}} - p_j)$ est la fonction de déformation quadratique standard, où la position relative de la partie i par rapport à une partie j. Ainsi, $\psi(p_j + r_{ji}^{m_{ji}} - p_i)$ est la fonction de la déformation de la position relative d'une partie j relativement à la partie i.

A.4.2 Modèle hiérarchique profond basé sur un réseau de neurones à convolution multi-résolution

A.4.2.1 Réseau de neurones à convolution

Le réseau de neurones à convolution a montré ses performances dans de nombreux domaines. Un DCNN se compose principalement de deux parties: couches de convolution, et des couches entièrement connectées. Les couches de convolution fonctionnent comme des fenêtres coulissantes et en sortie comprennent des cartes qui représentent la disposition spatiale des activations. En fait, les couches de convolution ne possèdent pas la limitation d'une taille d'image fixe et peuvent générer des cartes de toutes tailles. D'autre part, les couches entièrement connectées ont besoin d'avoir des entrées de taille fixe de par leur définition. Ainsi, les blocs d'image d'entrée doivent rester à la même taille. Ceci conduit au problème de la détermination des tailles de raccordement d'entrée de chaque partie du corps lors de l'expression de la formation. Il existe une méthode commune pour résoudre ce problème en redimensionnant tous les correctifs de la même partie du corps à partir de toutes les données de formation. Cette méthode ne tient pas compte du fait que les tailles d'image de patch d'une seule partie du corps sont différents dans chaque exemple de formation. D'autre part, la taille du patch unique ne peut pas fournir suffisamment d'informations de contexte. Motivé par ceux-ci, un modèle multi-résolution de convolution (LMR-CNN) est proposé. Comme représenté sur la Fig. 5.7 (a), il contient des échelles à trois niveaux de l'articulation du coude $((e+o)*(e+o), e*e, (e-o)*(e-o))$. o est défini comme étant le décalage dans ce échelles multiples. e est l'échelle de centre des parties du raccord d'entrée dans LMR-CNN. Tout d'abord, il faut rogner ces correctifs à trois niveaux de l'image donnée, et redimensionner chaque patch dans la même taille $e*e$. Ensuite, ces correctifs sont enchaînés ensemble. Puisqu'une image RGB dispose de 3 canaux, la

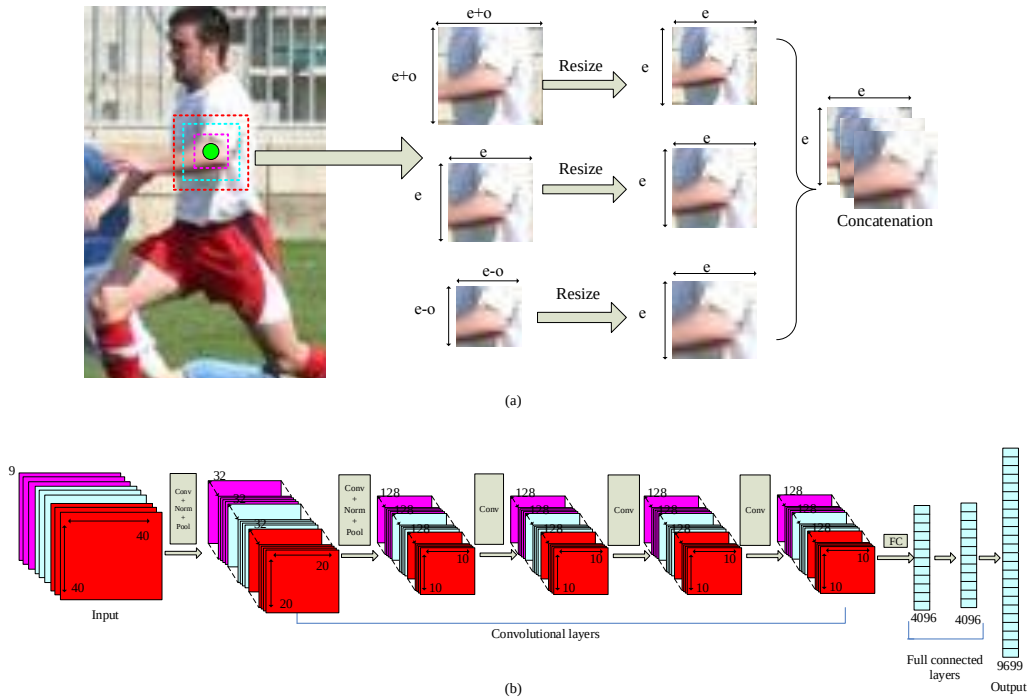


Figure A.8: Contexte riche réseau de neurones à convolution. (a) montre l'enchaînement des différentes échelles d'une partie de garçon pour augmenter le canal de données d'entrée. (b) les concaténations de chacune des parties du corps sont utilisées comme données d'entrée et les couches traitées par convolution et des couches connectées à part entière.

concaténation de trois patches possède 9 canaux. Il est à noter que ces échelles à trois niveaux peuvent être étendues à plusieurs niveaux. Fig. A.8 (b) présente la canalisation de la proposition de LMR-CNN qui prend la concaténation de patchs en tant que données d'entrée. Il y a 5 couches de convolution à sortie caractéristiques à trois niveaux qui a le niveau du contexte différent. L'entrée de la couche entièrement connecté est toutes les fonctionnalités de convolution. Ainsi, le projet de LMR-CNN fournit suffisamment d'informations de contexte, et permet également de capturer la fonction locale.

A.4.2.2 modèle de branche hiérarchique profond pour l'estimation de la pose

En raison de la petite taille des pièces communes, le modèle n'est pas sémantiquement significatif et ne peut pas capturer les différentes granularités de détails. Les modè des poselet présentent la limitation de la construction de détecteurs de membres non-rigides car ils sont variables en apparence. Un modèle hiérarchique peut être considéré comme le pont entre ces deux approches populaires pour l'estimation de pose. En comparant à un modèle hiérarchique traditionnel, le modèle proposé

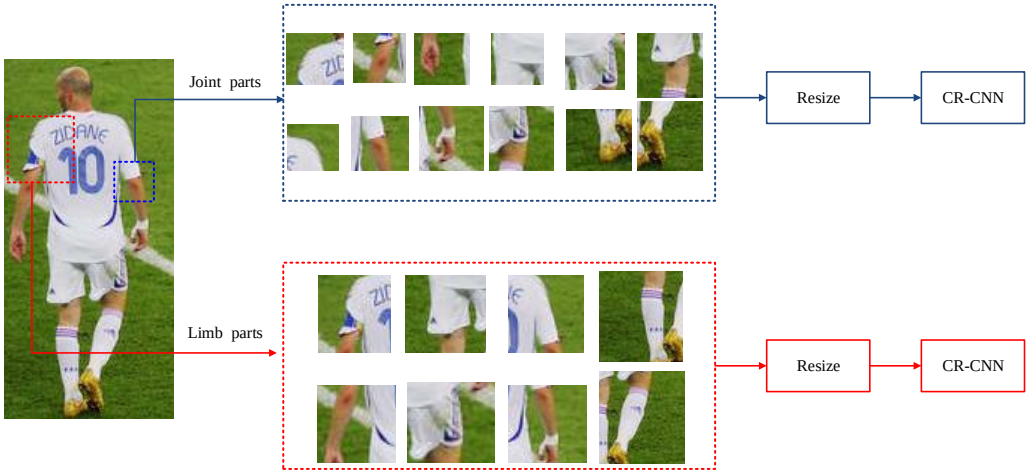


Figure A.9: Profonde modèle de branche hiérarchique. La boîte bleue montre des parties communes et le rouge montre des parties des membres.

se concentre sur les membres qui sont plus complexes dans la structure. Ici, la branche hiérarchique signifie que les variantes de tailles de modèle d'apparence entre les parties de branche et des pièces communes. En outre, l'architecture de l'apprentissage en profondeur est introduite dans le modèle hiérarchique pour pose estimation. Ainsi, nous appelons cette méthode aussi profond modèle de branche hiérarchique. Comme représenté sur la Fig. A.9, elle comporte deux parties: ordres de parties d'articulation et les parties de branche. En ce qui concerne les parties communes, les paramètres de la LMR-CNN sont similaires à la Fig. A.8. Néanmoins, les parties de branche sont plus grandes que les articulations des pièces et ne peuvent pas partager un réseau avec des pièces communes. Ainsi, un réseau supplémentaire est utilisé pour la formation des pièces de membre et est appelé *Limbnet*. L'entrée de Limbnet contient également trois échelles au niveau de l'articulation du coude: $(e'+o)*(e'+o), e'*e', (e'-o)*(e'-o)$. e' est l'échelle de centre de pièces de membre dans LMR-CNN. Patchs à trois niveaux de l'image donnée est redimensionnée dans la même taille $e'*e'$. Les parties communes et les parties de branche partagent le décalage o.

Dans le modèle proposé, le LMR-CNN est utilisé pour apprendre la fonction au lieu d'utiliser les filtres du HOG. Ainsi, le modèle d'apparence est basé sur le patch d'image locale $I(p_{i^*})$ à l'emplacement p_{i^*} de la partie i. La caractéristique $\phi(I,p_{i^*}), \phi(I,p_{k^*})$ dans Eq. A.24 peut être définie comme:

$$S_a(I,P) = \sum_{i^* \in V^*} \omega_{i^*}^{m_{i^*}} \cdot f(i^*|I,p_{i^*};\theta) + \sum_{k^* \in V_l^*} \omega_{k^*}^{m_{k^*}} \cdot f(k^*|I,p_{k^*};\theta'), \qquad \text{(A.26)}$$

où f est la distribution de probabilité conditionnelle de pièces i^*, k^* appris par LMR-CNN. θ désignent les paramètres appris sur la base de toutes les parties communes, tandis θ' sont les paramètres de réseau du membre. Pour le modèle dé-

formable, l'Eq. A.25 peut être réécrite comme:

$$S_d(I,P) = \sum_{i,j\in E} \omega_{ij}^{m_{ij}} \cdot \psi(p_i + r_{ij}^{m_{ij}} - p_j) + \sum_{i,j\in E} \lambda_{ij} f(m_{ij}|I,p_i,i;\theta,\theta') + \sum_{j,i\in E} \omega_{ji}^{m_{ji}} \cdot \psi(p_j + r_{ji}^{m_{ji}} - p_i) + \sum_{j,i\in E} \lambda_{ji} f(m_{ji}|I,p_j,j;\theta,\theta'), \quad \text{(A.27)}$$

où $f(m_{ij}|I,p_i,i;\theta,\theta')$ désigne la distribution pour la partie i avec le mélange par rapport m_{ij} dépendait des paramètres θ,θ', tandis que $f(m_{ji}|I,p_j,j;\theta,\theta')$ pour la partie j. λ_{ij} est le paramètre de poids.

A.4.3 Inférence et apprentissage

A.4.3.1 Inférence

Étant donné une image, le problème d'inférence est de trouver la pose optimale. Ici, la configuration optimale est définie en maximisant la fonction complète de score:

$$p^* = \arg\max_p S_a(I,P) + S_d(I,P) + b, \quad \text{(A.28)}$$

où p^* désigne l'emplacement des parties du corps, et b est le terme de polarisation pour le filtre de racine. Les scores profonds sont utilisés pour générer la détection multiple dans l'image I les achète un seuil et en appliquant la suppression non-maximale (NMS). Un retour en arrière est utilisé pour trouver l'emplacement et le type de chaque partie dans chaque configuration maximale.

A.4.3.2 Apprentissage

Nous considérons le problème de l'apprentissage des paramètres du modèle à partir d'images étiquetées avec des positions de partie. Ceci est le type de données disponibles dans les bases de données d'estimation de pose [56,125,127,128]. Chaque jeu de données contient des milliers d'images et chaque image contient les emplacements des pièces annotées.

Le modèle proposé se compose de trois ensembles de paramètres. Le premier est la moyenne des emplacements relatifs r qui est appris par l'algorithme K-means. Deuxièmement, il y a le paramètre θ,θ' de la période d'apparition appris par LMR-CNN. Troisièmement, il y a des paramètres de poids ω appris par SVM structuré.

Pendant la formation, nous avons accès à des images de formation avec la réalité de terrain conjointe corps / emplacements $\{I^t, P^t, M^t\}_{t=1}^T$. T est le nombre d'images d'apprentissage. I^t désigne l'image avec la valeur de vérité de terrain P^t, l'indice de mélange M^t.

Paramètres pour LMR-CNN: Dans la formation d'un LMR-CNN, chaque patch d'image locale $I(p_i^t)$ centrée sur un emplacement de la pièce annotée p_i dans l'exemple de l'image t. m_{ij}^t est le mélange entre la partie i et la partie j dans l'image I^t et représente la relation entre la partie i sa partie voisine j dans le modèle de déformation. En raison de la structure de l'arbre, la partie i pourrait avoir plus d'une partie voisine. Comme représenté sur laFig. A.10, joints comme du poignet et de

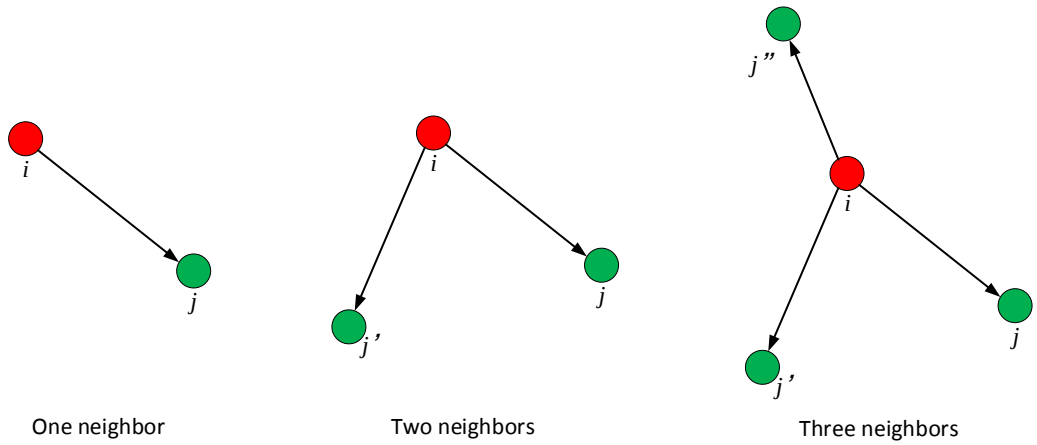


Figure A.10: Les différents nombres de voisins de l'articulation centrale. Les nez rouges indiquent l'articulation centrale, tandis que les nœuds verts sont les articulations voisines.

la cheville ont un voisin, les articulations comme les coudes avoir deux voisins, et les articulations comme les épaules avoir trois voisins. Quant à la formation CNN, nous avons besoin de diviser les types de partie i dépendais de toutes les parties voisines de la partie en cours. Cela peut ba définies comme suit:

$$\begin{cases} \mathbb{C}_{i\mathbb{N}(i)} = \{1, \cdots, M_{ij}\}, & \mathbb{N}(i) = \{j\} \\ \mathbb{C}_{i\mathbb{N}(i)} = \{1, \cdots, M_{ij}\} \times \{1, \cdots, M_{ij'}\}, & \mathbb{N}(i) = \{j, j'\} \\ \mathbb{C}_{i\mathbb{N}(i)} = \{1, \cdots, M_{ij}\} \times \{1, \cdots, M_{ij'}\} \times \{1, \cdots, & M_{ij''}\}, \\ & \mathbb{N}(i) = \{j, j', j''\} \end{cases} \tag{A.29}$$

où $\mathbb{N}(i)$ est un ensemble de voisins de parti i, $\mathbb{C}_{i\mathbb{N}(i)}$ désigner un ensemble de catégories de partie i dépendait de tous ses voisins. En fait, $\mathbb{C}_{i\mathbb{N}(i)}$ sont la combinaison de toutes les paires mélanges relatifs. $M_{ij}, M_{ij'}, M_{ij''}$ sont le nombre de mélanges relatifs à chaque connexion par paire. j, j', j'' désigner différents voisins de la partie i, de celui-ci a trois voisins.

Comme mentionné ci-dessus, le numéro de référence est décrit par $i^t \in \{1, \cdots, K+N\}$. Ainsi, chaque image a K des pièces ou des patchs. $c^t_{i^t\mathbb{N}(i^t)} \in \mathbb{C}^t_{i^t\mathbb{N}(i^t)}$ est le mélange relatif basé voisin-de la partie i dans l'exemple t. De cette façon, nous avons accès à des images de formation avec des endroits du corps de vérité terrain qui sont étiquetés comme un ensemble de correctifs $\mathbb{P} = \{I(p^t_i), i^t, c^t_{i^t\mathbb{N}(i^t)}\}^{i=K+N,t=T}_{i=1,t=1}$. Il est à noter que cet ensemble $\mathbb{P}$ peuvent être divisés en deux sous-ensembles $\mathbb{P}_1, \mathbb{P}_2$ pour des parties communes et des parties de branche respectivement. Semblable à d'autres méthodes de la formation CNN, les numéros de mélange des taches de fond à partir d'exemples négatifs sont définis comme 0. Nous avons effectué la descente de gradient stochastique de lot standard pour former ce multiclassent LMR-CNN.

Paramètre de poids: Pour illustrer l'apprentissage des paramètres de poids dans le modèle proposé, nous écrivons $Z_t = (P_t, M_t)$. La fonction de partition peut être exprimée en termes de produit scalaire, $S(I_t, Z_t) = \beta \cdot \Phi(I_t, Z_t)$, entre un vecteur de paramètres du modèle β et un vecteur de caractéristiques $\Phi(I_t, Z_t)$,

$$\begin{aligned}
&\beta = (\omega_1^{m_1}, \cdots, \omega_N^{m_N}, \omega_{N+1}^{m_{N+1}}, \cdots, \omega_{N+K}^{m_{N+K}}, \cdots, \omega_{ij}^{m_{ij}}, \cdots && \text{(A.30)}\\
&\cdots, \lambda_{ij}, \cdots, \cdots, \omega_{ji}^{m_{ji}}, \cdots, \cdots, \lambda_{ji}, \cdots, b), \quad (i,j \in E)\\
&\Phi(I_t, Z_t) = (f(1|I, p_1; \theta), \cdots, f(N|I, p_N; \theta), f(N+1|I, p_{N+1}; \theta'), \cdots, && \text{(A.31)}\\
&f(N+K|I, p_{N+K}; \theta'), \cdots, \psi(p_i + r_{ij}^{m_{ij}} - p_j), \cdots, \cdots, f(m_{ij}|I, p_i, i; \theta, \theta'), \cdots,\\
&\cdots, \psi(p_j + r_{ji}^{m_{ji}} - p_i), \cdots, \cdots, f(m_{ji}|I, p_j, j; \theta, \theta'), \cdots, 1), \quad (i,j \in E)
\end{aligned}$$

où N est le nombre des parties d'articulation et K est le nombre des parties de branche. Ainsi, nous définissons une grande marge d'apprentissage semblable au travail de Ref. 63, 138:

$$\begin{aligned}
\arg \min_{\beta, \xi_t \geq 0} \frac{1}{2}\|\beta\|^2 + \mathcal{C} \sum_t \xi_t, && \text{(A.32)}\\
s.t. \forall t \in pos \quad \langle \beta, \Phi(I_t, Z_t) \rangle \geq 1 - \xi_t,\\
\forall t \in neg \quad \langle \beta, \Phi(I_t, Z_t) \rangle \leq -1 + \xi_t,
\end{aligned}$$

où $\mathcal{C}$ contrôle du poids relatif de la période de régularisation et ξ_t représente les variables d'écart de la fonction objectif.

A.5 Conclusion

L'estimation de la pose humaine est un problème fondamental dans la vision par ordinateur. Dans cette thèse, nos contributions sont résumées comme suit. Tout d'abord, le modèle de mélanges flexibles est introduit. Il s'agit de faire coopérer le filtre à particules recuit (APF) pour le suivi des parties du corps et l'AMP. Le suivi et la détection sont combinées pour estimer et mettre à jour l'état de la pose. Deuxièmement, une base de modèles de mélange multiple du haut du corps (MMP) est proposé. Il consiste à diviser l'estimation de la pose en deux étapes: pré-estimation et estimation. Dans la phase de pré-estimation, la tâche principale est de trouver un modèle plus efficace et discriminante pour classer les poses. Dans l'étape de l'estimation, la tâche principale est de détecter chaque partie du corps en utilisant différents modèles en fonction du membre. Troisièmement, nous avons amélioré les performances de l'estimation de la pose dans le cadre de l'apprentissage profond. Un réseau local multi-résolution de convolution de neurones (LMR-CNN) est proposé pour apprendre la représentation de chaque partie du corps. Puis l'algorithme LMR-CNN est introduit dans un modèle hiérarchique.

Bibliography

[1] Y. Yang and D. Ramanan, "Articulated pose estimation with flexible mixtures-of-parts," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 1385–1392. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5995741 (Cited on pages xi, 16, 18, 20, 37, 38, 39, 43, 54, 55, 56, 57, 59, 75 and 78.)

[2] D. Hogg, "Model-based vision: a program to see a walking person," *Image and Vision computing*, vol. 1, no. 1, pp. 5–20, 1983. (Cited on pages 1 and 8.)

[3] J. O'Rourke and N. I. Badler, "Model-based image analysis of human motion using constraint propagation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, no. 6, pp. 522–536, 1980. (Cited on page 1.)

[4] A. Zhu, H. Snoussi, T. Wang, and A. Cherouat, "Human pose estimation with multiple mixture parts model based on upper body categories," *Journal of Electronic Imaging*, vol. 24, no. 4, p. 043021, 2015. [Online]. Available: http://dx.doi.org/10.1117/1.JEI.24.4.043021 (Cited on pages 1 and 16.)

[5] S. A. Johnson, "Articulated human pose estimation in natural images," Ph.D. dissertation, University of Leeds, 2012. (Cited on page 1.)

[6] Y. Yang and D. Ramanan, "Articulated human detection with flexible mixtures-of-parts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, pp. 1–14, 2012. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6380498 (Cited on pages 5 and 26.)

[7] D. Ramanan, "Learning to parse images of articulated bodies," in *Advances in neural information processing systems*, 2006, pp. 1129–1136. (Cited on pages 5 and 26.)

[8] T. B. Moeslund and E. Granum, "A survey of computer vision-based human motion capture," *Computer vision and image understanding*, vol. 81, no. 3, pp. 231–268, 2001. (Cited on page 8.)

[9] R. Poppe, "Vision-based human motion analysis: An overview," *Computer vision and image understanding*, vol. 108, no. 1, pp. 4–18, 2007. (Cited on page 8.)

[10] D. C. Hogg, "Interpreting images of a known moving object," Ph.D. dissertation, University of Sussex, 1984. (Cited on page 8.)

[11] T. B. Moeslund and E. Granum, "3d human pose estimation using 2d-data and an alternative phase space representation," *Procedure Humans 2000*, 2000. (Cited on page 8.)

[12] K. Rohr, "Human movement analysis based on explicit motion models," in *Motion-based recognition*. Springer, 1997, pp. 171–198. (Cited on pages 8 and 9.)

[13] E.-J. Ong and S. Gong, "Tracking hybrid 2d-3d human models from multiple views," in *Modelling People, 1999. Proceedings. IEEE International Workshop on*. IEEE, 1999, pp. 11–18. (Cited on page 8.)

[14] V. Pavlović, J. M. Rehg, T.-J. Cham, and K. P. Murphy, "A dynamic bayesian network approach to figure tracking using learned dynamic models," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 1. IEEE, 1999, pp. 94–101. (Cited on page 8.)

[15] D. M. Gavrila and L. S. Davis, "3-d model-based tracking of humans in action: a multi-view approach," in *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR'96, 1996 IEEE Computer Society Conference on*. IEEE, 1996, pp. 73–80. (Cited on page 9.)

[16] Y. Kameda, M. Minoh, and K. Ikeda, "Three dimensional pose estimation of an articulated object from its silhouette image," in *Asian Conference on Computer Vision*, 1993, pp. 612–615. (Cited on page 9.)

[17] C. Hu, Q. Yu, Y. Li, and S. Ma, "Extraction of parametric human model for posture recognition using genetic algorithm," in *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*. IEEE, 2000, pp. 518–523. (Cited on page 9.)

[18] N. Jojic, J. Gu, H. C. Shen, and T. Huang, "3-d reconstruction of multipart self-occluding objects," in *Computer Vision-ACCV1998*. Springer, 1997, pp. 455–462. (Cited on page 9.)

[19] I. Kakadiaris and D. Metaxas, "Vision-based animation of digital humans," in *Computer Animation 98. Proceedings*. IEEE, 1998, pp. 144–152. (Cited on page 9.)

[20] D. Meyer, J. Denzler, and H. Niemann, "Model based extraction of articulated objects in image sequences for gait analysis," in *Image Processing, 1997. Proceedings., International Conference on*, vol. 3. IEEE, 1997, pp. 78–81. (Cited on page 9.)

[21] C. Bregler, J. Malik, and K. Pullen, "Twist based acquisition and tracking of animal and human kinematics," *International Journal of Computer Vision*, vol. 56, no. 3, pp. 179–194, 2004. (Cited on page 9.)

[22] S. Wachter and H.-H. Nagel, "Tracking of persons in monocular image sequences," in *Nonrigid and Articulated Motion Workshop, 1997. Proceedings., IEEE*. IEEE, 1997, pp. 2–9. (Cited on page 9.)

[23] C. Sminchisescu and B. Triggs, "Estimating articulated human motion with covariance scaled sampling," *The International Journal of Robotics Research*, vol. 22, no. 6, pp. 371–391, 2003. (Cited on page 9.)

[24] ——, "Kinematic jump processes for monocular 3d human tracking," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 1. IEEE, 2003, pp. I–69. (Cited on page 9.)

[25] M. W. Lee and I. Cohen, "Proposal maps driven mcmc for estimating human body pose in static images," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2. IEEE, 2004, pp. II–334. (Cited on page 9.)

[26] T. Moeslund, *Pose estimating the human arm using kinematics and the sequential monte carlo framework*. INTECH Open Access Publisher, 2005. (Cited on page 9.)

[27] T. B. Moeslund, C. B. Madsen, and E. Granum, "Modelling the 3d pose of a human arm and the shoulder complex utilising only two parameters," *Integrated Computer-Aided Engineering*, vol. 12, no. 2, pp. 159–175, 2005. (Cited on page 9.)

[28] R. Navaratnam, A. Thayananthan, P. H. Torr, and R. Cipolla, "Hierarchical part-based human body pose estimation." in *BMVC*, 2005. (Cited on page 9.)

[29] Q. Delamarre and O. Faugeras, "3d articulated models and multiview tracking with physical forces," *Computer Vision and Image Understanding*, vol. 81, no. 3, pp. 328–357, 2001. (Cited on page 10.)

[30] P. Fua *et al.*, "Articulated soft objects for multiview shape and motion capture," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 9, pp. 1182–1187, 2003. (Cited on page 10.)

[31] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=978374 (Cited on pages 10 and 25.)

[32] A. Zhu, H. Snoussi, and A. Cherouat, "Articulated human motion tracking with foreground learning," in *Signal Processing Conference (EUSIPCO), 2014 Proceedings of the 22nd European*. IEEE, 2014, pp. 366–370. (Cited on page 10.)

[33] J. MacCormick and M. Isard, "Partitioned sampling, articulated objects, and interface-quality hand tracking," in *Computer Vision–ECCV 2000*. Springer, 2000, pp. 3–19. (Cited on page 10.)

[34] J. Deutscher, A. Blake, and I. Reid, "Articulated body motion capture by annealed particle filtering," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2000, pp. 126–133. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=854758 (Cited on pages 10, 28, 86 and 88.)

[35] J. Deutscher and I. Reid, "Articulated body motion capture by stochastic search," *International Journal of Computer Vision*, vol. 61, no. 2, pp. 185–205, 2005. [Online]. Available: http://link.springer.com/article/10.1023/B:VISI.0000043757.18370.9c (Cited on pages 10 and 36.)

[36] L. Sigal, A. O. Balan, and M. J. Black, "Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion," *International Journal of Computer Vision*, vol. 87, no. 1-2, pp. 4–27, 2010. [Online]. Available: http://link.springer.com/article/10.1007/s11263-009-0273-6 (Cited on pages 11, 25 and 34.)

[37] J. Carranza, C. Theobalt, M. A. Magnor, and H.-P. Seidel, "Free-viewpoint video of human actors," *ACM Transactions on Graphics (TOG)*, vol. 22, no. 3, pp. 569–577, 2003. (Cited on page 11.)

[38] R. Kehl, M. Bray, and L. Van Gool, "Full body tracking from multiple views using stochastic sampling," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2. IEEE, 2005, pp. 129–136. (Cited on page 11.)

[39] T. B. Moeslund, A. Hilton, and V. Krüger, "A survey of advances in vision-based human motion capture and analysis," *Computer vision and image understanding*, vol. 104, no. 2, pp. 90–126, 2006. (Cited on page 12.)

[40] H. Sidenbladh and M. J. Black, "Learning image statistics for bayesian tracking," in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 2. IEEE, 2001, pp. 709–716. (Cited on page 12.)

[41] H. Sidenbladh, M. J. Black, and D. J. Fleet, "Stochastic tracking of 3d human figures using 2d image motion," in *Computer Vision–ECCV 2000*. Springer, 2000, pp. 702–718. (Cited on page 12.)

[42] H. Sidenbladh, M. J. Black, and L. Sigal, "Implicit probabilistic models of human motion for synthesis and tracking," in *Computer Vision–ECCV 2002*. Springer, 2002, pp. 784–800. (Cited on page 12.)

[43] H. Sidenbladh and M. J. Black, "Learning the statistics of people in images and video," *International Journal of Computer Vision*, vol. 54, no. 1-3, pp. 183–209, 2003. (Cited on page 12.)

[44] I. Karaulova, P. M. Hall, and A. D. Marshall, "A hierarchical model of dynamics for tracking people with a single video camera." in *BMVC*, 2000, pp. 1–10. (Cited on page 12.)

[45] A. Agarwal and B. Triggs, "Tracking articulated motion with piecewise learned dynamical models," in *European Conference on Computer Vision*, vol. 3, 2004, pp. 54–65. (Cited on page 12.)

[46] N. R. Howe, M. E. Leventon, and W. T. Freeman, "Bayesian reconstruction of 3d human motion from single-camera video." in *NIPS*, vol. 99, 1999, pp. 820–6. (Cited on page 12.)

[47] L. Sigal, S. Bhatia, S. Roth, M. J. Black, and M. Isard, "Tracking loose-limbed people," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 1. IEEE, 2004, pp. I–421. (Cited on page 12.)

[48] R. Urtasun and P. Fua, "3d human body tracking using deterministic temporal motion models," in *Computer Vision-ECCV 2004*. Springer, 2004, pp. 92–106. (Cited on page 12.)

[49] R. Urtasun, D. J. Fleet, A. Hertzmann, and P. Fua, "Priors for people tracking from small training sets," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, vol. 1. IEEE, 2005, pp. 403–410. (Cited on page 12.)

[50] R. Urtasun, D. J. Fleet, and P. Fua, "3d people tracking with gaussian process dynamical models," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1. IEEE, 2006, pp. 238–245. (Cited on page 12.)

[51] J. Wang, A. Hertzmann, and D. M. Blei, "Gaussian process dynamical models," in *Advances in neural information processing systems*, 2005, pp. 1441–1448. (Cited on page 12.)

[52] K. Moon and V. Pavlović, "Impact of dynamics on subspace embedding and tracking of sequences," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1. IEEE, 2006, pp. 198–205. (Cited on page 12.)

[53] G. Mori, X. Ren, A. Efros, J. Malik *et al.*, "Recovering human body configurations: Combining segmentation and recognition," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2. IEEE, 2004, pp. II–326. (Cited on page 13.)

[54] D. A. Forsyth and M. M. Fleck, "Body plans," in *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*. IEEE, 1997, pp. 678–683. (Cited on page 13.)

[55] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition," *International Journal of Computer Vision*, vol. 61, no. 1, pp. 55–79, 2005. [Online]. Available: http://link.springer.com/article/10.1023/B:VISI.0000042934.15159.49 (Cited on pages 13, 15, 28 and 88.)

[56] M. Eichner and V. Ferrari, "Better appearance models for pictorial structures," in *Proceedings of the British Machine Vision Conference (BMVC)*, September 2009. (Cited on pages 13, 20, 51, 72 and 102.)

[57] M. A. Fischler and R. A. Elschlager, "The representation and matching of pictorial structures," *IEEE Transactions on computers*, no. 1, pp. 67–92, 1973. (Cited on pages 13, 14 and 15.)

[58] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient matching of pictorial structures," in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 2. IEEE, 2000, pp. 66–73. (Cited on pages 13 and 15.)

[59] X. Lan and D. P. Huttenlocher, "Beyond trees: Common-factor models for 2d human pose recovery," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, vol. 1. IEEE, 2005, pp. 470–477. (Cited on page 15.)

[60] R. Ronfard, C. Schmid, and B. Triggs, "Learning to parse pictures of people," in *Computer Vision–ECCV 2002*. Springer, 2002, pp. 700–714. (Cited on page 15.)

[61] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, June 2005, pp. 886–893 vol. 1. (Cited on pages 15, 44, 75 and 91.)

[62] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008, pp. 1–8. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4587597 (Cited on pages 16, 18, 44 and 91.)

[63] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010. (Cited on pages 16, 51, 74, 96 and 104.)

[64] Y. Wang, D. Tran, and Z. Liao, "Learning hierarchical poselets for human parsing," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 1705–1712. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2011.5995519 (Cited on pages 16, 18, 19, 43 and 56.)

[65] M. Sun and S. Savarese, "Articulated part-based model for joint object detection and pose estimation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, ser. ICCV '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 723–730. [Online]. Available: http://dx.doi.org/10.1109/ICCV.2011.6126309 (Cited on pages 16 and 19.)

[66] F. Wang and Y. Li, "Beyond physical connections: Tree models in human pose estimation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 596–603. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6618927 (Cited on pages 16, 18, 19, 43, 56, 57, 58, 59 and 78.)

[67] B. Sapp and B. Taskar, "Modec: Multimodal decomposable models for human pose estimation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 3674–3681. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6619315 (Cited on pages 16, 20, 54, 74 and 75.)

[68] A. Zhu, H. Snoussi, and A. Cherouat, "Articulated pose estimation via multiple mixture parts model," in *Advanced Video and Signal Based Surveillance (AVSS), 2015 12th IEEE International Conference on.* IEEE, 2015, pp. 1–5. (Cited on page 16.)

[69] Y. Tian, C. L. Zitnick, and S. G. Narasimhan, "Exploring the spatial hierarchy of mixture models for human pose estimation," in *Proceedings of European Conference on Computer Vision (ECCV)*, 2012, pp. 256–269. (Cited on pages 18, 19, 56 and 57.)

[70] D. Park and D. Ramanan, "N-best maximal decoders for part models," in *Computer Vision (ICCV), 2011 IEEE International Conference on.* IEEE, 2011, pp. 2627–2634. (Cited on pages 18 and 20.)

[71] N. Ukita, "Articulated pose estimation with parts connectivity using discriminative local oriented contours," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 3154–3161. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6248049 (Cited on pages 18 and 29.)

[72] L. Bourdev and J. Malik, "Poselets: Body part detectors trained using 3d human pose annotations," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 1365–1372. (Cited on pages 18 and 19.)

[73] L. Bourdev, S. Maji, T. Brox, and J. Malik, "Detecting people using mutually consistent poselet activations," in *Computer Vision–ECCV 2010*. Springer, 2010, pp. 168–181. (Cited on page 19.)

[74] L. Bourdev, S. Maji, and J. Malik, in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1543–1550. (Cited on page 19.)

[75] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele, "Poselet conditioned pictorial structures," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013, pp. 588–595. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6618926 (Cited on pages 19, 43, 56, 57 and 78.)

[76] V. Ramakrishna, D. Munoz, M. Hebert, J. A. D. Bagnell, and Y. A. Sheikh, "Pose machines: Articulated pose estimation via inference machines," in *Proceedings of European Conference on Computer Vision (ECCV)*, no. CMU-RI-TR-, July 2014. (Cited on pages 19, 57 and 78.)

[77] Y. Chen, L. Zhu, C. Lin, H. Zhang, and A. L. Yuille, "Rapid inference on a novel and/or graph for object detection, segmentation and parsing," in *Advances in Neural Information Processing Systems*, 2007, pp. 289–296. (Cited on page 19.)

[78] K. Duan, D. Batra, and D. J. Crandall, "A multi-layer composite model for human pose estimation." in *BMVC*, 2012, pp. 1–11. (Cited on page 19.)

[79] D. Ramanan, D. A. Forsyth, and A. Zisserman, "Strike a pose: Tracking people by finding stylized poses," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 271–278. (Cited on page 20.)

[80] M. Eichner, M. Marin-Jimenez, A. Zisserman, and V. Ferrari, "2d articulated human pose estimation and retrieval in (almost) unconstrained still images," *International Journal of Computer Vision*, vol. 99, 2012. (Cited on pages 20, 43, 44, 52, 53 and 91.)

[81] B. Sapp, D. Weiss, and B. Taskar, "Parsing human motion with stretchable models," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 1281–1288. (Cited on page 20.)

[82] K. Fragkiadaki, H. Hu, and J. Shi, "Pose from flow and flow from pose," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 2059–2066. (Cited on page 20.)

[83] A. Cherian, J. Mairal, K. Alahari, and C. Schmid, "Mixing body-part sequences for human pose estimation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Columbus, OH, États-Unis: IEEE, Jun. 2014. [Online]. Available: http://hal.inria.fr/hal-00978643 (Cited on page 20.)

[84] H.-J. Yoo, "Deep convolution neural networks in computer vision," *IEIE Transactions on Smart Processing & Computing*, vol. 4, no. 1, pp. 35–43, 2015. (Cited on page 21.)

[85] P. O. Glauner, "Deep convolutional neural networks for smile recognition," *arXiv preprint arXiv:1508.06535*, 2015. (Cited on pages 21 and 63.)

[86] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006. (Cited on page 21.)

[87] O. Alsharif and J. Pineau, "End-to-end text recognition with hybrid hmm maxout models," *arXiv preprint arXiv:1310.1811*, 2013. (Cited on page 21.)

[88] I. J. Goodfellow, Y. Bulatov, J. Ibarz, S. Arnoud, and V. Shet, "Multi-digit number recognition from street view imagery using deep convolutional neural networks," *arXiv preprint arXiv:1312.6082*, 2013. (Cited on page 21.)

[89] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic data and artificial neural networks for natural scene text recognition," *arXiv preprint arXiv:1406.2227*, 2014. (Cited on page 21.)

[90] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 1701–1708. (Cited on page 21.)

[91] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105. (Cited on page 21.)

[92] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 1717–1724. (Cited on page 21.)

[93] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 580–587. (Cited on page 21.)

[94] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Weakly supervised object recognition with convolutional neural networks," 2014. (Cited on page 21.)

[95] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," *arXiv preprint arXiv:1312.6229*, 2013. (Cited on page 21.)

[96] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 1725–1732. (Cited on page 21.)

[97] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems*, 2014, pp. 568–576. (Cited on pages 21 and 23.)

[98] T. Pfister, "Advancing human pose and gesture recognition," Ph.D. dissertation, University of Oxford, 2015. (Cited on page 21.)

[99] L. Cun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in Neural Information Processing Systems*. Morgan Kaufmann, 1990, pp. 396–404. (Cited on page 22.)

[100] Y. LeCun, F. J. Huang, and L. Bottou, "Learning methods for generic object recognition with invariance to pose and lighting," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2, 2004, pp. II–97–104 Vol.2. (Cited on page 22.)

[101] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?" in *Proc. International Conference on Computer Vision (ICCV'09)*. IEEE, 2009. (Cited on page 22.)

[102] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ser. ICML '09. ACM, 2009, pp. 609–616. [Online]. Available: http://doi.acm.org/10.1145/1553374.1553453 (Cited on page 22.)

[103] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," in *Advances in Neural Information Processing Systems 26*, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds., 2013, pp. 2553–2561. [Online]. Available: http://media.nips.cc/nipsbooks/nipspapers/paper_files/nips26/1210.pdf (Cited on page 22.)

[104] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," in *International Conference on Learning Representations (ICLR 2014)*. CBLS, April 2014. [Online]. Available:

http://openreview.net/document/d332e77d-459a-4af8-b3ed-55ba (Cited on page 22.)

[105] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *arXiv preprint arXiv:1411.4038*, 2014. (Cited on page 22.)

[106] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (Cited on page 22.)

[107] J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders, "Selective search for object recognition," vol. 104, no. 2. Springer US, 2013, pp. 154–171. [Online]. Available: http://dx.doi.org/10.1007/s11263-013-0620-5 (Cited on page 22.)

[108] N. Zhang, J. Donahue, R. Girshick, and T. Darrell, "Part-based r-cnns for fine-grained category detection," in *Computer Vision ECCV 2014*, ser. Lecture Notes in Computer Science, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., vol. 8689. Springer International Publishing, 2014, pp. 834–849. (Cited on page 22.)

[109] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Computer Vision–ECCV 2014*. Springer, 2014, pp. 346–361. (Cited on page 22.)

[110] X. Chen and A. Yuille, "Articulated pose estimation by a graphical model with image dependent pairwise relations," in *Advances in Neural Information Processing Systems (NIPS)*, 2014. (Cited on pages 22 and 78.)

[111] A. Toshev and C. Szegedy, "Deeppose: Human pose estimation via deep neural networks," in *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1653–1660. (Cited on page 22.)

[112] J. Tompson, M. Stein, Y. Lecun, and K. Perlin, "Real-time continuous pose recovery of human hands using convolutional networks," *ACM Trans. Graph.*, vol. 33, no. 5, pp. 169:1–169:10, Sep. 2014. [Online]. Available: http://doi.acm.org/10.1145/2629500 (Cited on page 22.)

[113] R. Fergus, G. Williams, I. Spiro, C. Bregler, and G. W. Taylor, "Pose-sensitive embedding by nonlinear nca regression," in *Advances in Neural Information Processing Systems*, 2010, pp. 2280–2288. (Cited on page 22.)

[114] A. Jain, J. Tompson, M. Andriluka, G. W. Taylor, and C. Bregler, "Learning human pose estimation features with convolutional networks," in *International Conference on Learning Representations (ICLR)*, April 2014. (Cited on page 22.)

[115] J. J. Tompson, A. Jain, Y. Lecun, and C. Bregler, "Joint training of a convolutional network and a graphical model for human pose estimation," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 1799–1807. (Cited on page 22.)

[116] X. Fan, K. Zheng, Y. Lin, and S. Wang, "Combining local appearance and holistic view: Dual-source deep neural networks for human pose estimation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. (Cited on page 22.)

[117] A. Jain, J. Tompson, Y. LeCun, and C. Bregler, "Modeep: A deep learning framework using motion features for human pose estimation," in *Computer Vision–ACCV 2014*. Springer, 2014, pp. 302–315. (Cited on page 23.)

[118] T. Pfister, K. Simonyan, J. Charles, and A. Zisserman, "Deep convolutional neural networks for efficient pose estimation in gesture videos," in *Computer Vision–ACCV 2014*. Springer, 2015, pp. 538–552. (Cited on page 23.)

[119] T. Pfister, J. Charles, and A. Zisserman, "Flowing convnets for human pose estimation in videos," *arXiv preprint arXiv:1506.02897*, 2015. (Cited on page 23.)

[120] R. Leonid and R. Michael, "Using gaussian process annealing particle filter for 3d human tracking," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, pp. 1–13, 2008. (Cited on pages 26 and 85.)

[121] N. J. Gordon, D. J. Salmond, and A. F. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," in *IEE Proceedings F (Radar and Signal Processing)*, vol. 140, no. 2, 1993, pp. 107–113. [Online]. Available: http://digital-library.theiet.org/content/journals/10.1049/ip-f-2.1993.0015 (Cited on pages 28 and 88.)

[122] A. Lopez and J. R. Casas, "Feature-based annealing particle filter for robust body pose estimation." in *Proceedings of International Conference on Computer Vision Theory and Applications(VISAPP)*, 2009, pp. 438–443. (Cited on page 28.)

[123] G. W. Taylor, L. Sigal, D. J. Fleet, and G. E. Hinton, "Dynamical binary latent variable models for 3d human pose tracking," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 631–638. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5540157 (Cited on page 31.)

[124] A. O. Balan, L. Sigal, and M. J. Black, "A quantitative evaluation of video-based 3d person tracking," in *Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on*. IEEE, 2005, pp. 349–356. (Cited on page 36.)

[125] V. Ferrari, M. Marin-Jimenez, and A. Zisserman, "Progressive search space reduction for human pose estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2008. (Cited on pages 39, 51, 52, 54, 72, 75 and 102.)

[126] S. Johnson and M. Everingham, "Learning effective human pose estimation from inaccurate annotation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2011, pp. 1465–1472. (Cited on pages 39, 55 and 56.)

[127] ——, "Clustered pose and nonlinear appearance models for human pose estimation," in *Proceedings of the British Machine Vision Conference (BMVC)*, 2010, doi:10.5244/C.24.12. (Cited on pages 39, 44, 51, 56, 72, 74, 91 and 102.)

[128] D. Tran and D. Forsyth, "Improved human parsing with a full relational model," in *Proceedings of European Conference on Computer Vision (ECCV)*, 2010, pp. 227–240. (Cited on pages 39, 51, 72 and 102.)

[129] V. N. Vapnik and A. Lerner, "Pattern recognition using generalized portrait method," *Automation and remote control*, vol. 24, pp. 774–780, 1963. (Cited on page 39.)

[130] C. P. Diehl and G. Cauwenberghs, "Svm incremental learning, adaptation and optimization," in *Proceedings of International Joint Conference on Neural Networks (IJCNN),Portland, OR, US, July*, vol. 4, 2003, pp. 2685–2690. (Cited on page 39.)

[131] M. Pontil and A. Verri, "Properties of support vector machines," *Neural Computation*, vol. 10, no. 4, pp. 955–974, 1998. (Cited on page 40.)

[132] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of ACM the fifth annual workshop on Computational learning theory (COLT), Pittsburgh, PA, USA, July*, 1992, pp. 144–152. (Cited on page 40.)

[133] C. Piciarelli, C. Micheloni, and G. L. Foresti, "Trajectory-based anomalous event detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1544–1554, 2008. (Cited on page 40.)

[134] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press:Chambridge,UK, 2000. (Cited on page 40.)

[135] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun, "Support vector machine learning for interdependent and structured output spaces," in *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004, p. 104. (Cited on page 42.)

[136] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2001, pp. I–511–I–518 vol.1. (Cited on pages 44, 45, 52, 91 and 93.)

[137] M. Andriluka, S. Roth, and B. Schiele, "Monocular 3d pose estimation and tracking by detection," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 623–630. (Cited on page 46.)

[138] M. Kumar, A. Zisserman, and P. Torr, "Efficient discriminative learning of parts-based models," in *Proceedings of International Conference on Computer Vision (ICCV)*, Sept 2009, pp. 552–559. (Cited on pages 51, 74, 96 and 104.)

[139] J. C. Niebles and L. Fei-Fei, "A hierarchical model of shape and appearance for human action classification," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on.* IEEE, 2007, pp. 1–8. (Cited on page 52.)

[140] L. Pishchulin, A. Jain, M. Andriluka, T. Thormahlen, and B. Schiele, "Articulated people detection and pose estimation: Reshaping the future," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012, pp. 3178–3185. (Cited on pages 54 and 75.)

[141] M. Kiefel and P. V. Gehler, "Human pose estimation with fields of parts," in *Computer Vision–ECCV 2014.* Springer, 2014, pp. 331–346. (Cited on pages 57 and 78.)

[142] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, 1943. (Cited on page 62.)

[143] D. O. Hebb, "The organization of behavior," 1949. (Cited on page 62.)

[144] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain." *Psychological review*, vol. 65, no. 6, p. 386, 1958. (Cited on page 62.)

[145] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of the national academy of sciences*, vol. 79, no. 8, pp. 2554–2558, 1982. (Cited on page 62.)

[146] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," DTIC Document, Tech. Rep., 1986. (Cited on page 62.)

[147] T. M. Mitchell, "Machine learning. wcb," 1997. (Cited on page 64.)

[148] A. Ng, "Machine learning," *Coursera*, 2014. (Cited on page 64.)

[149] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 2559–2566. (Cited on page 65.)

[150] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the ACM International Conference on Multimedia*. ACM, 2014, pp. 675–678. (Cited on page 75.)

[151] M. Eichner and V. Ferrari, "Appearance sharing for collective human pose estimation," in *Computer Vision–ACCV 2012*. Springer, 2012, pp. 138–151. (Cited on page 78.)

[152] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele, "Strong appearance and expressive spatial models for human pose estimation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 3487–3494. (Cited on page 78.)

[153] L. Fu, J. Zhang, and K. Huang, "Beyond tree structure models: A new occlusion aware graphical model for human pose estimation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1976–1984. (Cited on page 78.)

Aichun ZHU

Doctorat : Optimisation et Sûreté des Systèmes

Année 2016

Détection et suivi de la posture humaine dans les images fixes et les vidéos

L'estimation de la pose du corps humain est un problème difficile en vision par ordinateur et les actions de toutes les difficultés de détection d'objet. Cette thèse se concentre sur les problèmes de l'estimation de la pose du corps humain dans les images ou vidéo, y compris la diversité des apparences, les changements de scène et l'éclairage de fond de confusion encombrement. Pour résoudre ces problèmes, nous construisons un modèle robuste comprenant les éléments suivants. Tout d'abord, les méthodes top-down et bottom-up sont combinés à l'estimation pose humaine. Nous étendons le modèle structure picturale (PS) de coopérer avec filtre à particules recuit (APF) pour robuste multi-vues estimation de la pose. Deuxièmement, nous proposons plusieurs parties de mélange à base (MMP) modèle d'une partie supérieure du corps pour l'estimation de la pose qui contient deux étapes. Dans la phase de pré-estimation, il y a trois étapes: la détection du haut du corps, catégorie estimation du modèle pour le haut du corps, et la sélection de modèle complet pour pose estimation. Dans l'étape de l'estimation, nous abordons le problème d'une variété de poses et les activités humaines. Enfin, le réseau de neurones à convolution (CNN) est introduit pour l'estimation de la pose. Un Local Multi-résolution réseau de neurones à convolution (LMR-CNN) est proposé pour apprendre la représentation pour chaque partie du corps. En outre, un modèle hiérarchique sur la base LMR-CNN est défini pour faire face à la complexité structurelle des parties de branche. Les résultats expérimentaux démontrent l'efficacité du modèle proposé.

Mots clés : vision par ordinateur - détection du signal - posture - machines à vecteurs de support - réseaux neuronaux (informatique).

Articulated Human Pose Estimation in Images and Video

Human pose estimation is a challenging problem in computer vision and shares all the difficulties of object detection. This thesis focuses on the problems of human pose estimation in still images or video, including the diversity of appearances, changes in scene illumination and confounding background clutter. To tackle these problems, we build a robust model consisting of the following components. First, the top-down and bottom-up methods are combined to estimation human pose. We extend the Pictorial Structure (PS) model to cooperate with annealed particle filter (APF) for robust multi-view pose estimation. Second, we propose an upper body based multiple mixture parts (MMP) model for human pose estimation that contains two stages. In the pre-estimation stage, there are three steps: upper body detection, model category estimation for upper body, and full model selection for pose estimation. In the estimation stage, we address the problem of a variety of human poses and activities. Finally, a Deep Convolutional Neural Network (DCNN) is introduced for human pose estimation. A Local Multi-Resolution Convolutional Neural Network (LMR-CNN) is proposed to learn the representation for each body part. Moreover, a LMR-CNN based hierarchical model is defined to meet the structural complexity of limb parts. The experimental results demonstrate the effectiveness of the proposed model.

Keywords: computer vision - signal detection - posture - support vector machines - neural networks (computer science).

Thèse réalisée en partenariat entre :

utt
université de technologie
Troyes

Ecole Doctorale "Sciences et Technologies"