Bridging the Gap between Detection and Tracking for 3D Human Motion Recovery

THÈSE N^{O} 4699 (2010)

PRÉSENTÉE LE 4 JUIN 2010
À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS
LABORATOIRE DE VISION PAR ORDINATEUR
PROGRAMME DOCTORAL EN INFORMATIQUE, COMMUNICATIONS ET INFORMATION

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Andrea FOSSATI

acceptée sur proposition du jury:

Prof. M. Hasler, président du jury
Prof. P. Fua, directeur de thèse
Dr R. Boulic, rapporteur
Prof. A. Hilton, rapporteur
Prof. B. Leibe, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE
Suisse
2010

Abstract

The aim of this thesis is to build a system able to automatically and robustly track human motion in 3–D starting from monocular input. To this end two approaches are introduced, which tackle two different types of motion: The first is useful to analyze activities for which a characteristic pose, or key-pose, can be detected, as for example in the walking case. On the other hand the second can be used for cases in which such pose is not defined but there is a clear relation between some easily measurable image quantities and the body configuration, as for example in the skating case where the trajectory followed by a subject is highly correlated to how the subject articulates.

In the first proposed technique we combine detection and tracking techniques to achieve robust 3D motion recovery of people seen from arbitrary viewpoints by a single and potentially moving camera. We rely on detecting key postures, which can be done reliably, using a motion model to infer 3D poses between consecutive detections, and finally refining them over the whole sequence using a generative model. We demonstrate our approach in the cases of golf motions filmed using a static camera and walking motions acquired using a potentially moving one. We will show that this approach, although monocular, is both metrically accurate because it integrates information over many frames and robust because it can recover from a few misdetections.

The second approach is based on the fact that the articulated body models used to represent human motion typically have many degrees of freedom, usually expressed as joint angles that are highly correlated. The true range of motion can therefore be represented by latent variables that span a low-dimensional space. This has often been used to make motion tracking easier. However, learning the latent space in a problem independent way makes it non trivial to initialize the tracking process by picking appropriate initial values for the latent variables, and thus for the pose. In this thesis, it will be shown that by directly using observable quantities as latent variables, this issues can be eliminated.

Index Words: Human Body Detection and Tracking, Motion Models, Low-Dimensional Embedding.

Sommario

Lo scopo di questa tesi é di costruire un sistema in grado di tracciare automaticamente e in maniera robusta il movimento umano in 3 dimensioni, a partire da una singola sequenza video. Per questo obiettivo sono introdotti due approcci, che sono volti ad analizzare due differenti tipologie di movimento: Il primo é utile per studiare attivit´a per le quali é definibile una posa caratteristica, o posa chiave, come per esempio nel camminare. Il secondo invece puó essere utilizzato per casi nei quali tale posa non é definibile ma vi é una chiara relazione tra alcune quantit´a direttamente misurabili nell'immagine e la configurazione del corpo, come ad esempio nel caso del pattinaggio nel quale la traiettoria seguita dal soggetto analizzato é strettamente correlata a come muove le sue articolazioni.

Nel primo algoritmo proposto vengono combinate tecniche di rilevazione e tracciamento per stimare in maniera robusta il movimento di persone riprese da un punto di vista arbitrario tramite una singola videocamera, eventualmente mobile. Esso si basa sul rilevamento di posture base, che puó essere effettuato in maniera affidabile, e utilizza modelli di movimento per stimare le pose 3 dimensionali tra rilevamenti consecutivi. Queste stime sono infine ottimizzate su tutta la sequenza utilizzando un modello generativo. Il funzionamento di tale approccio é stato dimostrato su sequenze di golf, acquisite tramite una videocamera statica, e su sequenze di persone che camminano, utilizzando una videocamera in movimento. Questa tecnica, seppure non stereo, é sia precisa metricamente, dato che integra informazioni provenienti da molteplici immagini, sia robusta perch´e anche in grado di sopperire a delle eventuali mancate rilevazioni

Il secondo approccio é invece basato sul fatto che i modelli articolati che sono in genere utilizati per rappresentare il movimento umano hanno numerosi gradi di libert´a, solitamente espressi in termini di angoli delle articolazioni che sono tra loro correlati. Per questo motivo il reale spettro di movimenti pu´o essere rappresentato da variabili di minore dimensionalitá in uno spazio sottodimensionale. Questa soluzione é stata spesso adottata per rendere piú semplice la stima del movimento umano. Tuttavia, l'apprendimento di tale rapprensentazione sottodimensionale in maniera indipendente dal tipo di problema rende molto complessa l'inizializzazione del processo di tracciamento, in particolare nella selezione dei valori iniziali per le variabili sottodimensionali. In questa tesi verrá invece mostrato come, utilizzando come variabili sottodimensionali delle misure direttamente rilevabili dalle immagini, questo problema possa essere risolto.

Parole Chiave: Tracciamento del corpo umano, Modelli di movimento, Rappresentazione sottodimensionale.

Acknowledgments

This Thesis has been a very pleasant journey and for this reason I would like to thank all the people that have made it possible. First of all I say thank you to my supervisor, Prof. Pascal Fua, who took me to the CVLab and allowed me to do study and research on the field that interested me more. His inspiration and suggestions helped me a lot in developing my knowledge and most importantly the research attitude who will be a strong basis for my future career. I would also like to thank Dr. Vincent Lepetit whose support and advice have been extremely helpful for my Thesis, specially during Pascal's sabbatical year. Furthermore these years of study would have been much harder without the help of our secretary Josiane, who has been able to solve small and big practical problems that without her would have still remained unsolved. Another big thank you goes to all my former and present colleagues at CVLab, which made my daily stay at the lab more enjoyable and helped me with discussions that were always interesting, without forgetting the wonderful karting races and laser game competitions we had together. Moreover I would like to thank all the members of the jury for accepting this task and for giving me very useful comments to improve the Thesis quality.

On the personal side I would like to thank a lot all my friends, both in Lausanne (specially all the people in Renens Basket) and in Italy, whose help has always been available and who always gave me reasons to cross the border in one direction or in the other. Thanks also to those friends which have been travelling back and forth with me during all these years.

Finally I have to keep the biggest thanks to my family, specially my mum, my dad

and my sister, and to my other half Rita, because without them all this journey would have definitely not been possible.

Contents

List of Abbreviations **18**

1 Introduction **19**
1.1 Motivation 21
1.2 Goal and Contributions 22
1.3 Thesis Structure 25

2 State of the Art **27**
2.1 Commercial Motion Capture Devices 28
2.2 Video-Based Motion Capture 29
2.2.1 Multi-View Human Pose Estimation 31
2.2.2 People Detection 34
2.2.3 People Tracking 37
2.2.4 Approaches combining Detection and Tracking 41

3 Human Body and Motion Parameterization **45**
3.1 Human Body Model 45
3.2 Pose and Motion Definition 47
3.3 PCA on Poses 50
3.4 PCA on Sequences 52

4 Bridging the Gap between Detection and Tracking **55**
4.1 Approach 56
4.2 Key-Pose Detection 58
4.3 Linking Detections to obtain 2D Trajectories 61
4.4 Interpolation of 3D Poses between Detections 67
4.5 Image Likelihood for Refinement 69
4.5.1 Appearance-based Likelihood 71
4.5.2 Generalized Expectation Maximization for Limb to Edge Matching 74
4.6 Optimization 82
4.6.1 Frame-Wise Optimization 82
4.6.2 Sequence-Based Optimization 82
4.7 Results 86
4.7.1 Golfing 87
4.7.2 Walking 91

5 Observable Subspaces 101
- 5.1 Framework 102
  - 5.1.1 Motion Representation 103
  - 5.1.2 Gaussian Processes 103
  - 5.1.3 Linear Subspace Motion Model 105
  - 5.1.4 Fitting the Model to Image Data 105
- 5.2 Experimental Results 109
  - 5.2.1 Obtaining Training Data 110
  - 5.2.2 Retrieving Trajectory Parameters 112
  - 5.2.3 Motion Recovery 112

6 Conclusions 123

List of Figures

1.1 Human body tracking. **First Column:** Input images. **Second Column:** Results overlaid on the input images. **Third Column:** Results seen from a different 3–D perspective. 20

1.2 Examples of issues. **(a) Reflection ambiguity and self-occlusion:** If considering the silhouette only it is not clear which leg is in the front. Moreover the body occludes one of the arms. **(b) Cluttered background. (c) Loose clothes.** 21

1.3 Examples of Motion Capture systems. **(a) Optical based system.** Several reflective markers are placed on a special suit, and their 3D position is recovered through a set of infrared cameras, allowing an easy retrieval of the body pose. **(b) Inertial system.** Small gyroscopes are attached to the body limbs, in order to recover their position and orientation. Data are then sent to a central computer for real-time elaboration. 23

2.1 Commercial Motion Capture systems. **(a)** PhaseSpace Impulse System, which is also based on reflective markers. An example of reconstructed and animated pose is shown. **(b)** Polhemus Liberty System, based on an electromagnetic field. It is capable to track sensors' in 6 degrees of freedom at a 240Hz rate. **(c)** MotionStar Ascension System. It is a wearable system but as can be noticed from the image it is quite cumbersome for the subject. **(d)** Xsens MVN is a wearable inertial based motion capture tracker. Small gyroscopes are embedded in the suit, to allow an accurate motion reconstruction of the subject. 30

2.2 Multi-camera settings in engineered studios. (a) The Multi-Camera and Multi-Lighting Dome at the Tsinghua University of Beijing, China. (b) The Grimage system at the INRIA-Rhone Alpes in Grenoble, France. . . 33

2.3 The poselets approach presented in [L. Bourdev 09]. The poselets correspond to parts that are clustered both in the appearance and configuration space. The figure shows examples of the Frontal Face, Right Arm crossing Torso, Pedestrian, Right Profile, and Shoulder and Legs Frontal View poselets. 36

2.4 Tracking results of the approach presented in [Rosenhahn 07], where a 3–D model of the subject is available and tracked motion patterns are matched to training ones to estimate future poses. 39

2.5 Body Pose Estimation combining Detection and Tracking. (a) Tracking algorithm proposed in [Ramanan 06]. It tracks the subject in 2–D in very long sequences. (b) The approach presented in [Andriluka 08] can handle multiple pedestrians and large occlusions but the pose estimation is quite rough. (c) Pose estimation results obtained with the algorithm presented in [Gammeter 08], which requires a narrow-baseline stereo stream as input. 43

3.1 Cylinder-based representation used as human body model in our experiments. 46

3.2 The SCAPE model has been developed by Stanford University and Intel through laser scan on several thousands of real people [Anguelov 05]. . . 47

3.3 (a) Superquadrics models used in [Sminchisescu 03b]. (b) Implicit surface reconstruction of the human body, proposed in [Plänkers 03]. (c) 3D mesh representing the human body reconstructed in a multi-camera setup, as proposed in [Carranza 03]. (d) Another 3D mesh reconstruction of the human body from multiple video streams, proposed in [Starck 03]. 48

3.4 Skeleton representing the joint angles used in our body pose estimation algorithms. The markers correspond to the body joints, and the configuration of each one of the joints is described by 3 Euler angles. 49

3.5 Low-dimensional embedding of the pose space. Each point in the latent PCA space, on the left, corresponds to a body pose in the high dimensional body configuration space, on the right. 51

3.6 Low-dimensional embedding of the motion space. Each point in the latent PCA space, on the left, corresponds to sequence of poses in the high dimensional motion space, on the right. 53

4.1 Our approach. **First row:** Input sequence acquired using a moving camera with silhouettes detected at the beginning and the end of the walking cycle. The projection of the ground plane is overlaid as a blue grid. **Second row:** Projections of the 3–D poses inferred from the two detections. **Third row:** Synthesized images that are most similar to the input. **Fourth row:** Projections of the refined 3–D poses. **Fifth row:** 3–D poses seen from a different viewpoint. 57

4.2 Key pose detections at the beginning and at the end of a golf swing. . . . 59

4.3 Creating spatio-temporal templates. (a) Eight virtual cameras are placed around the model. Viewpoints 3 and 7 are not considered because they are not discriminant enough, resulting in six different training views. (b) A template corresponding to a particular view consists of several silhouettes computed at three consecutive instants. The small blue arrows in image Camera 1 / Frame 1 represent edge orientations used for matching silhouettes for some of the contour pixels. (c) The three silhouettes of a walking template are superposed to highlight the differences between outlines. (d) Superposed silhouettes of a golf swing template. 60

4.4 **Ground plane tracking.** Given the corresponding interest points in each pair of consecutive frames in the sequence $I_i, i = 0..N$ it is possible to compute the pairwise homographies $H^i_{i+1}, i = 0..N-1$ between consecutive frames. Furthermore, knowing the initial homography between the world ground plane and the reference image H^w_0, we compute the required homographies $H^w_i, i = 1..N$ between each of the frames and the world ground plane 63

4.5 **Transitional probabilities for the hidden state** (X, Y, O)**.** They are represented by three Gaussian distributions corresponding to three possible previous orientations. Each Gaussian covers a 2D area bounded by two circles of radii $d_c - \delta d_c$ and $d_c + \delta d_c$, where δd_c represents an allowable deviation from the mean, and by two lines defined by tolerance angle $\delta\varphi$. 64

4.6 Filtering silhouettes with temporal consistency on an outdoor sequence acquired by a moving camera. **First two rows:** Detection hypotheses. **Third and fourth row:** Detections after filtering out the detection hypotheses that do not lie on the recovered most probable trajectory. Note that the extremely similar poses in which is very hard to distinguish which leg is in the front are successfully disambiguated by our algorithm. 68

4.7 Recovered trajectory for the sequence depicted by Fig. 4.6. Dark blue squares represent detection hypotheses and bright short lines inside them represent the detection orientations. Smaller light green squares and lines represent the retained detections and their orientations respectively. These detections form the most probable trajectory depicted by dark blue lines. . 69

4.8 Filtering silhouettes with temporal consistency on an outdoor sequence acquired by a moving camera. **First two rows:** Detection hypotheses. **Third and fourth row:** Detections after filtering out the detection hypotheses that do not lie on the recovered most probable trajectory. 70

4.9 Recovered hypothetical trajectory for the sequence depicted by Fig. 4.8. Dark blue squares represent detection hypotheses and bright short lines inside them represent the detection orientations. Smaller light green squares and lines represent the retained detections and their orientations respectively. These detections form the most probable trajectory depicted by dark blue lines. 71

4.10 Synthesizing a background image. **First row:** The central image is the reference image whose background we want to synthesize. The other four are those before and after it in the sequence. **Second row:** The same four images warped to match the reference image. Computing the median image of these and the reference image yields the central image, which is the desired background image. 72

4.11 Appearance based refinement process. The images are from left to right the input image, the initialization given by the interpolation process, the result obtained without using a background image, and finally the result obtained as proposed. Whole parts of the body can be missed when the background is not exploited. 73

4.12 Summary of the pre-processing algorithm: (a) Input image. (b) Edges extracted from the input image. (c) Background edges reconstructed through homographies. (d) Subtraction between (b) and (c). (e) Outliers retrieved by the camera motion estimation technique. (f) Final output of the algorithm, obtained as intersection between (d) and (e). 77

4.13 The continuous curve represents the real trajectory of the subject, while the dashed lines show its approximation by finite differences. 85

4.14 Reconstructing a golf swing performed by a professional player. **First and third row:** Frames from the input video with reprojected 3D skeletons. **Second and fourth row:** 3D skeleton seen from a different viewpoint. 88

4.15 Reconstructing a golf swing performed by a novice player. **First row:** Frames from the input video with reprojected 3D skeletons. **Second row:** 3D skeleton seen from a different a different viewpoint. 88

4.16 Background image used to generate the results of Fig. 4.14. Notice that there are some artifacts for instance in the feet area, which are anyway overcome by our algorithm. 89

4.17 Recovered 3D skeletons reprojected into individual images of the sequence of Fig. 4.6, which was acquired by a camera translating to follow the subject. 89

4.18 Final result for the subject of Fig. 4.8 who moves away from the camera and is eventually seen from behind. **First and third rows:** Frames from the input video with reprojected 3D skeletons. **Second and fourth rows:** 3D skeletons seen from a different viewpoint. The 3–D pose is correctly estimated over the sequence, even when the person goes far away and eventually turns his back to the camera. Note that the extremely similar poses in which it is very hard to distinguish which leg is in front are successfully disambiguated by our algorithm. 90

4.19 Subject walking in a circle. **First and third rows:** Frames from the input video with reprojected 3D skeletons. **Second and fourth rows:** 3D skeletons seen from a different viewpoint. The numbers in the bottom right corner are the instantaneous speeds derived from the recovered motion parameters. 93

4.20 Recovered 2D trajectory of the subject of Fig. 4.19. The underlying grid is made of 1×1 meter squares and the arrows represent the direction he is facing. (a) When orientation and motion are not linked, he appears to walk sideways. (b) When they are, he walks naturally. 94

4.21 Pedestrian tracking and reprojected 3D model in a second sequence. **First and third rows:** Frames from the input video with reprojected 3D skeletons. **Second and fourth rows:** 3D skeletons seen from a different viewpoint. The numbers in the bottom right corner are the instantaneous speeds derived from the recovered motion parameters. 95

4.22 Recovered 2D trajectory of the subject of Fig. 4.21. As in Fig. 4.20, when orientation and motion are not linked, he appears to walk sideway (a) but not when they are (b). 96

4.23 Robustness to misdetection. **First two rows:** Initial and refined poses for a sequence in which 3 consecutive key-poses are detected. **Last two rows:** Initial and refined poses for the same sequence when ignoring the central detection and using the other two. The initial poses are less accurate but the refined ones are indistinguishable. 97

4.24 Absolute mean 3D error in joint location obtained on frames 21-248 of the HumanEvaII dataset for subject S4 and using only camera C1 as input. It is expressed in millimeters. 98

4.25 Tracking subject S4 from the HumanEvaII dataset using only camera C1. Obtained results projected onto the input frames. 98

5.1 Overview of the runtime procedure to recover the body pose, explained in Section 5.1.4. 107

5.2 Input silhouettes used to compute our results. The silhouettes were extracted using a standard background subtraction technique on skating and golfing examples, while on skiing an intensity threshold was instead applied. 108

5.3 Powell's Method. The figure shows one iteration of Powell's Method on a sample 2-dimensional function. 109

5.4 (a) For walking, we use a step low dimensional parameterization, including curvature of the trajectory, x and y displacement and a binary variable indicating which foot is moving. (b) For skating, we show two consecutive skating motions that correspond to the test sequence of Figure 5.6. The blue dots represent the tracked average feet locations on the ground plane and the black line is a second order polynomial fitted to them. The underlying grid is composed of 20cm$\times$20cm squares. The first latent parameter is the curvature of the polynomial, whose sign changes if the subject is turning left or right. The second one is a binary variable indicating if the subject is in the first or second half of the turn. (c) For golfing, we use hands' motion, corresponding to the second golfing sequence of Figure 5.11. The blue dots depict the tracked hand locations, while the 3 lines show the polynomials fitted to the different phases of the swing, whose 3 curvatures are the latent parameters. 113

5.5 (a) Average frame-wise error for the sequence of Figure 5.6, in degrees. (b) Mean errors for different joint angles, in degrees, averaged throughout the sequence. The bars represent the standard deviations of the errors. . . 115

5.6 Roller skating in a studio setup. **First row:** We reprojected the recovered body poses in the input images. **Second row:** Zoomed version of the first row. **Third row:** To highlight the 3D nature of the results, we display the 3D skeleton seen from a different viewpoint. 116

5.7 Roller skating. **First and fourth rows:** Recovered body pose reprojected in the input image. **Second and fifth rows:** Zoomed versions of the first and fourth rows, respectively. **Third and sixth rows:** 3D skeleton of the subject seen from a different viewpoint. 118

5.8 We used the model trained on skating motions to recover a skiing one. **First row:** Recovered body pose reprojected in the input image. **Second row:** Zoomed versions of the first row. **Third row:** 3D skeleton of the subject seen from a different viewpoint. 119

5.9 Recovered pose in the walking case for 4 steps. The resulting estimated pose is shown in red for the first and the last frame of each step. In the case of the last step it is projected over the walking subject. 119

5.10 Estimated pose for the whole curvilinear walking trajectory. The top image shows the input sequence while the bottom image displays the corresponding body poses. 120

5.11 Golf swing tracking. **First and third rows:** Two different subjects performing a golf swing. The recovered body poses have been reprojected in the input images. **Second and fourth rows:** The 3D skeleton of the person is seen from a different viewpoint. 121

List of Abbreviations

EM	Expectation Maximization
GEM	Generalized Expectation Maximization
GP	Gaussian Process
GPDM	Gaussian Process Dynamical Model
GPLVM	Gaussian Process Latent Variable Model
GPS	Global Positioning System
HSV	Hue, Saturation and Value
Km/h	Kilometers per hour
LED	Light Emitting Diod
MCMC	Mean Field Monte Carlo
MoCap	Motion Capture
PCA	Principal Component Analysis
RBF	Radial Basis Function
SCAPE	Shape Completion and Animation of People

Chapter 1

Introduction

Retrieving 3–D human body poses from monocular video sequences, as depicted by Fig. 1.1, has many potential applications, such as video surveillance, animations for the entertainment industry or performance analysis in sports, just to cite a few examples, and has been addressed by the Computer Vision community for several years. The monocular approach has the advantage, over multi-camera motion capture, of being much less expensive and also much easier to deploy, not needing ad-hoc engineered studios.

Unfortunately 3–D human pose estimation from a single sequence is often poorly-constrained, due to reflection ambiguities, self-occlusion, cluttered backgrounds, non-rigidity of tissue and clothing, and poor resolution. Some of such issues are shown in Fig. 1.2. As a consequence some kind of prior information is needed to resolve ambiguities, minimize estimator variance and to cope with occlusions. Moreover a good way to exploit the images is also necessary, so that the tracking does not only depend on priors but also on accurate and robust image measurements.

In this thesis we focus on robust tracking over long sequences. This is well beyond

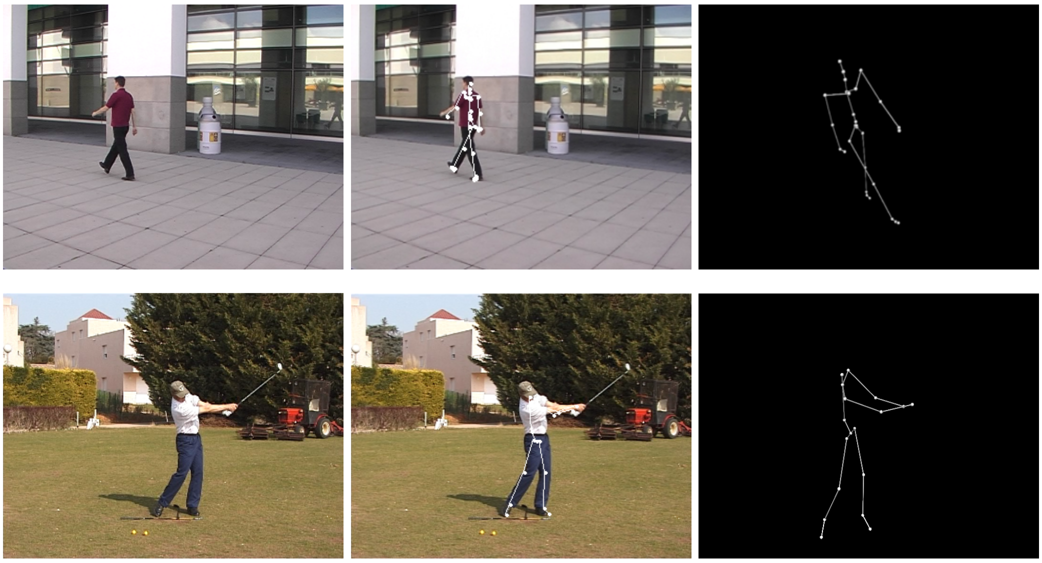
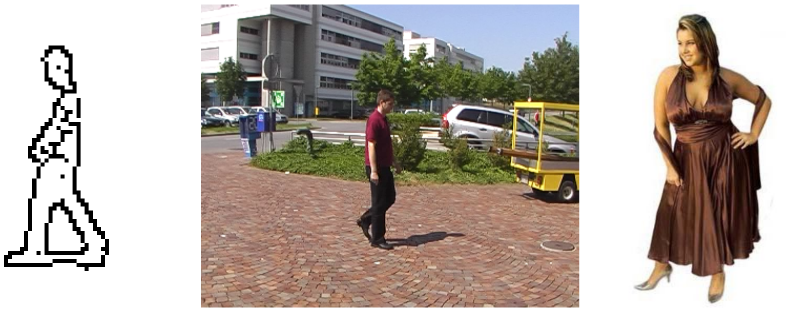


Figure 1.1: Human body tracking. **First Column:** Input images. **Second Column:** Results overlaid on the input images. **Third Column:** Results seen from a different 3–D perspective.

the state of the art, which typically lacks robustness and involves very short sequences. To this end, we have developed two algorithms able to handle different kinds of human motion: For activities in which a characteristic pose, or key-pose, can be defined, as for example in the walking case, we studied a technique to automatically initialize and re-initialize the system after tracking failures, and more effectively exploit the image-data, even in case of moving camera. On the other hand the second algorithm can be used for cases in which such key-pose is not defined but there is a clear relation between some easily measurable image quantities and the body configuration, as for example in the skating case where the trajectory followed by a subject is highly correlated to how the subject articulates. This approach allows the use of such directly measurable image quantities to automatically infer a valuable estimation of the body pose.



(a) (b) (c)

Figure 1.2: Examples of issues. **(a) Reflection ambiguity and self-occlusion:** If considering the silhouette only it is not clear which leg is in the front. Moreover the body occludes one of the arms. **(b) Cluttered background. (c) Loose clothes.**

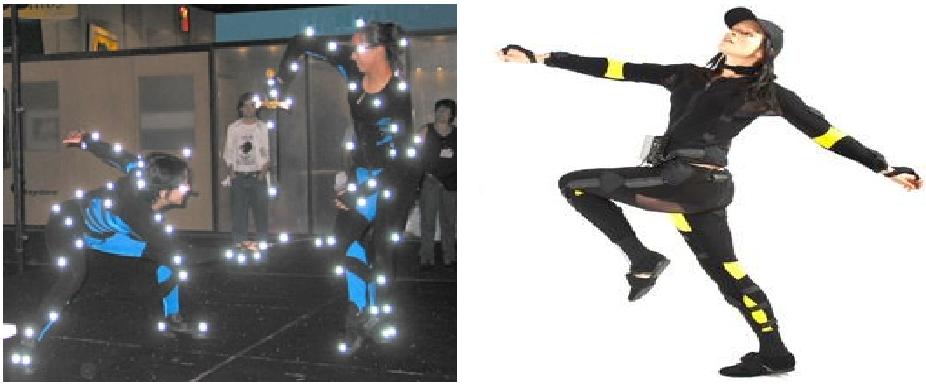
1.1 Motivation

Motion Capture applications range from the entertainment industry, including movies and computer games, to surveillance, sport performance analysis and clinical studies. Several recent movies like Pirates of the Caribbean and Harry Potter, and also computer animated stories like Finding Nemo and Wall-E, made use of motion capture data to animate their characters in a more realistic and artistic way. Many computer games, such as the NBA Live series and Pro Evolution Soccer to cite only a couple of best sellers, make intensive use of motion capture to increase realism. Examples of surveillance applications are detection of atypical motions, the use of gait signature for recognition, and the monitoring of physically impaired people. Finally motion capture systems are nowadays being used to analyze performances and detect mistakes in sports such as golf and tennis, and also to study the effect of different types of prosthesis in clinical studies. Recently there have also been some trials of using cameras to take sequences of moving people as a Human-Machine interface, as all the efforts put in the Microsoft NATAL project show.

Over the years, because there are so many applications to tracking the human body in 3 dimensions a number of approaches have been proposed, some of which are nowadays commercially adopted: For example the VICON [h] system relies on reflective markers placed on a special suit, whose 3D positions are then captured using a set of infrared cameras (see Fig. 1.3(a)). Another approach involves directly placing accelerometers and gyroscopes on the subject's body in order to accurately collect data about his/her motion [b, f], as shown in Fig. 1.3(b). In recent years, multi-camera systems have also appeared. By engineering the environment and using many cameras, they can retrieve the 3D movement of one or more persons [Mitchelson 03, Starck 03, Horaud 09]. Unfortunately, these systems are either invasive or require a very expensive set-up, both in terms of money and time. For these reasons, a system that could capture 3D human motion simply from standard video sequences would be a great improvement. Furthermore, it could also be used to process pre-existing videos.

1.2 Goal and Contributions

In this work, we focused on designing and implementing fully automated systems able to robustly track in 3–D people in monocular sequences of theoretically arbitrary length. The approaches we developed include the following features:



(a) (b)

Figure 1.3: Examples of Motion Capture systems. **(a) Optical based system.** Several reflective markers are placed on a special suit, and their 3D position is recovered through a set of infrared cameras, allowing an easy retrieval of the body pose. **(b) Inertial system.** Small gyroscopes are attached to the body limbs, in order to recover their position and orientation. Data are then sent to a central computer for real-time elaboration.

- *Robustness to long sequences:* Most of the state of the art systems present results on short sequences, the main issue being drift. Our algorithm on the other hand does not suffer from this problem and, thanks to the detection phase, can easily recover from tracking failures.

- *Automatic initialization:* Another common weakness of human body tracking systems is the initialization. Usually this is done manually, specially in single camera settings. By contrast, our algorithm performs initialization fully automatically.

- *Intuitive parameterization of the motion:* The articulated body models used to represent human motion typically have many degrees of freedom, usually expressed as joint angles that are highly correlated. The true range of motion can therefore be represented by latent variables that span a low-dimensional space. This has often been used to make motion tracking easier. However, learning the latent space in a problem independent way makes it non trivial to initialize the tracking process by picking appropriate initial values for the latent variables, and thus for the pose. In our work, we have shown that by directly using observable quantities as latent variables, this issues can be eliminated.

To achieve such challenging tasks we have introduced some new technical ideas to the tracking framework, that helped us in obtaining realistic and accurate results:

- *Robust image measurements:* To make the pose estimation match the input image we have developed a couple of innovative image likelihood functions which allow to track the subject even if the camera is moving. The first technique involves generating a synthetic image to be matched with the input one, while the second makes use of Generalized Expectation Maximization to find correspondences between image edges and body limbs.

- *Exploiting the dependence between pose and global motion:* Estimating the 3 dimensional trajectory of the body in a global coordinate system is also a very complex task starting from a monocular input. The traditional ways of tackling it consist either in learning the body root's motion from the training data or in using a zero acceleration model to obtain a smooth estimate. We have introduced another kind of prior, which directly relates the facing direction of the person and his motion, which has proved to make the tracking results both accurate and very realistic.

By taking advantage of all this we contributed to advance the state of the art: As a proof it is easy to assert that usual results on single-view 3–D people tracking include simple activities like walking or running. Instead the framework we have developed can be extended to motions which are much harder to track like golfing, skating and skiing.

1.3 Thesis Structure

The organization of this manuscript will be as follows: In Chapter 2 an extensive review of the state of the art will be presented. Then Chapter 3 will describe the model we used to represent the human body, explaining the different needed parameterizations. In Chapter 4 we will introduce the first approach we have developed, which involves studying how to take advantage of both the detection and tracking paradigms. Then in Chapter 5 we will describe a second approach, which consists in generating a subspace which has as dimensions image measurements and then using it for the tracking phase. Finally Chapter 6 will enumerate conclusions and possible extensions of this work.

Chapter 2

State of the Art

Nowadays, several ways exist to perform motion capture. Most commercially available techniques rely on inertial, electro-magnetic, acoustic or optical and infra-red devices, while video-based motion capture, which is the main subject of this thesis, is not considered to be solved yet. However, as also discussed in the previous chapter, the latter has lots of advantages compared to the former techniques: First of all it is orders of magnitude cheaper, since it does not require any special hardware apart from a standard video camera. Moreover, there is no need for the subject to wear special suits, nor to be in ad-hoc engineered environments and therefore the behavior can be much more natural. Finally video-based techniques could be applied also to pre-existing videos. For such reasons we will first briefly discuss commercial approaches and then turn to the study of video-based algorithms.

2.1 Commercial Motion Capture Devices

The motion capture devices that are currently available on the marked can be divided into optical ones, which are the most commonly adopted, inertial and electromagnetic.

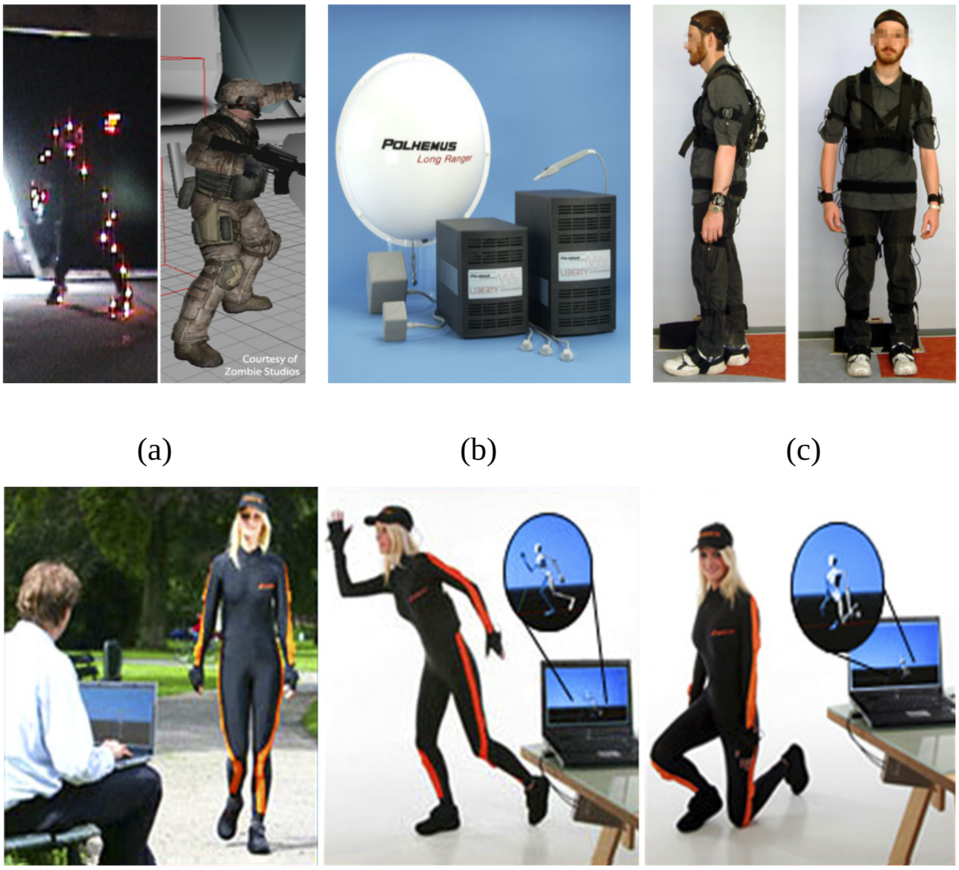
- Optical motion capture systems can be of two types, either reflective or LED based. They consist of a special suit to be worn by a subject. On this suit a set of small ball-shaped markers is placed, usually in correspondence of the body joints. The 3D position of the markers is captured with the help of dedicated infrared cameras which also emit light, in the case of reflective markers, or simply collect the light emitted by the LEDs, in the active markers case. Then the body motion is recovered with the help of a semi-automatic piece of software, which fits a kinematic skeleton to the marker measurements. For this reason the tracking results are not available in real-time but need some post-processing. The number of cameras composing such systems can vary a lot, starting from 3 to 64 and even more, and is directly proportional both to the quality of the obtained results (mostly due to the fewer occlusions) and to the global cost of the system. Examples of such systems are the Vicon [h] (Fig. 1.3 (a)), the Motion Analysis [c] systems and the Impulse system by PhaseSpace [e] (Fig. 2.1 (a)).

- Inertial systems are usually based on a suit, called exo-skeleton, onto which miniature gyroscopes are placed, at different body locations. The data collected by such inertial sensors are then sent to a computer which processes them in real-time and recovers the full body motion. The advantages with respect to optical motion capture systems are that they do not suffer from occlusions, can cover larger spaces, are more portable and less expensive. On the other hand they are a bit less accurate and can suffer from drift of the estimates. Examples of inertial system are the Meta Motion Gypsy system [b] (Fig. 1.3 (b)) and the Xsens MVN system [d] (Fig. 2.1 (d)).

- Electro-Magnetic systems consist of an array of receivers that measure their spatial relationship to a nearby transmitter. They are placed on the subject's body and are all connected to an electronic control unit. The transmitter generates a low-frequency electromagnetic field which is detected by the receivers and input to an electronic control unit, where it is filtered and amplified, and then finally sent to a central computer where the software resolves each sensor's position and orientation. Such systems work in near real-time and are less expensive than optical ones, but can encounter problems in presence of different types of metal, thus making the output data quite noisy. They also allow to track different subject simultaneously, but also in this case some interferences are possible. An additional issue is that the performers are constrained by cables in most cases. Examples of electro-magnetic systems are the Polhemus Liberty [g] (Fig. 2.1 (b)) and the MotionStar Ascension [a] (Fig. 2.1 (c)).

2.2 Video-Based Motion Capture

Existing approaches to video-based 3–D motion capture remain fairly brittle for many reasons: Humans have a complex articulated geometry overlaid with deformable



(a) (b) (c)

(d)

Figure 2.1: Commercial Motion Capture systems. **(a)** PhaseSpace Impulse System, which is also based on reflective markers. An example of reconstructed and animated pose is shown. **(b)** Polhemus Liberty System, based on an electromagnetic field. It is capable to track sensors' in 6 degrees of freedom at a 240Hz rate. **(c)** MotionStar Ascension System. It is a wearable system but as can be noticed from the image it is quite cumbersome for the subject. **(d)** Xsens MVN is a wearable inertial based motion capture tracker. Small gyroscopes are embedded in the suit, to allow an accurate motion reconstruction of the subject.

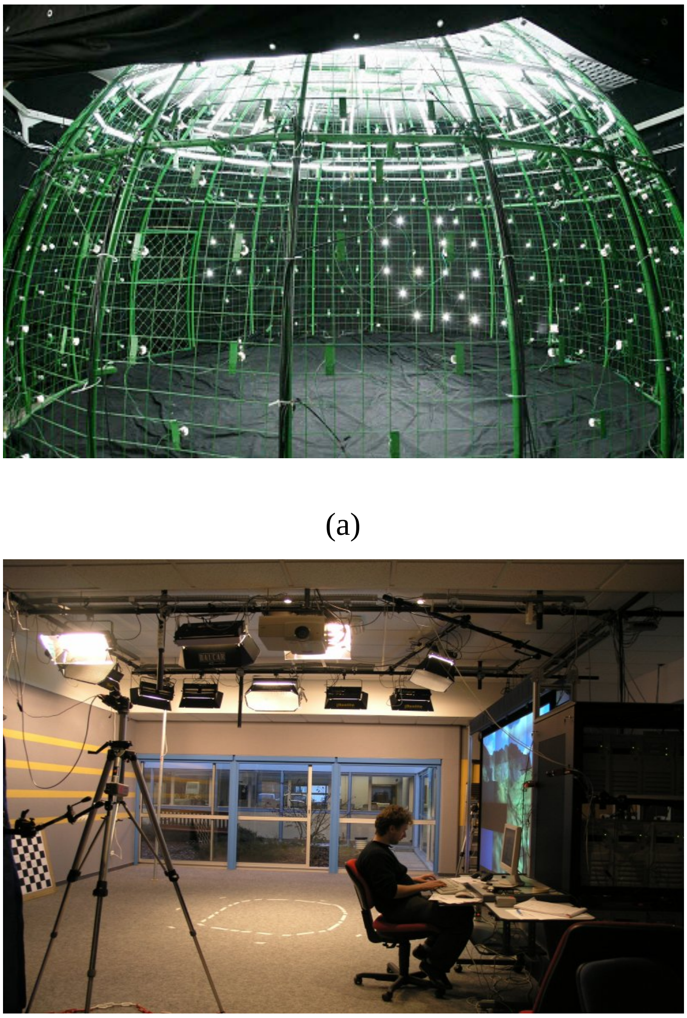
tissues, skin, and loose clothing. Their motion is often rapid, complex, self-occluding and presents joint reflection ambiguities. Furthermore, the 3–D body pose is only partially recoverable from its projection in one single image, where usually the background is cluttered and the resolution is poor. Reliable and robust 3–D motion analysis therefore requires good tracking across frames, which is difficult because of the poor quality of image-data and frequent occlusions. Recent approaches to handling these problems can roughly be classified into those that given a single input frame estimate the subject's pose (i.e. discriminative approaches) and those that track the subject in time during multiple consecutive frames, which in such hard conditions require a strong prior. In the remainder of this chapter, after a brief analysis of multi-view techniques that we give only for completeness' sake since in this thesis we only study monocular input, there will be a description of some detection algorithms, followed by an analysis of pure tracking approaches. Finally a comparison of the algorithms proposed in this thesis with other hybrid ones will conclude the chapter. Of course we plan to cover only the works that are more related to this thesis since the literature in the people detection and tracking field is too large to be totally covered. We refer to [Moeslund 06] for an extensive survey.

2.2.1 Multi-View Human Pose Estimation

Multi-camera techniques usually require quite expensive engineered studios, as the ones shown in Fig 2.2, to work correctly. In general, they can be classified as shape-based and motion-based. The so-called shape-based methods make use of 2–D shape cues like silhouettes and edge [Starck 03, Kakadiaris 00, Delamarre 99, Moeslund 00] or 3–D shape cues like voxels [Chu 03, Mikic 03, Muendermann 07, Sundaresan 08]. Usually the voxel representation of a person gives useful information about the 3–D shape and is often used in pose retrieval. Moreover these methods can be also 3–D model-based, as for example [Starck 03]. Instead the motion-based techniques [Yamamoto 98, Bregler 98] use information such as optical flow to track the pose. Typically these methods estimate pose variation and assume that the initial pose is available, while shape-based ones can be used both to initialize the pose and to perform tracking.

We first describe some approaches where an initial pose is not required: In [Mikic 03, Muendermann 07] all the steps for pose retrieval are performed using voxel based techniques, while in [Chu 03] volume data is used to acquire and track a human body model. Instead in [Cheung 03] the body kinematics are estimated using shape from silhouettes, with the drawback that the subject is asked to move one joint at a time to initialize the pose.

Other techniques instead, as mentioned above, require an initial pose estimate: Motion-based approaches like [Yamamoto 98, Bregler 98] make use of optical flow, while shape-based ones make use of different cues. The technique presented in [Gavrila 96] for example uses a generate-and-test algorithm in which the pose search is performed in a parameter space, and then the pose is matched to the input using a variant of Chamfer matching. In [Kakadiaris 00] silhouettes from multiple cameras are used to estimate the 3–D motion, similarly to what is done in [Delamarre 99], where the authors use silhouettes in conjunction to 3–D articulated models: Forces are applied to the contours obtained from the model projection so that they move towards the silhouette contours. Also in [Moeslund 00, Starck 03] a 3–D model is animated and fitted to the input using



(a)

(b)

Figure 2.2: Multi-camera settings in engineered studios. (a) The Multi-Camera and Multi-Lighting Dome at the Tsinghua University of Beijing, China. (b) The Grimage system at the INRIA-Rhone Alpes in Grenoble, France.

correspondences from silhouette, stereo and feature cues. Finally in [Sigal 03, Sigal 04] nonparametric belief propagation is used to track motion in a multicamera setup.

To summarize the pros and cons of such methods, we can argue that motion-based trackers suffer from the problem of drift, i.e. they estimate the change in pose from frame to frame and as a result the error accumulates over time. On the other hand, shape-based methods rely on absolute cues and do not face the drift problem but shape cues are very likely non extractable in every single input frame. They typically attempt to minimize an objective function (which measures the error in the pose) and are prone to converge to incorrect local minima. Specifically, background subtraction or voxel reconstruction errors in voxel-based methods result in cases where body segments are missing or adjacent body segments are merged into one.

2.2.2 People Detection

A standard detection approach implies recognizing postures from a single image by matching it against a database and has become increasingly popular recently: In [Thayananthan 03] the authors propose a Chamfer matching technique to match shapes, and in [Dimitrijevic 06] a similar approach is applied to human silhouettes. They show that such a technique is robust to clutter, but nonetheless requires the use of multiple templates to handle pose, scale and viewpoint changes in the subject. Also in [Howe 04] Chamfer distance is used, but in this case to make a direct silhouette lookup to select candidate poses from a training database.

The authors of [Agarwal 04] instead directly learn a mapping between shape descriptors, extracted from silhouettes, and pose, using nonlinear regression. Their algorithm is also view-independent since the mapping is learnt from silhouettes obtained by projecting 3–D models onto several viewing planes. A similar idea was earlier developed in [Rosales 01], in which the mapping from visual features of a segmented person to a static pose is learnt using neural networks, in a framework which is invariant to speed and direction of movement. Also in [Shakhnarovich 03] a mapping from input silhouettes to estimated pose is studied, with the slight difference that this mapping is purely exemplar-based and makes use of efficient hashing functions to achieve a faster computation. Recently more sophisticated techniques using Bayesian Mixture of Experts [Sminchisescu 05, Bo 08] and mixture of Gaussian Process [Urtasun 08a] made this mapping more powerful, by increasing the number of examples in the training set and decreasing the computation time by orders of magnitude.

A related idea is proposed in [Elgammal 04] where the authors learn two mappings, one from the visual input, which is again a silhouette, to an activity dependent manifold, and then another mapping from such manifold to the 3–D body pose.

In [Mori 04] a different idea is presented, in which body parts detectors are combined in a bottom-up fashion to retrieve a 2D representation of the body pose, with the advantage of using as input a standard image and not a clean silhouette. This idea originarily comes from [Forsyth 97], where the authors first introduced the notion of *body plans* to represent people or animals as a structured assembly of parts learnt from images. A further evolution of such idea has been recently presented [L. Bourdev 09], in which the authors introduce the concept of *poselets,* which basically constitute parts that are tightly clustered in both appearance and configuration space, as shown in Fig. 2.3. Then

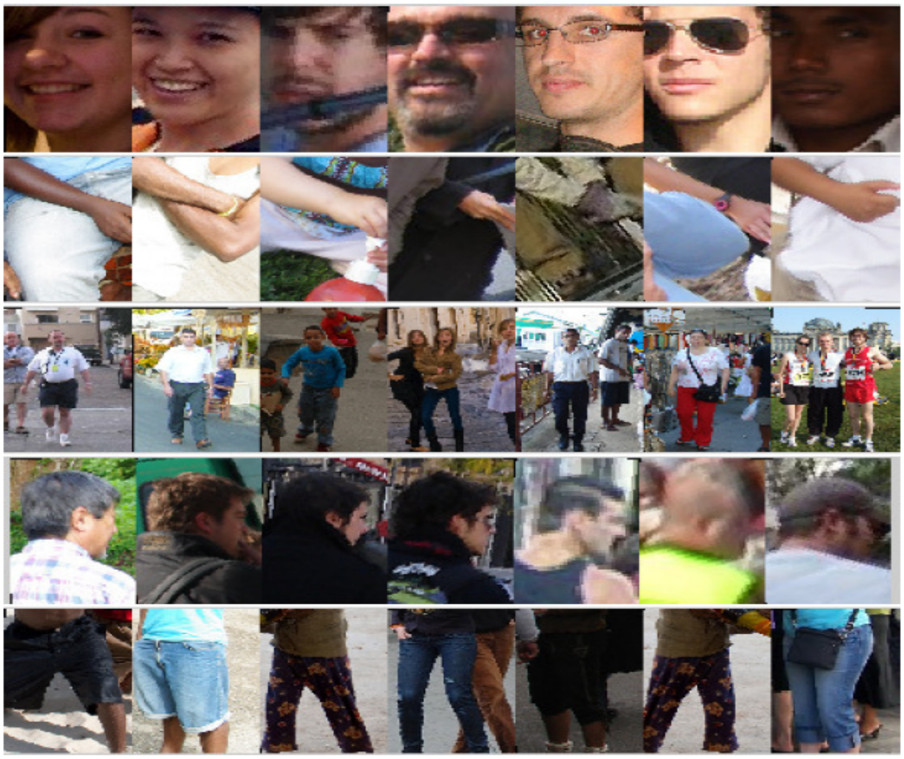


Figure 2.3: The poselets approach presented in [L. Bourdev 09]. The poselets correspond to parts that are clustered both in the appearance and configuration space. The figure shows examples of the Frontal Face, Right Arm crossing Torso, Pedestrian, Right Profile, and Shoulder and Legs Frontal View poselets.

an additional classification layer allows to use the poselets as a starting point to estimate the 3–D pose of the subject. A related bottom-up technique is the use of pictorial structures [Felzenszwalb 00], which has also been applied to the people detection and pose estimation problem [Andriluka 09].

On the other hand, in [Leibe 05] a higher level technique is presented, which can locate and track multiple pedestrians in very cluttered scenes, with the drawback of not being able to estimate their pose.

In general the use of such discriminative algorithms has several drawbacks: They usually require very large sets of examples to be effective, and in this context it was even proposed in [Enzweiler 08] to artificially increase the size of the training set through selective sampling of a generative model. Moreover they often rely on background subtraction and on clean silhouettes, such as those that can be extracted from the HumanEva dataset [Sigal 06], which require static cameras or controlled environments. Finally these methods are usually able just to obtain a good reconstruction of the body pose but cannot correctly locate people in a 3–D environment.

2.2.3 People Tracking

Contrary to detection, tracking involves predicting the pose in a frame given observation of the previous one. It requires thus an initial pose and can easily fail if errors start accumulating in the prediction, causing the estimation process to diverge. The possibility of drifting is usually mitigated by introducing sophisticated statistical techniques for a more effective search. For example [Deutscher 00] and [Davison 01] propose the use of an annealed particle filtering technique to effectively optimize the very high-dimensional objective function matching the body configuration to the input images. Similarly, in [Choo 01] an Hybrid Monte Carlo filter is used to obtain samples in such a high-dimensional space. Instead, in [Wu 03] the authors propose to divide the problem in subparts, which are separately analyzed using a Dynamic Markov Network, and then these smaller subproblems interact through a Mean Field Monte Carlo algorithm to solve the higher dimensional problem. Finally, the authors of [Sminchisescu 03a] observed that the direction of maximal uncertainty is where alternative poses with good correspondence to the input are most likely to be found. For this reason they introduced a technique named *covariance scaled sampling*, still based on particle filters, which increases the covariance in such direction to generate samples close to local minima in the objective function. In a successive work [Sminchisescu 03b] they also studied the causes of visual ambiguities, as potential kinematic minima, and incorporated them in the sampling process to make pose estimation more robust and efficient.

An alternative solution is to use strong dynamic motion models as priors: In [Sidenbladh 00] both the use of a constant velocity model and of a walking specific one are proposed. In [Ormoneit 01] a set of training motions is used to learn the body dynamics, using PCA. On the same direction also in [Sidenbladh 02] the same authors propose an algorithm to match the motion dynamics of the body in the scene to one of the training examples, which is then used to estimate the future motion. A very similar idea, whose output is shown in Fig. 2.4, is described in [Rosenhahn 07], where tracking is performed in a multi-camera environment with the help of a 3–D model of the subject, and tracked motion patterns are matched to training patterns to predict future poses. The authors of [Urtasun 06] propose instead the use of a Gaussian Process Dynamical Model (GPDM) to learn both a low dimensional embedding of the motion and a dynamical model inside such embedding. In [Taycher 06] the use of a Conditional Random Field model is proposed to make the estimation problem discrete and thus efficiently solvable. Furthermore in [Rosenhahn 09] the authors show how, also in a multi-camera setting, incorporating suitable motion priors can help in regularizing and stabilizing the tracking results. In this case the correct prior to apply is chosen after a first rough classification of the type of activity that the subject is performing. In recent years also the application of physics-based

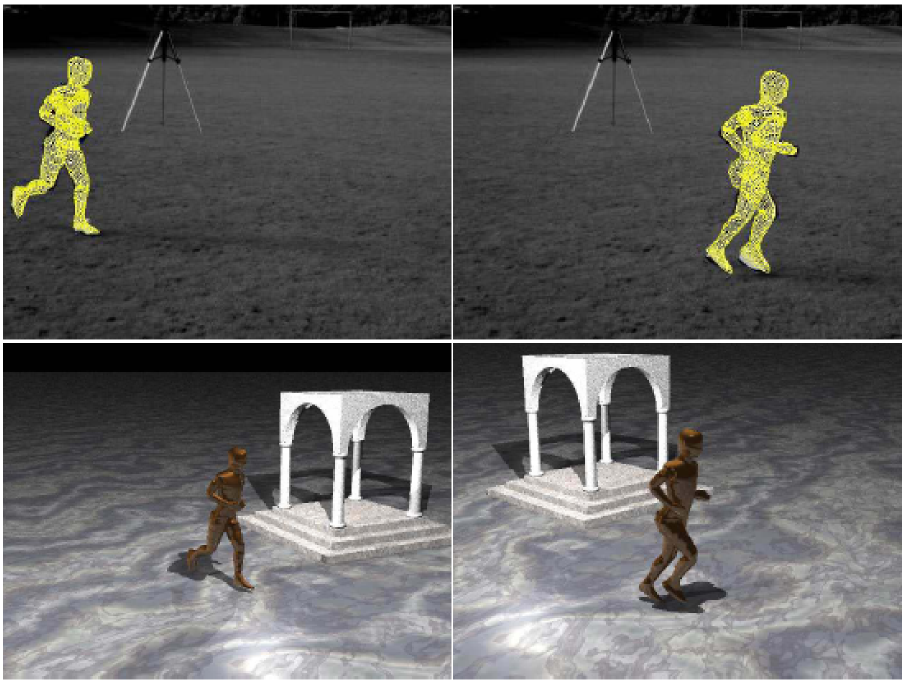


Figure 2.4: Tracking results of the approach presented in [Rosenhahn 07], where a 3–D model of the subject is available and tracked motion patterns are matched to training ones to estimate future poses.

models to tracking has shown good results [Brubaker 07, Brubaker 08, Vondrak 08]. The physics-based approaches are attractive for motions such as walking or running for which appropriate models have been developed. For others, assuming that motion-capture data can be obtained, the learning-based approaches are much easier to deploy. They all rely on the fact that the space of poses for a particular activity can be modeled as a low-dimensional manifold, embedded in the much higher-dimensional space of all possible poses of an articulated human body model. As a result, recovering sequences of body poses can be achieved by optimizing over the low-dimensional manifold rather than the high-dimensional pose space.

To this end, the manifold is usually parameterized by a few latent variables and the mapping between them and the poses, or pose sequences, can be either linear, as in [Sidenbladh 00, Ormoneit 01, Urtasun 04], or not [Elgammal 04, Sminchisescu 04, Urtasun 05, E.-J-Ong 06]. For example, in [Urtasun 05], a Gaussian Process Latent Variable Model (GPLVM) [Lawrence 04] was used to learn a differentiable manifold from modest amounts of training data, which allowed motion recovery by continuous optimization of an image-based objective function. There has been attempts at constraining the topology of the latent space to assume known configurations, such as circles, or to respect the distances in the high-dimensional space between neighboring examples [Urtasun 08b]. However, such techniques still require learning a latent space, which remains a complex optimization problem, whereas, in the algorithm proposed in Chapter 5, we propose to directly make use of directly observable image quantities as a latent space.

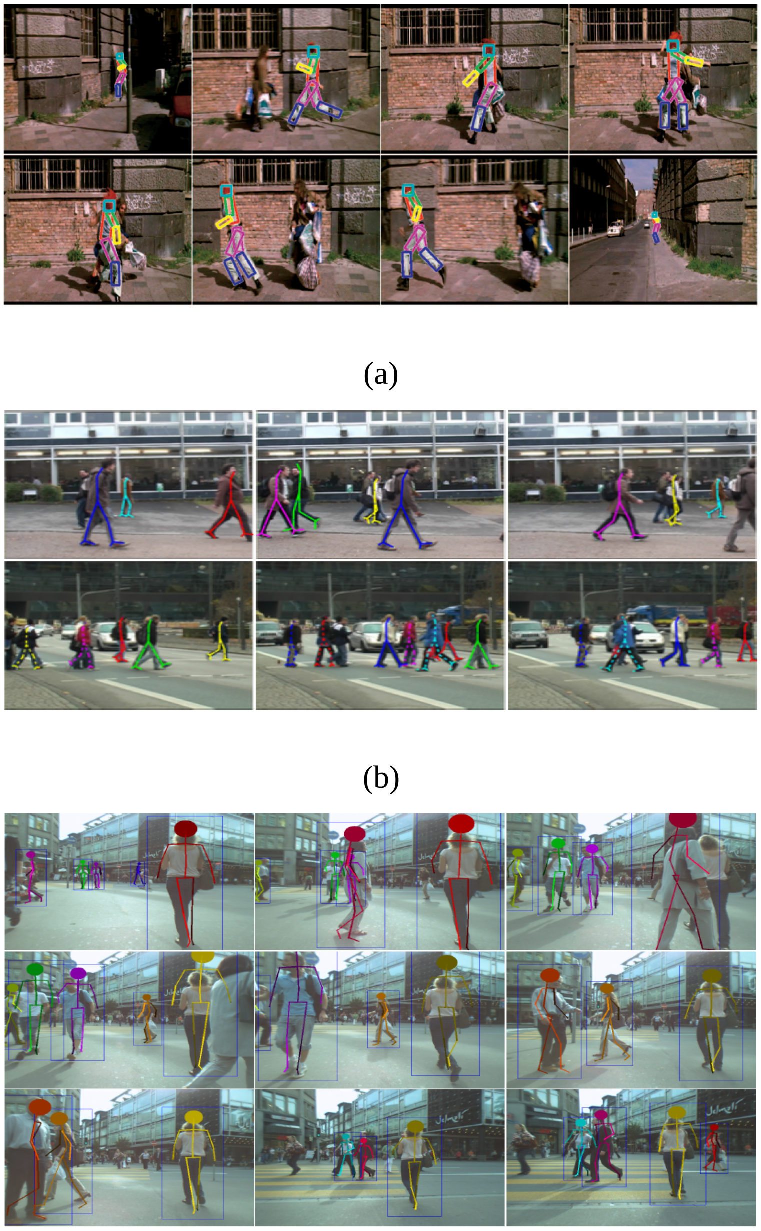
Another issue with such approaches is that usually the latent variables have no physical meaning and are hard to initialize from image data. In GPLVM approaches such as [Urtasun 05], the process is initiated by finding a training example that best fits the data and using the corresponding latent variable for initialization purposes. This implies a search that our technique avoids. This difficulty was addressed in [Navaratnam 07, Shon 06, Ham 06] by learning a common low-dimensional latent space both for pose and image data. However, because learning the joint latent space is more involved than learning individual latent spaces, standard techniques require more training data than is normally available. As a result, the authors of [Navaratnam 07] had to develop a more sophisticated algorithm able to use not only examples for which the correspondence between pose and image data is known, but also examples for which it is not. Unfortunately, learning a GPLVM is computationally expensive and sensitive to initialization of the latent variables. Furthermore, it yields a complex objective function with many local minima, which is not always ideal for inference purposes.

In the algorithm described in Chapter 5, by contrast, we rely on ordinary Gaussian Processes (GP) [Rasmussen 06] to establish a direct mapping between low-dimensional observable image data and the high dimensional pose space. Since fewer parameters need to be learned, training is more straightforward and requires far less data. Initialization is similarly easy since the image data directly give us a mapping to a pose sequence, which we then simply need to refine. As described in the previous section, GPs were also used in [Urtasun 08a] to map image data directly to 3–D poses. To model the multimodality of this mapping, they required using not a single GP but a mixture of them. In our case, due to our choice of image measurements, and because we map them to sequences of poses, the mapping is unimodal and can be modeled with a single GP.

2.2.4 Approaches combining Detection and Tracking

Neither detection nor tracking has yet been proved to be superior, and both are actively studied and sometimes combined: Manually introducing a few 3–D keyframes is known to be a powerful way to constrain 3–D tracking algorithms [DiFranco 01, Loy 04]. In the 2–D case, it has recently been shown that this can be done in a fully automated fashion to track multiple people in extremely long sequences [Ramanan 06]. This involves tracking forwards and backwards from individual and automatically detected canonical poses, and the results of such algorithm are shown in Fig. 2.5 (a). While effective, this approach to tracking still has the potential to diverge. In the approach described in Chapter 4 of this thesis, we avoid this problem and go to full 3–D by observing that automated canonical pose detections can be linked into complete trajectories, which let us first recover rough 3–D poses by interpolating between these detections and then refining them by using a generative model over full sequences. A similar approach has been proposed for 3–D hand tracking [Tomasi 03] but makes much stronger assumptions than we do by requiring high-quality images so that the hand outlines can accurately and reliably be extracted from the background.

More recently a similar technique has been developed also by other researchers: Andriluka et al. [Andriluka 08] start from the idea presented in our first paper [Fossati 07] and described in Chapter 4 of this thesis. They propose an algorithm that, mixing detection and tracking, is able to cope with multiple pedestrians and large occlusions, but which only gives 2–D results, with a very rough pose estimation that does not include the arms' positions, as visible in Fig. 2.5 (b). Another drawback of such technique is that it only works if the viewpoint is static and from the side of the subjects, with a small tolerance. A further similar idea has been presented in [Gammeter 08], where the authors propose a framework that uses as input a narrow-baseline stereo stream, captured with the help of a special moving rack, therefore with much richer input cues than the ones we adopt. It first detects and tracks in 3–D bounding boxes around pedestrians, using a previously published technique by the same authors [Ess 08], and then estimates the body articulation of such pedestrians by learning a mapping between silhouettes and poses with the help of Gaussian Processes, assuming that subjects are continously walking. The results of such technique are presented in Fig. 2.5 (c).



(a)

(b)

(c)

Figure 2.5: Body Pose Estimation combining Detection and Tracking. (a) Tracking algorithm proposed in [Ramanan 06]. It tracks the subject in 2–D in very long sequences. (b) The approach presented in [Andriluka 08] can handle multiple pedestrians and large occlusions but the pose estimation is quite rough. (c) Pose estimation results obtained with the algorithm presented in [Gammeter 08], which requires a narrow-baseline stereo stream as input.

Finally it must be stated that the work presented in Chapter 4 of this thesis builds on some earlier results. We rely on spatio-temporal templates to detect the people in canonical poses [Dimitrijevic 06] and on PCA-based motion models [Urtasun 06] to perform the interpolation. However, unlike in this latter paper, our system does not require manual initialization. This means that we had to develop a strategy to link detections, infer initial 3–D poses from them, and perform the pose refinement even when the camera moves or the background is cluttered. As a result, we can now operate fully automatically under far more challenging conditions than before.

Chapter 3

Human Body and Motion Parameterization

3.1 Human Body Model

We represent the human body as cylinders attached to an articulated 3–D skeleton as shown in Figure 3.1. Our body model has standard dimensions and proportions, thus it allows us to obtain reasonable tracking results on different subjects without the need of being specifically trimmed. In this case using a model that is slightly too small or too big simply results in variations in the recovered camera position with respect to the subject, due to the scale ambiguity inherent to monocular reconstruction.

Adapting the skeleton proportions would have required an a priori knowledge of their more likely variations, as was done for example in [Balan 08] using the SCAPE model [Anguelov 05] depicted in Fig. 3.2, which required a few thousands of laser scans of real people to be built. Basically it allows to control the the body shape, along the

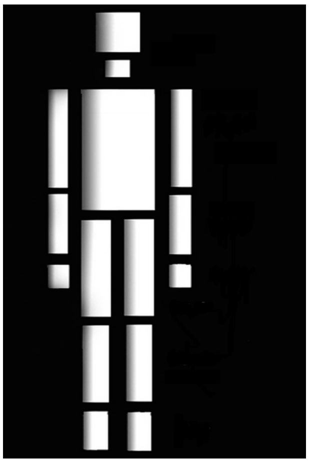


Figure 3.1: Cylinder-based representation used as human body model in our experiments.

directions of maximal inter-person variance, by tuning few PCA parameters.

Other complex models that have been proposed and used in literature are for example superquadrics (introduced in [Barr 84] and used in the body tracking context in [Sminchisescu 03b]), shown in Fig. 3.3 (a), implicit surfaces [Dewaele 04, Herda 04, Plänkers 03], depicted in Fig. 3.3 (b), and also directly 3–D meshes [Carranza 03, Starck 03, Starck 05, Franco 05, Gall 10], shown in Fig. 3.3 (c) and (d). All such models are definitely more accurate than the simple cylinder-based one, but on the other hand have several drawbacks in the cases we take into consideration: First of all a model of the tracked subject would be needed, thus requiring bulding a new one for each different subject, while our model is general and has been used for all the subjects shown in our experiments. Secondly, being more complex, they are also computationally heavier in case of rendering. Since all the generative techniques developed in this thesis require a projection of the body model onto the images to evaluate the objective function, and since several of such evaluation need to be performed in our stochastic optimization frame-

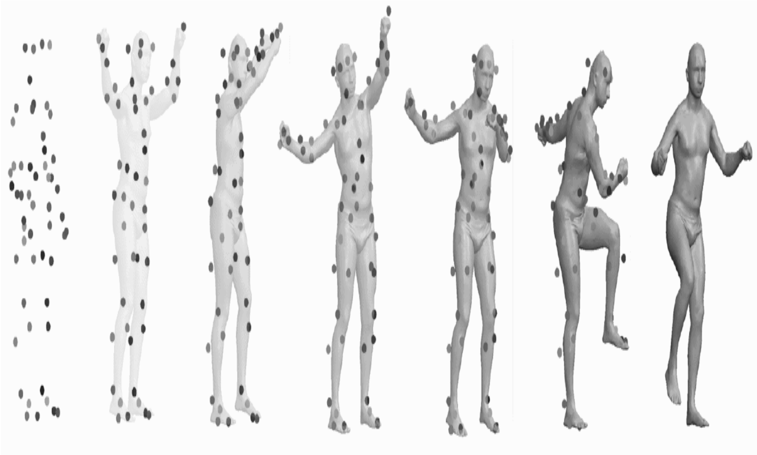
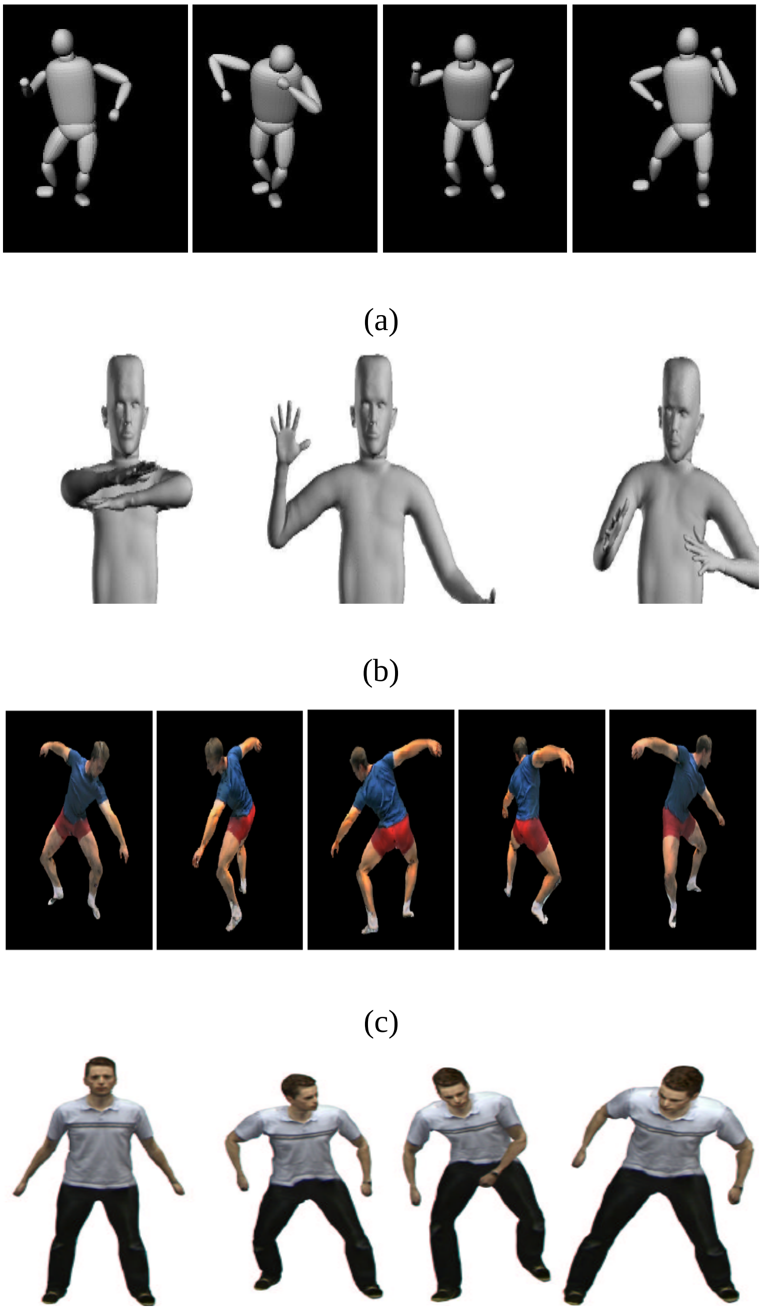


Figure 3.2: The SCAPE model has been developed by Stanford University and Intel through laser scan on several thousands of real people [Anguelov 05].

work, a simple model allows very large savings in terms of computation time. Finally and additional point is that accurate models require a very high quality input (i.e. multi-camera settings, usually in very well engineered studios) to fit the images. In cases such as ours, where the video stream is monocular, the resolution not too high and other possible shortcomings like cluttered backgrounds and occlusions usually occur, the robust tracking framework we propose is able to produce reliable results even with the simple cyilinder-based model we adopt.

3.2 Pose and Motion Definition

A *pose*, whether canonical or not, is given by the position and orientation of the body root node, defined at the sacroiliac, and a set of joint angles. More formally, let D denote the number of joint angles in the skeletal model (usually D was fixed to 51 in our experiments, as better depicted in Figure 3.4). A pose at time t is then given by a vector of



(a)

(b)

(c)

(d)

Figure 3.3: (a) Superquadrics models used in [Sminchisescu 03b]. (b) Implicit surface reconstruction of the human body, proposed in [Plänkers 03]. (c) 3D mesh representing the human body reconstructed in a multi-camera setup, as proposed in [Carranza 03]. (d) Another 3D mesh reconstruction of the human body from multiple video streams, proposed in [Starck 03].

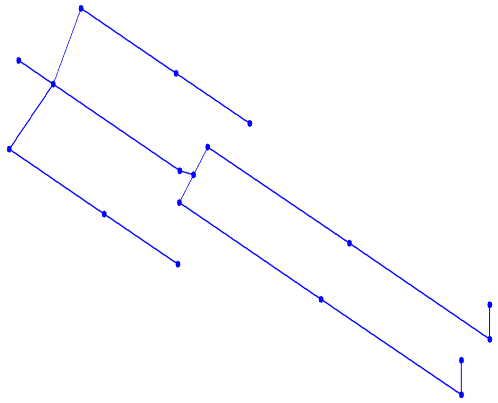


Figure 3.4: Skeleton representing the joint angles used in our body pose estimation algorithms. The markers correspond to the body joints, and the configuration of each one of the joints is described by 3 Euler angles.

joint angles, denoted $\psi_t = [\theta_1, \cdots, \theta_D]^T$, along with the global position and orientation of the root

$$\mathbf{g}_t \in \mathbb{R}^6 \quad . \tag{3.1}$$

A *motion* between two canonical poses can be viewed as a time-varying pose. While pose changes continuously with time, we assume a discrete representation in which pose is sampled at N distinct time instants. In this way, a motion becomes a sequence of N discrete poses

$$\begin{aligned} \Psi &= [\psi_1^T, \cdots, \psi_N^T]^T \in \mathbb{R}^{DN} \quad , \\ \mathbf{G} &= [\mathbf{g}_1^T, \cdots, \mathbf{g}_N^T]^T \in \mathbb{R}^{6N} \quad . \end{aligned} \tag{3.2}$$

Since motions can occur at different speeds, we encode them at a canonical speed and time-warp them to represent other speeds. We let the pose vary as a function of a phase parameter μ that is defined to be 0 at the beginning of the motion and 1 at the end. For periodic motions such as walking, the phase is periodic. For generic ones such as swinging a golf club, it is not. The canonical motion is then represented with a sequence of N poses, indexed by the phase of the motion. For frame $n \in [1, \, N]$, the discrete phase $\mu_n \in [0, 1]$ is simply

$$\mu_n \;=\; \frac{n-1}{N-1}\,. \tag{3.3}$$

3.3 PCA on Poses

A standard approach to human body tracking is based on retrieving the body pose independently in each frame. One of the main issues about this idea is that there are too many variables that need to be optimized at the same time, namely the D-dimensional vector ψ of joint angles and the 6-dimensional vector $\mathbf{g}_t$ describing position and orientation of the body root. In practice, we learn motion models from optical motion capture data including different subjects performing the same activity several times. For walking, we used a Vicon^{tm} system to capture the motions of four men and four women on a treadmill at speeds ranging from 3 to 7 km/h by increments of 0.5 km/h. The body model had $D = 51$ degrees of freedom. Four cycles of walking at each speed were used to capture the natural variability of motion from one gait cycle to the next for each person. Similarly, to learn the golf swing model, we asked two golfers to perform 24 swings each. Therefore, considering the fact that there is a good prior concerning the motion model to be used, applying Principal Component Analysis to our training data to reduce

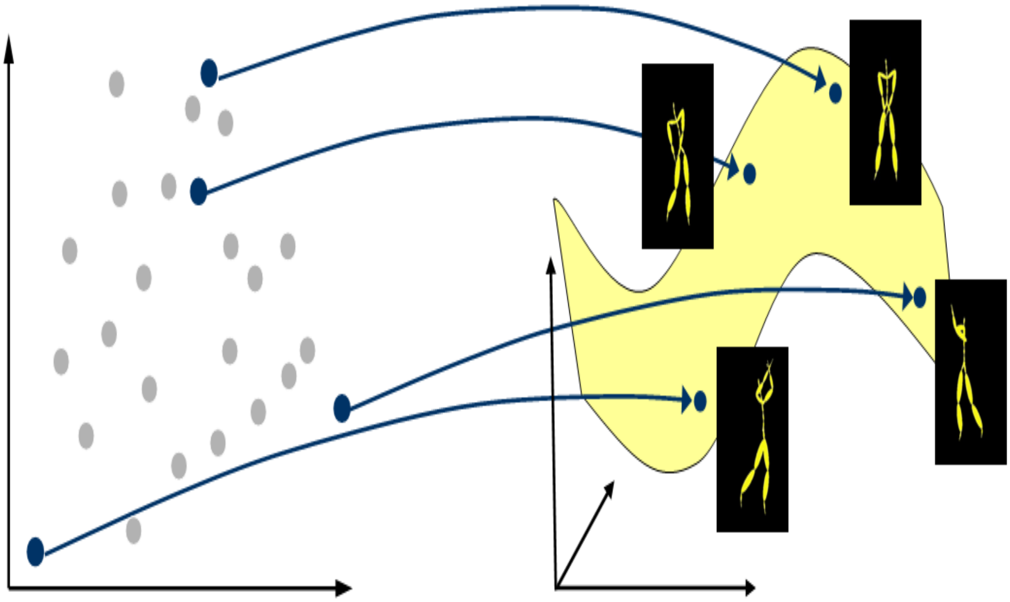


Figure 3.5: Low-dimensional embedding of the pose space. Each point in the latent PCA space, on the left, corresponds to a body pose in the high dimensional body configuration space, on the right.

the dimensionality of the problem, as shown in in Figure 3.5, results to be the best option. With such procedure all the poses in our training set can be approximated through a linear combination of the mean pose ψ_0 and a set of *eigen-poses* $\{\psi_i\}_{i=1}^{d}$:

$$\psi \;\approx\; \psi_0 + \sum_{i=1}^{d} \beta_i \psi_i \quad . \tag{3.4}$$

The scalar coefficients, $\{\beta_i\}$, characterize the pose, and $d \leq D$ controls the fraction of the total variance of the training data that is captured by the subspace.

3.4 PCA on Sequences

Optimizing the pose in each frame independently leads to quite accurate results if considered frame-wise, but the output sequences may suffer from jittering. For this reason an alternative approach is to optimize the objective function over the full sequence at one time, be it a walking cycle or a golf swing, and to make this manageable a different low dimensional embedding of our training data needs to be adopted. Given a training set of motions, denoted $\{\Psi_j\}$, where each motion is composed by N consecutive poses as explained in Section 3.2, we use Principal Component Analysis to find a low-dimensional basis with which we can effectively model them, as visually explained by Figure 3.6. In particular, the model approximates motions in the training set with a linear combination of the mean motion Θ_0 and a set of *eigen-motions* $\{\Theta_i\}_{i=1}^{m}$:

$$\Psi \;\approx\; \Theta_0 + \sum_{i=1}^{m} \alpha_i \Theta_i \quad . \tag{3.5}$$

The scalar coefficients, $\{\alpha_i\}$, characterize the motion, and $m \leq D\,N$ controls the fraction of the total variance of the training data that is captured by the subspace.

A pose can then be defined as a function of the scalar coefficients, $\{\alpha_i\}$, and a phase value, μ. We therefore write

$$\psi(\mu, \alpha_1, \cdots, \alpha_m) \;\approx\; \Theta_0(\mu) + \sum_{i=1}^{m} \alpha_i \Theta_i(\mu) \quad . \tag{3.6}$$

Note that now $\Theta_i(\mu)$ are *eigen-poses*, and $\Theta_0(\mu)$ is the mean pose for that particular phase.

A linear subspace representation like PCA has the advantages of being very simple and very fast to compute. On the other hand in the literature, as explained in Chapter 2,

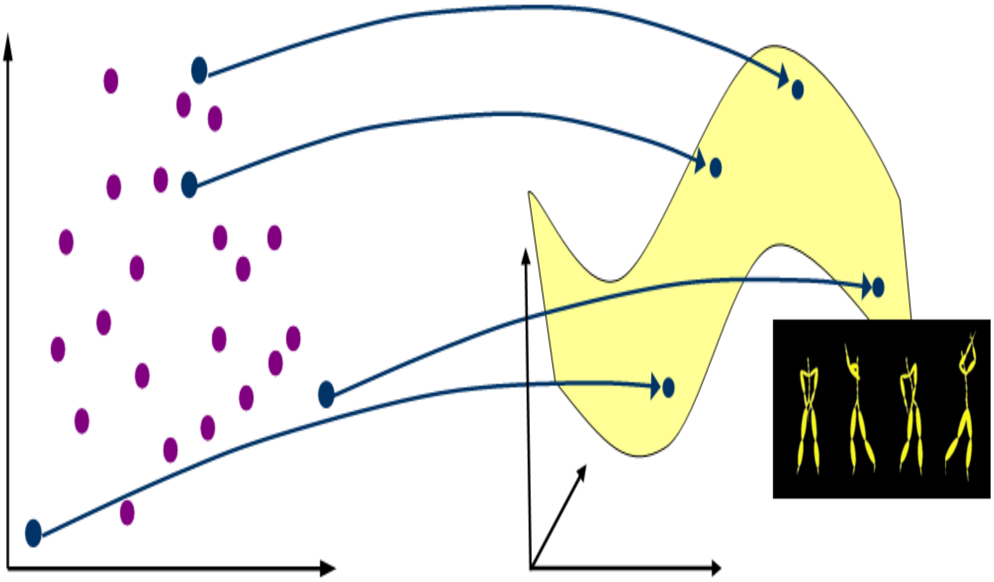


Figure 3.6: Low-dimensional embedding of the motion space. Each point in the latent PCA space, on the left, corresponds to sequence of poses in the high dimensional motion space, on the right.

also more complex and non-linear embeddings, like Kernel PCA, Local Linear Embedding, or Gaussian Process Latent Variable Models, have been successfully adopted in the Human Body Tracking context. We still believe that to model relatively simple activities like walking or golfing a linear subspace is a good compromise between complexity and model elasticity, and this will be proved in the next chapters.

Chapter 4

Bridging the Gap between Detection and Tracking

We combine detection and tracking techniques to achieve robust 3–D motion recovery of people seen from arbitrary viewpoints by a single and potentially moving camera. To do this we detect key postures, which can be done reliably, using a motion model to infer 3–D poses between consecutive detections, and finally refining them over the whole sequence using two different techniques.

We demonstrate our approach in the cases of golf motions filmed using a static camera and walking motions acquired using a potentially moving one. We will show that our approach, although monocular, is both metrically accurate because it integrates information over many frames and robust because it can recover from a few misdetections.

4.1 Approach

We first use a template-based algorithm [Dimitrijevic 06] to detect people in poses that are most characteristic of the target activity, as shown in the first row of Fig. 4.1. The templates consist of consecutive 2–D silhouettes obtained from 3–D motion capture data seen from six different camera views and at different scales. This way the motion information is incorporated into the templates and helps to distinguish actual people who move in a predictable way from static objects whose outlines roughly resemble those of humans. For each detection, the system returns a corresponding 3–D pose estimate.

In theory, a person should be detected every time a key pose is attained, which the template-based algorithm does very reliably. The few false positives tend to correspond to actual people but detected at somewhat inaccurate scales or orientations and false negatives occur when the relative position of the person with respect to the camera generates an ambiguous projection and the key pose becomes hard to distinguish from others. In our experiments, this almost never happened in the golfing case and sometimes did in the walking case when the camera moved and saw the subject against a cluttered background from a difficult angle. To handle such cases, we have implemented a Viterbi-style algorithm that links detections into consistent trajectories, even though a few may have been missed. Since the camera may move, we perform this computation in the ground plane, which we relate to the image plane via a homography that is automatically recomputed from frame to frame, after a first manual initialization.

Finally, we use consecutive detections to select and time-warp motions from a training database obtained via optical motion capture. As shown in the second row of Fig. 4.1,

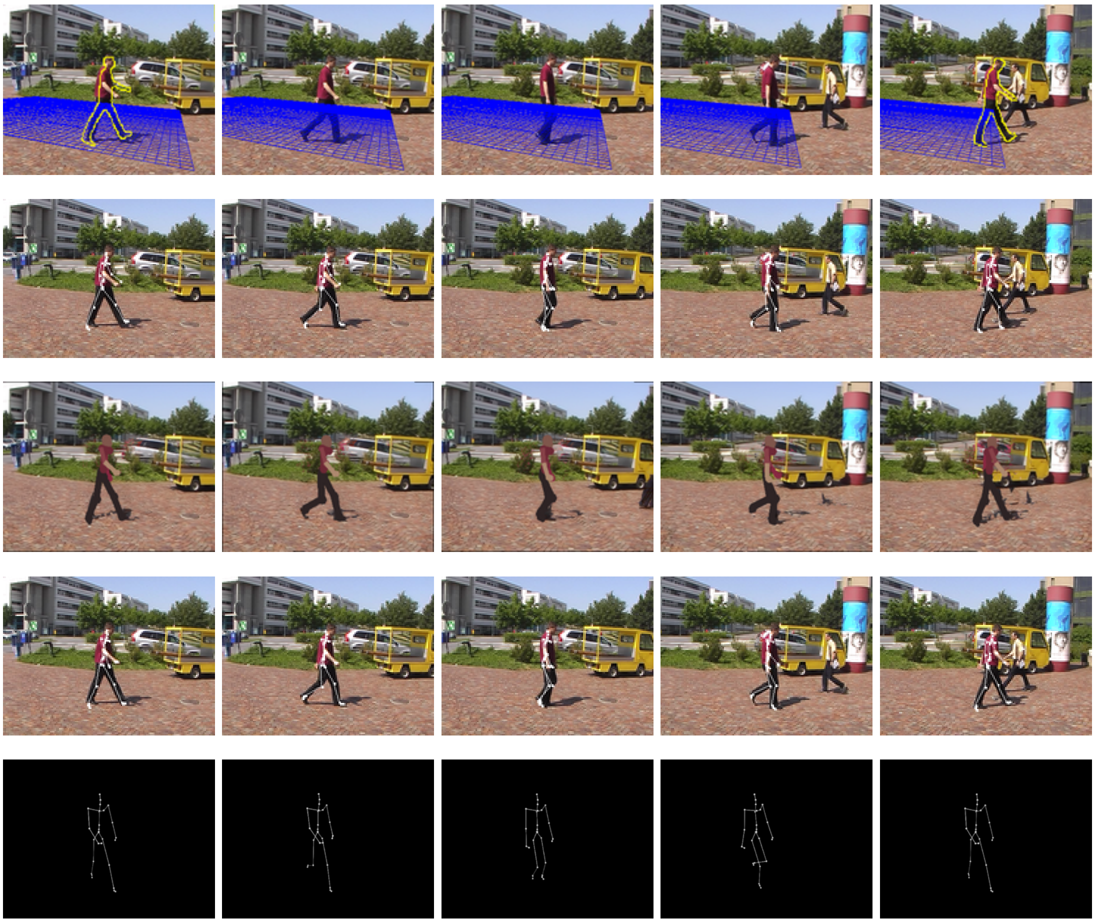


Figure 4.1: Our approach. **First row:** Input sequence acquired using a moving camera with silhouettes detected at the beginning and the end of the walking cycle. The projection of the ground plane is overlaid as a blue grid. **Second row:** Projections of the 3–D poses inferred from the two detections. **Third row:** Synthesized images that are most similar to the input. **Fourth row:** Projections of the refined 3–D poses. **Fifth row:** 3–D poses seen from a different viewpoint.

this gives us a rough estimate of the body's position and configuration in each frame between detections. To refine this initial estimate, and since the camera may move from frame to frame, we first compute homographies between consecutive frames and use them to synthesize a background image from which the moving person has been almost completely removed. When we know the camera to be static, we synthesize the background image by simple median filtering of the images between detections. At this point we propose two different image likelihood functions. In the first case we learn an appearance model from the detections and use it in conjunction with the synthesized background to refine the body position by minimizing an objective function that represents the likelihood of the observed image under the current model. In the second case, we adopt Generalized Expectation Maximization to match the body limbs to the image edges. Using the appearance based approach we obtain the refined poses depicted by the bottom three rows of Fig. 4.1.

In the remainder of this chapter, we first briefly describe our approach to detecting people in canonical poses, second to using these detections to estimate the motion between frames, and, finally, to refining this estimate.

4.2 Key-Pose Detection

To reconstruct golf swings, we treat as canonical poses the transition between the upswing and the downswing and the end of the upswing, as shown in Fig. 4.2. Since there is a little motion of the golfer's center of gravity during the swing, we take the $\mathbf{g}_t$ vectors of Eq. 3.1 to be initially all equal, and during the optimization we only allow a rotation

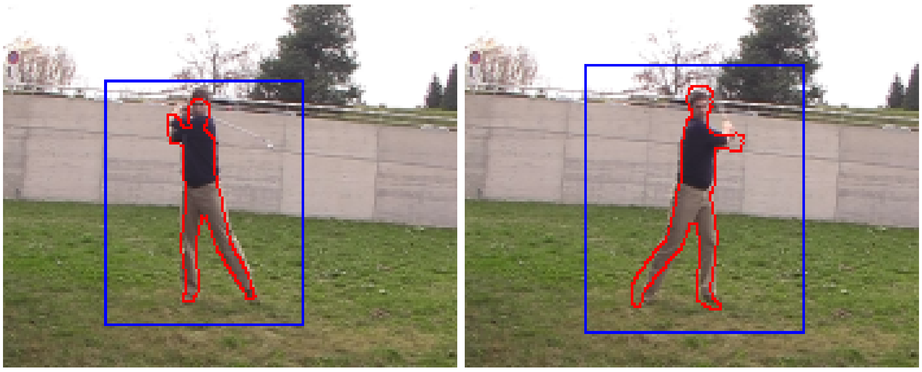
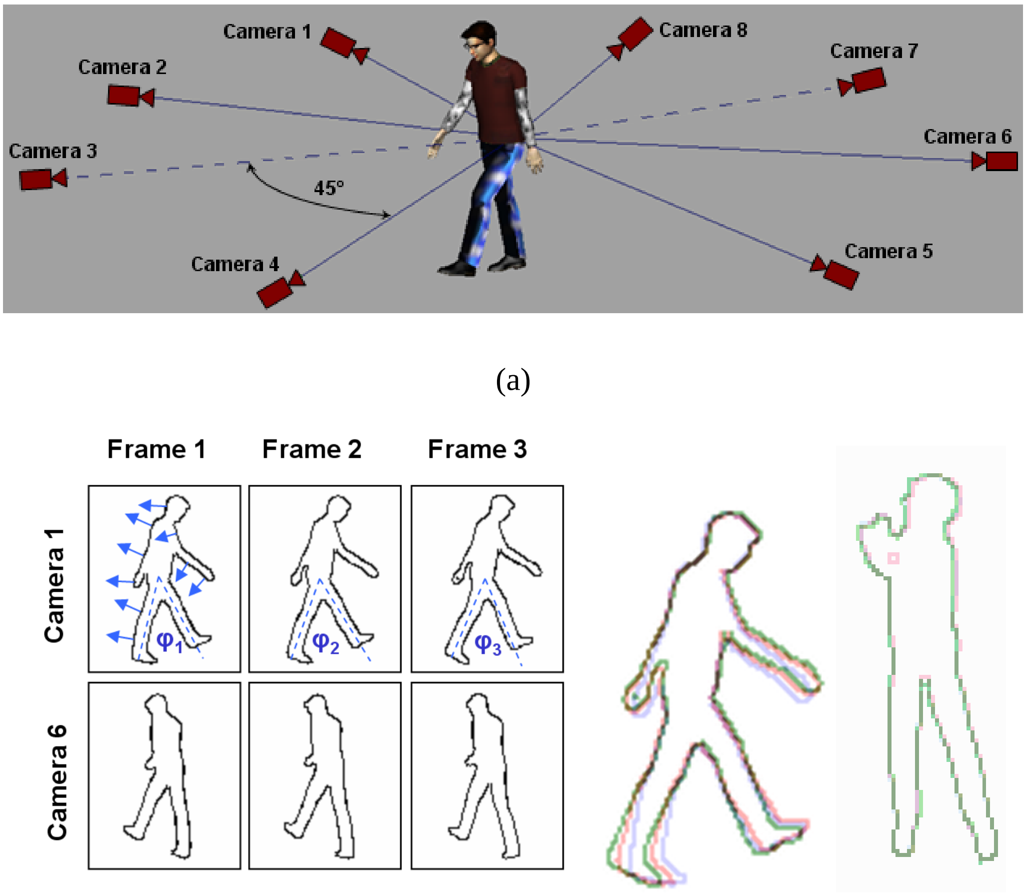

(a) (b)

Figure 4.2: Key pose detections at the beginning and at the end of a golf swing.

around the z axis. Furthermore, since the camera is static in the examples we use, we simply median filter frames in the whole sequence to synthesize the background image we need to run the pose refinement algorithm of Section 4.5.

To track walking people, we use the beginning of the walking cycle, when the legs are furthest apart, as our canonical pose. Our spatio-temporal templates detect this pose reliably but with the occasional false positive and false negative. Such errors must be eliminated and the valid detections linked into consistent trajectories, which is more involved than in the golf case since people move over time and the camera must be allowed to move as well to keep them in view.

As also explained in [Dimitrijevic 06], people in canonical poses are detected using *spatio-temporal templates* that are sequences of three silhouettes of a person, such as the ones of Fig. 4.3(c). The first corresponds to the moment just before they reach the target pose, the second to the moment when they have precisely the right attitude, and the third just after. Matching these templates against three-image sequences let us differentiate between actual people who move in a predictable way and static objects whose



(a)

(b) (c) (d)

Figure 4.3: Creating spatio-temporal templates. (a) Eight virtual cameras are placed around the model. Viewpoints 3 and 7 are not considered because they are not discriminant enough, resulting in six different training views. (b) A template corresponding to a particular view consists of several silhouettes computed at three consecutive instants. The small blue arrows in image Camera 1 / Frame 1 represent edge orientations used for matching silhouettes for some of the contour pixels. (c) The three silhouettes of a walking template are superposed to highlight the differences between outlines. (d) Superposed silhouettes of a golf swing template.

outlines roughly resemble those of humans, which are surprisingly numerous. As a result, it turns out to be much more robust than earlier template-based approaches to people detection [Olson 97, Gavrila 99, Giebel 04].

As shown in Fig. 4.3(a), to build these templates, we introduced a virtual character that can perform the same captured motions we used to build the motion model discussed in Chapter 3 and rendered images as seen from virtual cameras in six different orientations. The rendered images are then used to create templates such as those depicted by Fig. 4.3(b). They are rescaled to seven different sizes ranging from 52×64 to 92×113 pixels, so that an image at one scale is 10% larger than the image at one scale below. From each one of the rendered images, we extract the silhouette of the model. Each template is made of the silhouette corresponding to the canonical pose, the one before, and the one after. The silhouettes are represented as sets of oriented pixels that can be efficiently matched against image sequences. We refer to [Dimitrijevic 06] for further details.

4.3 Linking Detections to obtain 2D Trajectories

In our scheme, people should be detected at the beginning of every walking cycle but are occasionally missed. To link these sparse detections into a complete trajectory, we have implemented a Viterbi-style algorithm. It is important to note that the system is trained on a very specific pose, the left leg in front of the right one, which helps our algorithm resolve ambiguities by giving higher scores to the correct detections. We could have used two keyposes instead of one for each walking cycle, but we empirically found that using one was a good trade-off between tracking robustness and computational load.

As shown in Section 4.7 even missing a detection out of two can still lead to reliable results. Finally these detections include not only an image location but also the direction the person faces, which is an important cue for linking purposes.

Ground Plane Registration Since the camera may move, we work in the ground plane, which we relate to each frame by a homography that is computed using a standard technique [Simon 00], illustrated in Fig. 4.4. In practice, we manually indicate the ground plane in one frame and compute an initial homography between it and the world ground plane G_w – H_0^w. Then, we detect interest points in both the reference frame and the next one and match them. From the set of correspondences we compute the homography H_1^0 between the subsequent frame's ground plane and the reference frame's ground plane, and further the homography H_1^w from the subsequent frame's ground plane to the world ground plane. We repeat this process for all the frames $I_i, i = 1..N$ obtaining the homographies $H_i^w, i = 1..N$ between each of them and world ground plane, N being the number of the sequence frames. This makes it easy to compute the world ground plane coordinates of the detections knowing the 2D coordinates in the frame's ground plane. Since there are specific orientations associated with each detection, we also recalculate these orientations with respect to the world ground plane.

Formalizing the Problem The homographies let us compute ground plane locations and one of the possible orientations for all detections, which we then need to link while ignoring potential misdetections. To this end, we define a hidden state at time t as the oriented position of a person on the ground plane $L_t = (X, Y, O)$, where t is a frame

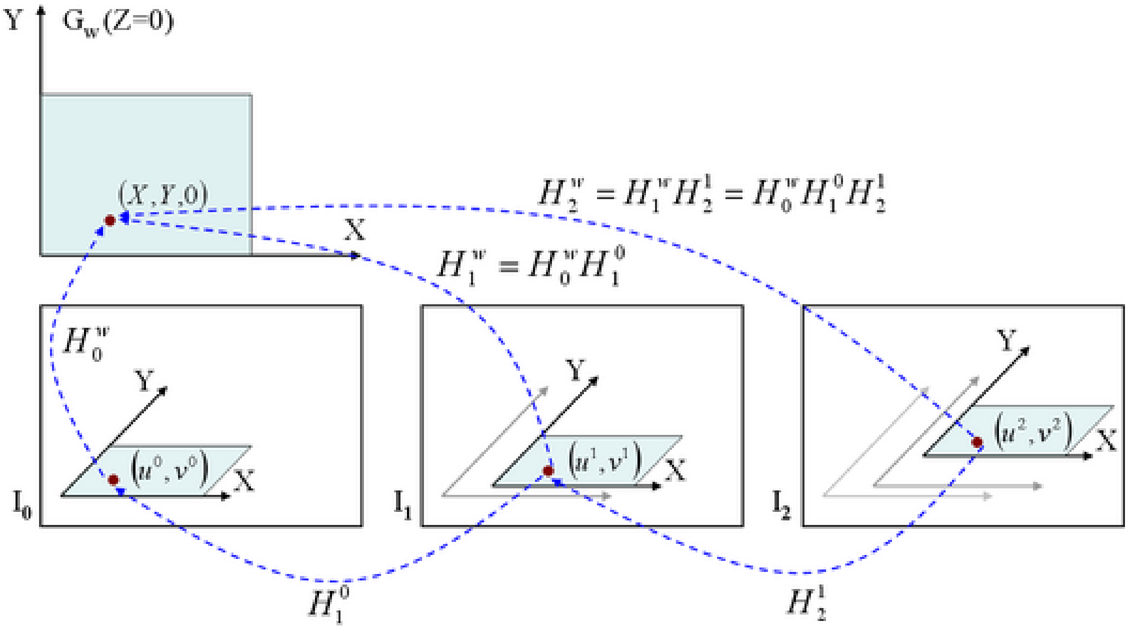


Figure 4.4: Ground plane tracking. Given the corresponding interest points in each pair of consecutive frames in the sequence $I_i, i = 0..N$ it is possible to compute the pairwise homographies $H_{i+1}^i, i = 0..N-1$ between consecutive frames. Furthermore, knowing the initial homography between the world ground plane and the reference image H_0^w, we compute the required homographies $H_i^w, i = 1..N$ between each of the frames and the world ground plane

.

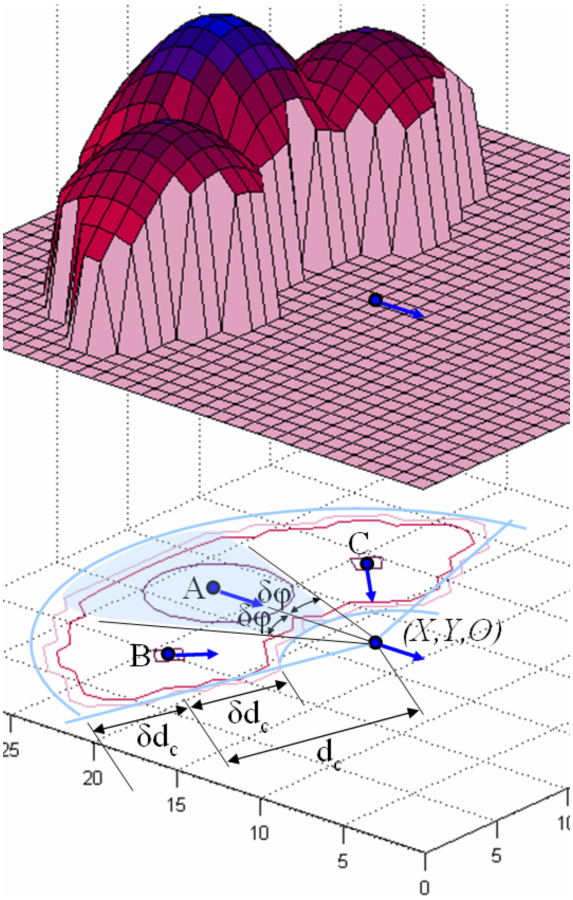


Figure 4.5: Transitional probabilities for the hidden state (X, Y, O)**.** They are represented by three Gaussian distributions corresponding to three possible previous orientations. Each Gaussian covers a 2D area bounded by two circles of radii $d_c - \delta d_c$ and $d_c + \delta d_c$, where δd_c represents an allowable deviation from the mean, and by two lines defined by tolerance angle $\delta\varphi$.

index, (X, Y) are discretized ground plane coordinates, considering a grid size of 10 cm, and O is one of the possible orientations.

We introduce the maximum likelihood estimate of a person's trajectory ending up at state i at time t

$$\Gamma_t(i) = \max_{l_1,...,l_n} P(I_1, L_1 = l_1, ..., I_n, L_n = l_n) \;\; , \tag{4.1}$$

where I_j represents the j^{th} frame of the video sequence. Casting the computation of Γ in a dynamic programming framework requires introducing probabilities of observing a particular image given a state and of transitioning from one state to the next.

We therefore take b_{it}, the probability of observing frame I_t given hidden state i, to be

$$b_{it} = P(I_t | L_t = i) \sim \frac{1}{d_{\text{bayes-chamfer}}} \;\; , \tag{4.2}$$

where $d_{\text{bayes-chamfer}}$ is a weighted average of the chamfer distances between projected template contours and actual image edges. This makes sense because the coefficients used to weight the contributions are designed to account for the relevance of different silhouette portions in a Bayesian framework [Dimitrijevic 06].

We also introduce the probability of transition from state j at time t' to state i at time t

$$a_{ji}^{\Delta t} = P(L_t = i | L_{t'} = j), \text{with } \Delta t = t - t'. \tag{4.3}$$

Since we only detect people when their legs are spread furthest apart, we can only expect a detection approximately every $N_c = 30$ frames for an average v = 5 km/h walking speed in a 25 Hz video. This implies an average distance $d_c = \frac{vN_c}{25}$ between detections. We therefore assume that $a_{ji}^{\Delta t}$ for state $i = (X, Y, O)$ follows a Gaussian distribution centered at (X_μ, Y_μ) such that

$$\sqrt{(X - X_\mu)^2 + (Y - Y_\mu)^2} = d_c \;\; , \tag{4.4}$$

and positioned in the direction 180° opposite to the orientation O, as depicted by point A in Fig. 4.5. This Gaussian covers only the hidden states with orientation equal to O and within the 2D area bounded by two circles with $d_c - \delta d_c$ and $d_c + \delta d_c$ radii, where δd_c represents an allowable deviation from the mean, and two lines defined by tolerance angle $\delta\varphi$. Similarly, we define two more Gaussians centered according to Equation (4.4) and positioned in the directions of $-3\pi/4$ and $+3\pi/4$ opposite to the orientation O, as depicted by points B and C in Fig. 4.5 respectively. These Gaussians cover the states with orientations $O + \pi/4$ and $O - \pi/4$, respectively. The standard deviations of the Gaussian distributions may be chosen arbitrarily as long as $\sigma_A > \sigma_B$ and $\sigma_B = \sigma_C$, which favors straight trajectories and penalizes sudden turns.

Linking Sparse Detections Given the probabilities of Eq. 4.2 and 4.3, if we could expect a detection in every frame, linking them into complete trajectories could be done using the Viterbi algorithm to recursively maximize the Γ_t likelihood of Eq. 4.1.

However, since we can only expect a detection approximately every $N_c = 30$ frames, we allow the model to change state directly from $L_{t'}$ at time t' to L_t at time t ($t' < t$), $N_c - \delta t < t - t' < N_c + \delta t$ and skip all frames in between. The value δt is a frame distance tolerance that we set to 10 in our implementation.

This lets us reformulate the maximization problem of Eq. 4.1 as one of maximizing

$$\begin{aligned}\Gamma_t(i) &= \max_{l_{t_1},\ldots,l_{t_n}} P(I_{t_1}, L_{t_1} = l_{t_1}, \ldots, I_{t_n}, L_{t_n} = l_{t_n}) \\ &= b_{it} \max_{j,\Delta t} (a_{ji}^{\Delta t} \Gamma_{t-\Delta t}(j)) \ , \qquad (4.5)\end{aligned}$$

where $t_1 < t_2 < \ldots < t_n$ are the indices of the frames I in which at least one detection occurred and $N_c - \delta t < \Delta t < N_c + \delta t$.

This formulation initially retains for each detection several hypotheses with different orientations, and then allows the dynamic programming algorithm to select those that provide the most likely trajectories according to the probabilities of Eq. 4.2 and 4.3. If a detection is missing, the algorithm simply bridges the gap using the transition probabilities only. For the sequences of Fig. 4.6 and Fig. 4.8, this yields the results depicted by Fig. 4.7 and Fig. 4.9.

4.4 Interpolation of 3D Poses between Detections

An added bonus of our approach to detecting people, is that to each detection we can associate the set of $\{\alpha_i\}$ PCA coefficients coming from the corresponding database motion, as defined in Eq. 3.5. Averaging the coefficients corresponding to two consecutive detections and sampling the μ_n phase parameter of Eq. 3.3 at regular intervals gives us a pose estimate in each intermediate frame. In the golfing case where the body's center of gravity moves little, this is enough to characterize the whole motion since we can assume that the $\mathbf{g}_t$ vector of Eq. 3.1 that encodes the position and orientation of the body root remains constant except for the component that encodes the rotation around the z axis. In the walking case, this is of course not true and we use the position of the detected silhouettes on the ground plane to estimate the person's 3D location and orientation. We then use cubic spline interpolation to derive estimate $\mathbf{g}_t$ values in between, as will be discussed in more details in Section 4.5.

The whole procedure is very simple and naturally extends to the case where a key

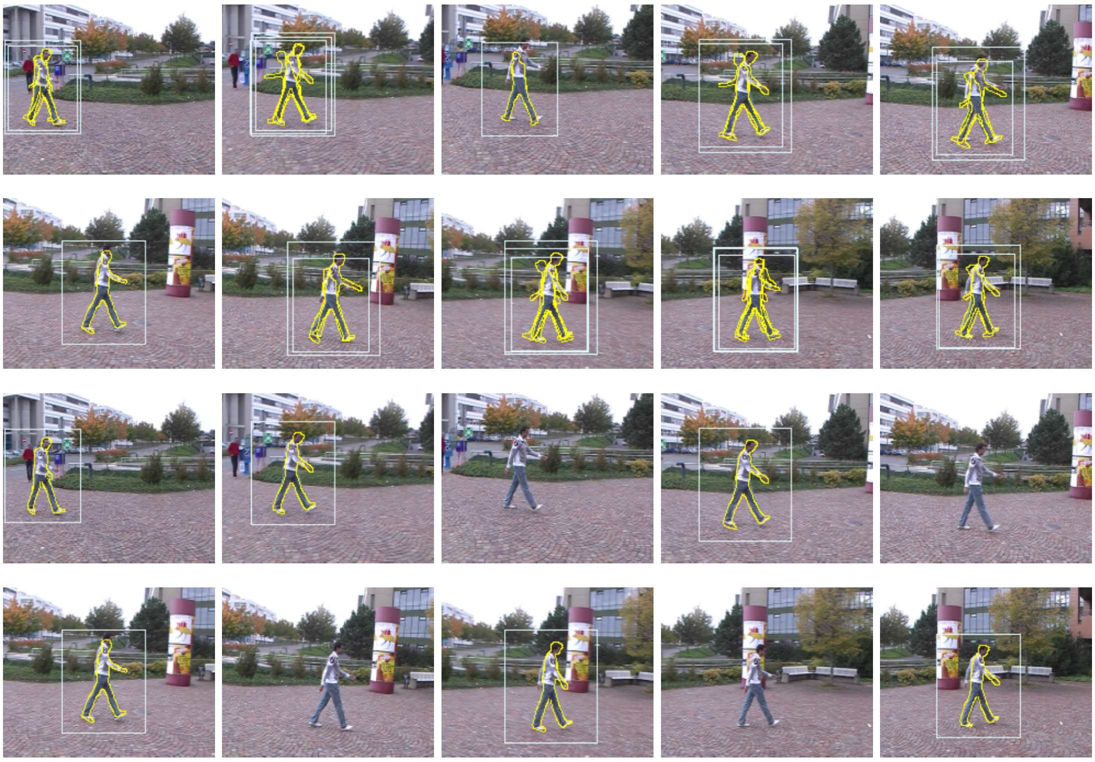


Figure 4.6: Filtering silhouettes with temporal consistency on an outdoor sequence acquired by a moving camera. **First two rows:** Detection hypotheses. **Third and fourth row:** Detections after filtering out the detection hypotheses that do not lie on the recovered most probable trajectory. Note that the extremely similar poses in which is very hard to distinguish which leg is in the front are successfully disambiguated by our algorithm.

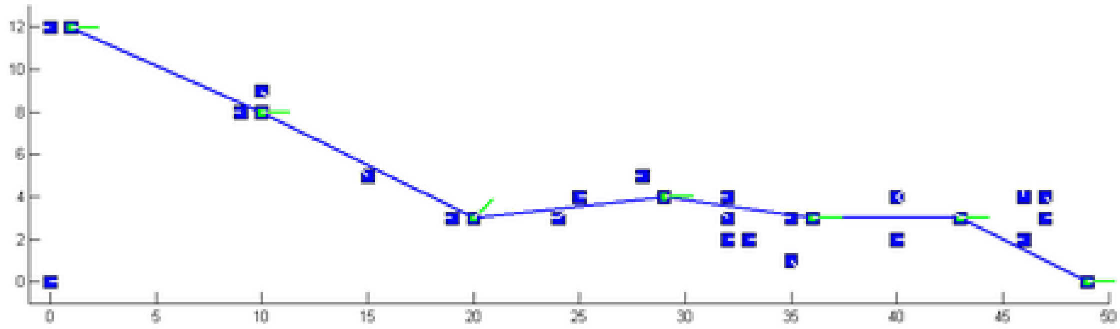


Figure 4.7: Recovered trajectory for the sequence depicted by Fig. 4.6. Dark blue squares represent detection hypotheses and bright short lines inside them represent the detection orientations. Smaller light green squares and lines represent the retained detections and their orientations respectively. These detections form the most probable trajectory depicted by dark blue lines.

posture has been missed, which can be easily detected by comparing the number of frames between consecutive detections to its median value for the whole sequence. In this case, a longer motion must be created by concatenating several motion cycles —usually 2, and never more than 3 in our experiments— depending on the number of frames between detections. This new motion is then resampled as before. Obviously the initial predictions then lose in accuracy, but they usually remain precise enough to retrieve the correct poses thanks to the refinement process described in the following subsection.

4.5 Image Likelihood for Refinement

In this section we will analyze the two techniques that we have developed to refine the initial estimates provided by the previous steps in order to better fit the images. The

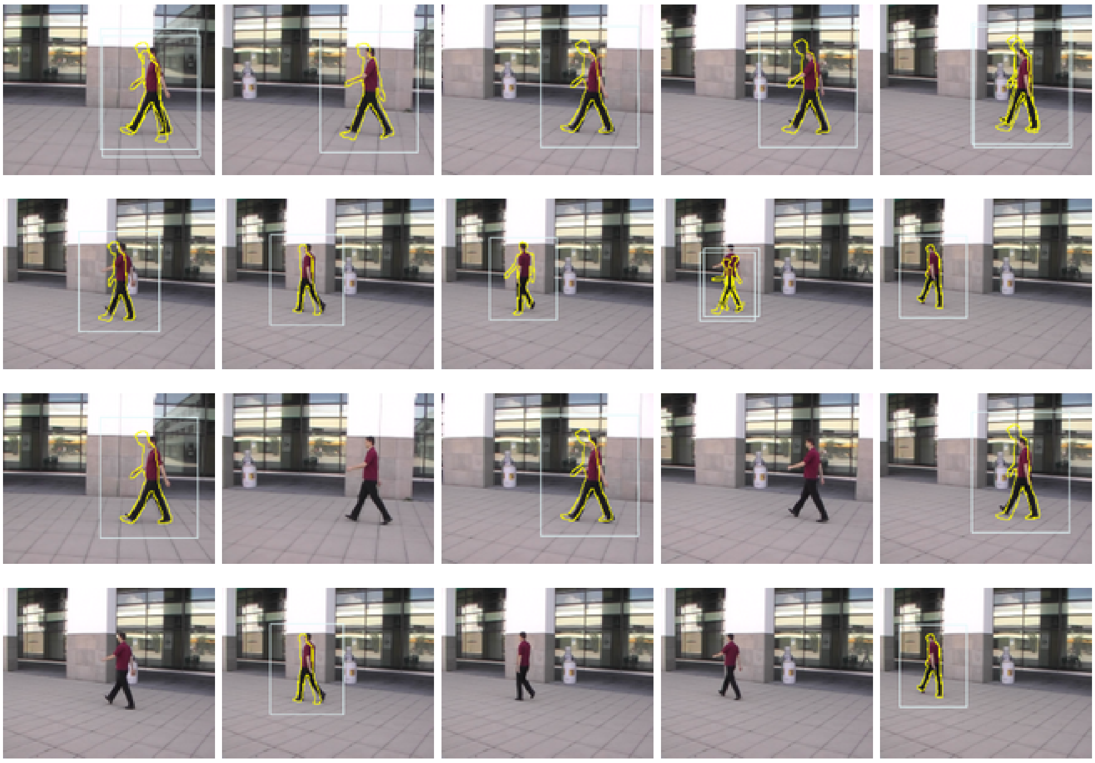


Figure 4.8: Filtering silhouettes with temporal consistency on an outdoor sequence acquired by a moving camera. **First two rows:** Detection hypotheses. **Third and fourth row:** Detections after filtering out the detection hypotheses that do not lie on the recovered most probable trajectory.

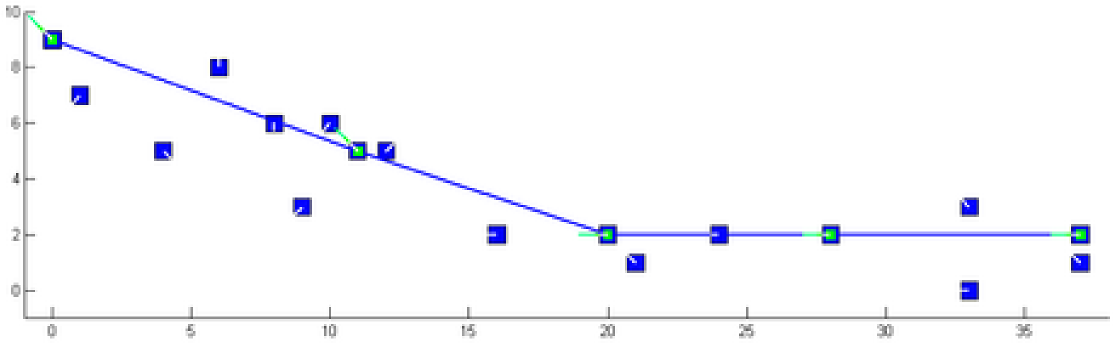


Figure 4.9: Recovered hypothetical trajectory for the sequence depicted by Fig. 4.8. Dark blue squares represent detection hypotheses and bright short lines inside them represent the detection orientations. Smaller light green squares and lines represent the retained detections and their orientations respectively. These detections form the most probable trajectory depicted by dark blue lines.

first one involves estimating the likelihood of the input images under the current model, while the second one makes use of Generalized Expectation Maximization to find correspondences between image edges and body limbs.

4.5.1 Appearance-based Likelihood

To track a person walking about, the camera usually has to move to keep him in view. Therefore we cannot use a simple background subtraction technique to create the background image we require for refinement purposes and adopt the more sophisticated approach depicted by Fig. 4.10. We treat each image of the sequence in turn as a reference and consider the few images immediately before and after. We compute homographies between the reference and all other images [Hartley 00, Simon 00], which is a reasonable

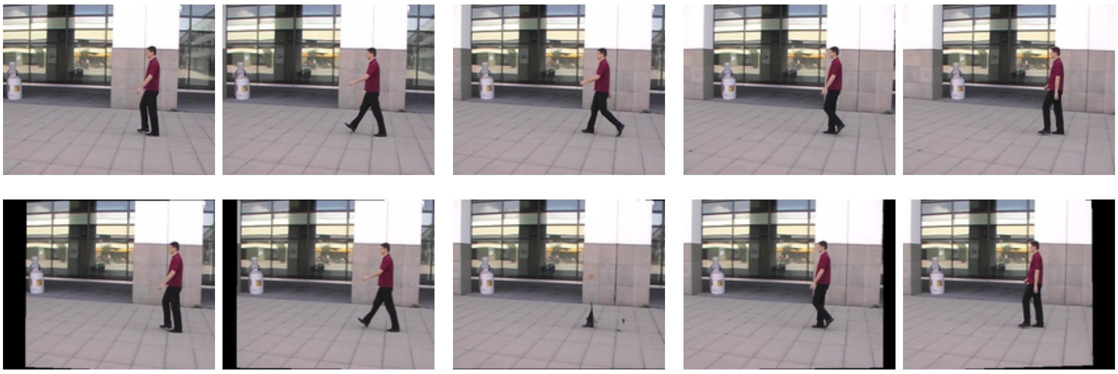


Figure 4.10: Synthesizing a background image. **First row:** The central image is the reference image whose background we want to synthesize. The other four are those before and after it in the sequence. **Second row:** The same four images warped to match the reference image. Computing the median image of these and the reference image yields the central image, which is the desired background image.

approximation of the frame-to-frame deformation because the time elapsed between successive frames is short and lets us warp all the images into the reference frame. Then, by computing the median of the V values for each pixel in the HSV color space and taking as output its corresponding pixel value, we obtain background images with few artifacts.

The poses obtained using the method discussed in the previous section are only approximative. To refine them, we use appearance information to compute the likelihood of the input image, given a particular pose and the obtained background mode. Maximizing such likelihood then lets us refine the poses in each individual frame.

In our implementation, we depart from typical generative approaches in an important way: As shown in the third row of Fig. 4.1, we not only create an appearance model for the person but also for the background. As illustrated by Fig. 4.11, this is important

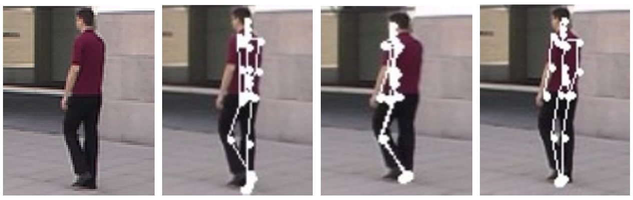


Figure 4.11: Appearance based refinement process. The images are from left to right the input image, the initialization given by the interpolation process, the result obtained without using a background image, and finally the result obtained as proposed. Whole parts of the body can be missed when the background is not exploited.

because it effectively constrains the projections of the reconstructed model to be at the right place and allows recovery of the correct pose even when the initial guess is far from it.

Assuming for simplicity that, given an estimated pose $\hat{\psi}_i$, a background, and a foreground model, all the pixels (u, v) in image I_i are conditionally independent, we write

$$p(I_i|\hat{\psi}_i, \hat{\mathbf{g}}_i) = \prod_{(u,v)\in I_i} p(I_i(u,v)|\hat{\psi}_i, \hat{\mathbf{g}}_i). \tag{4.6}$$

We estimate $p(I_i(u, v)|\hat{\psi}_i)$ for each pixel of frame i as follows: Given the generated background images, we project our human body model according to pose $\hat{\psi}_i$. As discussed above, individual limbs are modeled as cylinders to which we associate a color histogram obtained from the projected area of the limb in the frames where the silhouettes were detected. We project the body model onto the generated background image to obtain a synthetic image, such as those depicted by the third row of Fig. 4.1. If (u, v) is located within the projection of a body part, we take $p(I_i(u, v)|\hat{\psi}_i)$ to be proportional to the value of its corresponding bin in the color histogram of the body part. If, instead, (u, v) is located on the background, we sample $p(I_i(u, v)|\hat{\psi})$ from a Gaussian distribution centered on the corresponding pixel value in the synthetic background image B_i, with fixed covariance Σ. We therefore write

$$p(I_i(u, v)|\hat{\psi}, \hat{\mathbf{g}}) = \begin{cases} h_{part(u,v)}(I_i(u, v)) & \text{if } (u, v) \in F \\ N(B_i(u, v), \Sigma; I_i(u, v)) & \text{if } (u, v) \notin F \end{cases} , \tag{4.7}$$

where F represents the projection of the body model into the image. Modeling both the foreground and background appearance helps in achieving more accuracy and robustness, as already noted in the literature [Isard 01, Sidenbladh 03] for static camera cases. This function will then be optimized following the strategies explained in Section 4.6.

4.5.2 Generalized Expectation Maximization for Limb to Edge Matching

The second approach we have designed is structured as follows: first of all we obtain a reliable initial estimate of the 3D configuration of the person, using the key-pose detection technique together with the corresponding motion model, as described at the beginning of this chapter. Then we pre-process the video sequence we use as input in order to obtain a pretty clean edge image even if the camera is moving. Finally we use Generalized Expectation Maximization (GEM) to refine the initial pose estimation in a frame-wise fashion. This is done by matching the image edges to the edges obtained by projecting a 3D model of the person where limbs are considered as cylinders. We will explain in detail the different phases in the following paragraphs.

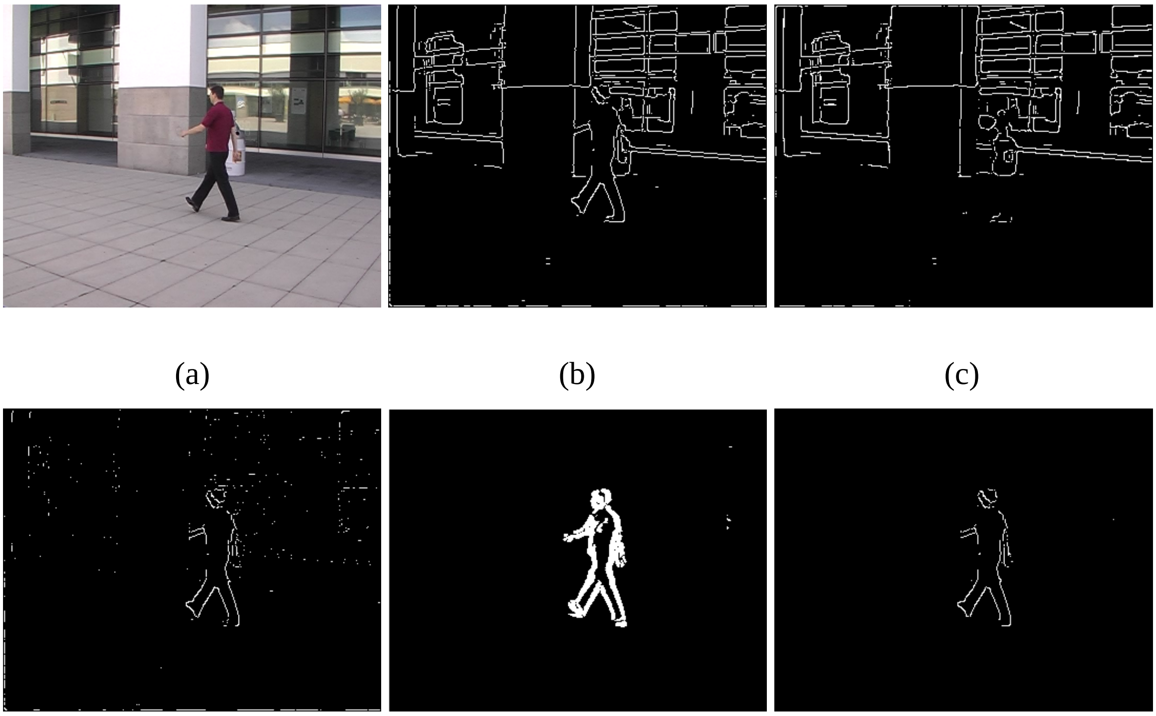
Sequence Pre-Processing To cope also with sequences shot by a moving camera, we decided to elaborate the input images in order to retrieve the edges corresponding to the moving objects. These are assumed to be the objects that move in the image at a different velocity than the background. This phase is composed of two main parts:

- *Motion Detection:* This step is taken from [Odobez 97] and simply retrieves, using optical flow, which pixels in the image are used to estimate the global motion of the camera. It also retrieves which pixels are considered as outliers for this estimation, and these are the pixels on which we will focus our attention since they are the ones that move at a different velocity than the background.

- *Background matching:* To obtain a more robust estimate of the edges belonging to the foreground, we also adopt a homography-based technique, similar to the one we use in the other type of objective function. Assuming that the motion of the camera is not too fast and not too close to the scene, we can consider the background to be planar. We then can simply take a window of N frames centered around the current one I_t and match them to I_t using a standard approach based on robust estimation of homographies using keypoints. Then we extract the edges, using a Canny-style edge detector, from all the frames in the window. Finally we warp all the obtained edge images to match I_t, using the previously computed homographies. For all the pixels we will now have a set of N observations, which correspond to the same pixel being edge (1) or not (0) in the warped images. Now simply taking the median of these values for each pixel will tell us which edge pixels belong to the foreground (if the median is 0) and which to the background (if the median is 1). At this point we have an estimate of the edge pixels belonging to the background at frame I_t, and simply subtracting this estimate from the edge extraction performed at I_t will give us an estimate of the edge pixels that belong to the foreground, i.e. to the moving person.

By making a simple intersection of the outputs of these two steps, for each input frame, we will obtain a robust estimate of which pixels belong to the foreground and are at the same time part of some edges. All the parts of this pre-processing algorithm are summarized in Fig. 4.12. We will use the output of this procedure as input for the following phase. Note that this phase can easily be replaced by a standard background subtraction algorithm if the camera is not moving.

Pose refinement through GEM Before explaining how we plugged the GEM algorithm into our framework, some definitions are provided. The observations points $\mathbf{x} = \{x_1, \ldots, x_M\}$ are the points belonging to the contours obtained in the pre-processing phase, from a single input frame. Our goal is then to fit a body configuration to these observation points. To do so, we suppose that $\mathbf{x}$ are sampled from a 2D mixture distribution of K components (Gaussian laws) and an outlier component (uniform law). Each Gaussian is associated to one limb's side of the projected body pose. The parameters of the k^{th} Gaussian, i.e. its mean and covariance, are denoted as θ_k. Let us note that θ_k is a function of the state of the body, and therefore a function of its low dimensional embedding $\lambda = (\beta_1 \ldots \beta_n)$. This parameterization is straightforward and is done as follows: From a given value of λ, the body state, defined by the 3-dimensional body pose $\mathbf{g}$ and by the set of joint angles ψ, is used to generate a 3D representation of the human body.



(a) (b) (c)

(d) (e) (f)

Figure 4.12: Summary of the pre-processing algorithm: (a) Input image. (b) Edges extracted from the input image. (c) Background edges reconstructed through homographies. (d) Subtraction between (b) and (c). (e) Outliers retrieved by the camera motion estimation technique. (f) Final output of the algorithm, obtained as intersection between (d) and (e).

This representation has limbs which are considered as cylinders of different radius and length, depending on the body part. Then this 3D model is projected onto the image and generates two segments for each cylinder, which should represent the 2 sides of the limb. Finally these segments are converted into Gaussian distributions, using their midpoint as representation of the mean and their length and a constant width to model an appropriate covariance matrix.

We then formalized the problem of fitting the projected body pose, now described as a Gaussian mixture, to the observed 2D cues as a classification task that could be carried out by the GEM algorithm. This problem boils down to the problem of finding an optimal value of λ such as the mixture components explain the image observation. The algorithm performs in 2 steps: First, each edge pixel is assigned to one of the components of the mixture. Let us note that a uniform component is added to the mixture to account for the corrupted observations. Second, the body configuration, i.e the mixture distribution, is fitted to the edge pixels by finding a new value of the parameter λ that decreases a distance function.

The assignment variables are denoted $\mathbf{z} = \{z_1, \ldots, z_M\}$. The event $z_m = k$, $m = 1, \ldots, M$, $k = 0, \ldots, K$ means that the observation x_m is generated from the k^{th} component of the mixture. The case $k = 0$ corresponds to the outlier case. By assuming conditional independence of the observations, we have:

$$p(\mathbf{x}|\mathbf{z}, \lambda) = \prod_{m=1}^{M} p(x_m|z_m, \lambda).$$

As explained before, the likelihood of an edge point being generated by the k^{th} limb's side is modeled as a Gaussian distribution of parameters $\theta_k(\lambda) = (\mu_k(\lambda), \Sigma_k(\lambda))$:

$$p(x_m|z_m = k, \lambda) \sim \mathcal{N}(x_m; \theta_k(\lambda)) \quad \text{if } (k \neq 0). \tag{4.8}$$

Similarly, we define the likelihood of an observation to belong to the outlier cluster as a uniform distribution:

$$p(x_m|z_m = 0, \lambda) = U[A] = \frac{1}{A}, \tag{4.9}$$

where A represent the observed data area i.e the image area. For simplicity, we assume that the assignment variables are independent. Their prior probabilities are denoted

$$p(z_m = k|\lambda) = p(z_m = k) = \pi_k \quad \forall k = 0, \dots K$$

with

$$\sum_{k=0}^{K} \pi_k = 1 \tag{4.10}$$

and therefore

$$\pi_k = \frac{1}{K+1}, \tag{4.11}$$

coming from the assumption that all the body parts have the same prior probability. The components posterior probabilities are denoted as α_{mk}:

$$\alpha_{mk} = p(z_m = k|x_m, \lambda). \tag{4.12}$$

By applying Bayes' rule, we can obtain the following expression, where the observation likelihood are given by Eq. (4.8-4.9):

$$\alpha_{mk} = \frac{\pi_k \, p(x_m|z_m = k, \lambda)}{\sum_{j=0}^{K} \pi_j \, p(x_m|z_m = j, \lambda)}.$$

For $k = 1, \dots, K$, we have:

$$\alpha_{mk} = \frac{\pi_k |\Sigma_k(\lambda)|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\|x_m - \mu_k(\lambda)\|^2_{\Sigma_k(\lambda)}\right)}{\frac{2\pi\pi_0}{A} + \sum_{j=1}^{K} \pi_j |\Sigma_j(\lambda)|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\|x_m - \mu_j(\lambda)\|^2_{\Sigma_j(\lambda)}\right)}, \tag{4.13}$$

where the notation $\| \mathbf{a} - \mathbf{b}\|_{\Sigma}^{2} = (\mathbf{a} - \mathbf{b})^T \Sigma^{-1} (\mathbf{a} - \mathbf{b})$ accounts for for the Mahalanobis distance. For $k = 0$, we have:

$$\alpha_{m0} = 1 - \sum_{k=1}^{K} \alpha_{mk}. \tag{4.14}$$

GEM framework Given the probabilistic model defined above, the goal is to determine the value of λ whose associated mixture distribution better explains the observations $\mathbf{x}$. Treating assignments as the hidden variables, the GEM algorithm helps in achieving this goal by maximizing the joint probability $p(\mathbf{x}, \mathbf{z}|\lambda)$. This probability can be written as:

$$\begin{aligned} p(\mathbf{x}, \mathbf{z}|\lambda) &= p(\mathbf{x}|\mathbf{z}, \lambda)\, p(\mathbf{z}|\lambda) \\ &= \prod_{m=1}^{M} p(x_m|z_m, \lambda)\, p(z_m|\lambda) \\ &= \prod_{m=1}^{M} \prod_{k=0}^{K} [\pi_k\, p(x_m|z_m = k, \lambda)]^{\delta_k(z_m)} \end{aligned} \tag{4.15}$$

The random variables $\delta_k(z_m)$ are defined as follows:

$$\delta_k(z_m) = \begin{cases} 1 & \text{if } z_m = k \\ 0 & \text{otherwise} \end{cases}$$

Starting with the initial value $\lambda^{(0)}$, the GEM algorithm proceeds iteratively and the iteration t consists in searching for the parameters λ that optimize the following expression:

$$Q(\lambda|\lambda^{(t)}) = E[\log p(\mathbf{x}, \mathbf{z}|\lambda)|\mathbf{x}, \lambda^{(t)}],$$

where $\lambda^{(t)}$ is the current estimate at iteration t. The expectation is calculated over all the possible assignments $\mathbf{z}$. Using eq (4.15), we have:

$$\log p(\mathbf{x}, \mathbf{z}|\lambda) = \sum_{m=1}^{M} \sum_{k=0}^{K} \log(\pi_k\, p(x_m|z_m = k, \lambda))\, \delta_k(z_m)).$$

Remarking that:

$$E[\delta_k(z_m)|\mathbf{x},\lambda^{(t)}] = \sum_{k=0}^{K} \delta_k(z_m)\, p(z_m = k|\mathbf{x},\lambda^{(t)}) = \alpha_{mk}^{(t)},$$

where $\alpha_{mk}^{(t)}$ are the posterior likelihood calculated using Eq. (4.13-4.14) with $\lambda = \lambda^{(t)}$, we have:

$$Q(\lambda|\lambda^{(t)}) = \sum_{m=1}^{M}\sum_{k=0}^{K} \alpha_{mk}^{(t)}\ \log(\pi_k\, p(x_m|z_m = k,\lambda)).$$

Replacing the likelihoods by their expression given by Eq. (4.8-4.9) leads to:

$$\begin{aligned} Q(\lambda|\lambda^t) \quad &= \quad \sum_{m=1}^{M}\sum_{k=1}^{K} \alpha_{mk}^{(t)} \left\{ -\frac{1}{2}\ \|x_m - \mu_k(\lambda)\|^2_{\Sigma_k(\lambda)} \right. \\ &\qquad \left. - \log\left(\pi_k\ (2\pi_k)^{-1}\ |\Sigma_k(\lambda)|^{-1/2}\right)\right\} \\ &+ \quad \sum_{m=1}^{M} \alpha_{m0}^{(t)}\ \log(A\ \pi_0) \end{aligned} \tag{4.16}$$

We can now formulate the GEM algorithm as iterations of two steps at time t:

- **E-step** From the current value $\lambda^{(t)}$, this step simply requires the computation of the posterior probabilities $\alpha_{mk}^{(t)}$ using eq. (4.13-4.14). Each probability $\alpha_{mk}^{(t)}$ represents the likelihood of assigning observation point m to the k^{th} limb's side or to the outlier class.

- **M-step** Provided that $\alpha_{mk}^{(t)}$ are computed, now $Q(\lambda|\lambda^t)$ needs to be maximized over λ. Since the analytical computation would be highly non-linear, the generalized version of the EM algorithm is applied. This simply means that, instead of maximizing $Q(\lambda|\lambda^t)$, we simply find a state $\lambda^{(t+1)}$ that increases the value of $Q(\lambda|\lambda^t)$. In practice, several λ_i are sampled around $\lambda^{(t)}$ until this condition is reached. Usually 300 particles are enough for such optimization.

We iterate this procedure a certain number of times until an improvement in Q is obtained, and then retain the corresponding body pose calculated from $\lambda^{(final)}$ for the current frame.

4.6 Optimization

4.6.1 Frame-Wise Optimization

At this point we have all the tools to compute the likelihood of a certain pose given an input frame, which can be evaluated using one of the proposed techniques. Incorporating both foreground and background into our algorithms makes the measure reliable enough so that we did not have to use a robust estimator for our experiments. However, because it produces many local minima when the pose changes, we use stochastic optimization techniques that sample the pose space around the predicted pose in the low dimensional pose-space and retain the sample that yields the best score. Because we only search for poses around the predicted ones, we still benefit from the constraints provided by interpolating between key-poses. On the other hand, a frame-wise optimization can lead to jittering results.

4.6.2 Sequence-Based Optimization

To overcome the jittering issue, we decided to perform batch optimization over the whole motion at one time, which was only possible using the appearance-based objective function. In the golfing case there was no need to take into consideration the global motion, while for the walking case we had to adopt a special procedure: To account for the fact that walking trajectories are smooth both spatially and temporally, we do not treat the $\mathbf{g}_i$ terms as independent from each other, and the same holds for the μ_i terms. Instead, as we did for initialization purposes, we represent trajectories as 2–D cubic splines lying on the ground plane and whose shape is completely defined by the position and orientation of the body root node at the endpoints of a sequence, which we denote as $\mathbf{g}_{start}$ and $\mathbf{g}_{end}$. In other words, we write all the $\mathbf{g}_i$ as functions of $\mathbf{g}_{start}$ and $\mathbf{g}_{end}$. Similarly, we introduce a parameter $0 < \mu_c < 1$ that defines what percentage of the walking cycle has been accomplished during the first half of the sequence and derive all the other μ_i by simple linear interpolation. If the speed remains constant during a walking cycle, the value of μ_c is 0.5. In practice, it can go from 0.3 to 0.7 if the person speeds-up or slows-down between the first and the second half-cycle.

We can now refine the pose sequence between two detections by optimizing the objective function $L(\hat{\Psi})$ of Eq. 4.20 with respect to $(\mu_c, \alpha_1, \ldots, \alpha_m, \mathbf{g}_{start}, \mathbf{g}_{end})$.

To perform the refinement, we define the objective function $L(\hat{\Psi})$ as

$$-\log(p(\hat{\Psi}|I_1, \ldots, I_N))$$

of a pose sequence $\hat{\Psi} = \Psi(\mu_1, \cdots, \mu_N, \mathbf{g}_1, \cdots, \mathbf{g}_N, \alpha_1, \cdots, \alpha_m)$ in an image sequence $I_1, \ldots, I_N$. To compute it we consider the standard Bayesian formula

$$p(\hat{\Psi}|I_1, \ldots, I_N) = \frac{p(I_1, \ldots, I_N|\hat{\Psi}) \cdot p(\hat{\Psi})}{p(I_1, \ldots, I_N)}. \quad (4.17)$$

The $p(I_1, \ldots, I_N)$ term is constant and can be ignored. Because we have a dependable way to initialize Ψ, we express the prior as a distance from its initial value and write its negative log as

$$-\log p(\hat{\Psi}) = \sum_{k=1}^{m} \left(\frac{\alpha_k - \alpha_k^0}{\sqrt{\lambda_k}} \right)^2, \qquad (4.18)$$

where α_k^0 represents the initialization value for the k^{th} PCA parameter, given by the detections, and λ_k is the eigenvalue associated to the k^{th} eigenvector.

Assuming conditional independence of the appearance in consecutive frames given the motion model, we can decompose $p(I_1, \dots, I_N|\hat{\Psi})$ as

$$p(I_1, \dots, I_N|\hat{\Psi}) = \prod_{i=1}^{N} p(I_i|\hat{\psi}_i, \hat{\mathbf{g}}_i), \qquad (4.19)$$

where $\hat{\psi}_i = \psi(\mu_i, \alpha_1, \dots, \alpha_m)$ is the pose in image I_i, as defined by Eq. 3.6.

Given Eqs. 4.18, 4.19 and 4.6, we can write

$$L(\hat{\Psi}) = -\log(p(\hat{\Psi})) + \sum_{i=1}^{N} \sum_{(u,v)\in I_i} -\log(p(I_i(u,v)|\hat{\psi}_i)) \qquad (4.20)$$

and refine all the poses between detections by minimizing $L(\hat{\Psi})$ with respect to $(\mu_1, \cdots, \mu_N, \mathbf{g}_1, \cdots, \mathbf{g}_N, \alpha_1, \cdots, \alpha_m)$, which define the motion in the whole sequence. This minimization is performed stochastically by sampling particles thrown in the parameter space around the initialization.

Linking pose and Motion

Our experiments have shown that using a low dimensional representation regularizes the motion and yields much better convergence properties than using the full parameterization. However, this formulation does not exploit the fact that people usually walk in the direction they are facing. To remedy this problem, we explicitly link pose and motion as follows: Given a subject moving along a trajectory as depicted by Fig. 4.13, the angle

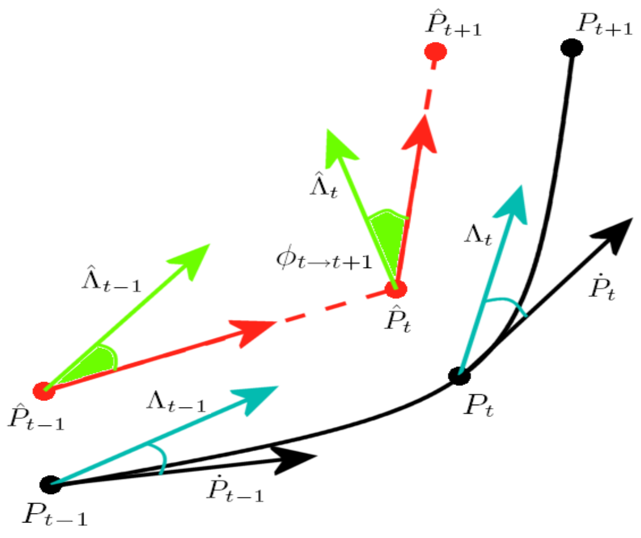


Figure 4.13: The continuous curve represents the real trajectory of the subject, while the dashed lines show its approximation by finite differences.

between $\dot{P}_t$, the derivative of the position, and the orientation Λ_t should in general be small. We can therefore write that

$$\frac{\dot{P}_t \cdot \Lambda_t}{||\dot{P}_t|| \cdot ||\Lambda_t||}$$

should be close to 1.

To enforce this, we can approximate the derivative of the locations using finite differences between estimated locations $\hat{P}$ at different time instants. This approximation is appropriate when we can estimate the location at a sufficiently high frequency (e.g. 25 Hz).

Our constraint then reduces to minimizing the angle between the finite differences approximation of the derivative of the trajectory at time t, given by $\hat{P}_{t+1} - \hat{P}_t$, and the object's estimated orientation given by $\hat{\Lambda}_t$. We write this angle, which is depicted as filled both at time $t-1$ and t in Fig. 4.13, as

$$\phi_{t \to t+1} = \mathrm{acos} \frac{\hat{\dot{P}}_t \cdot \hat{\Lambda}_t}{||\hat{\dot{P}}_t|| \cdot ||\hat{\Lambda}_t||} = \mathrm{acos} \frac{(\hat{P}_{t+1} - \hat{P}_t) \cdot \hat{\Lambda}_t}{||(\hat{P}_{t+1} - \hat{P}_t)|| \cdot ||\hat{\Lambda}_t||} \tag{4.21}$$

and will seek to minimize it. It is important to note that the constraint we impose is not a hard constraint, which can never be violated. Instead, it is a prior that can be deviated from if the data warrants it.

This in practice means that the body global position, which is controlled by the first three variables of the 6–D $\mathbf{g}_{start}$ and $\mathbf{g}_{end}$ vectors is not independent from the other three, which control orientation. We can therefore further improve our results by adding an additional term to our objective function to enforce this constraint. We define

$$L_{walk}(\hat{\Psi}) = L(\hat{\Psi}) + \mu \sum_{i=2}^{N} (\phi^2_{(i-1) \to i}) \tag{4.22}$$

where $\phi_{(i-1) \to i}$ is the angle, defined in Eq. 4.21, between the direction the person faces and the direction of motion and μ is a weighting term which is kept constant for all our experiments, and whose purpose is to make the two terms of the same order of magnitude. As demonstrated in [Fossati 08] and as will be shown in Section 4.7, minimizing $L_{walk}(\hat{\Psi})$ instead of $L(\hat{\Psi})$ has little influence on the recovered poses but yields more realistic global body orientations.

4.7 Results

In this section, we present our results, obtained using the appearance based likelihood, on golfing and walking sequences that feature subjects *other* than those we used to create our motion databases and seen from many different perspectives. A computationally expensive part of the algorithm is the refinement step since, for each particle, we must render the whole sequence, be it a walking cycle or a golf swing, and compute the image likelihood for each frame. In practice, the computation time is directly proportional to the number of particles adopted in the optimization phase. For a golf swing it is around 1 second for each particle, while it doubles for a walking step. Given that we usually took 300 particles to compute our results, our algorithm requires in average 5 minutes to extract the body poses of a golf swing and 10 minutes for a walking step.

4.7.1 Golfing

Fig. 4.14 depicts a golf swing by a professional golfer. By contrast, Fig. 4.15 depicts one performed by a former PhD student who does not play golf and whose motion is therefore far from correct. In both cases, our system correctly detects the key postures and recovers a 3D trajectory without any human intervention. This demonstrates that it is robust not only to the relatively low quality of the imagery but also to potentially large variations in the exact motion being recovered. Fig. 4.16 shows the background model that was recovered and used to generate the results of Fig. 4.14. Note that the feet are mistakenly made part of the background reconstruction and this results in their unwarranted motion. This is easily fixed by constraining them to remain on the ground.

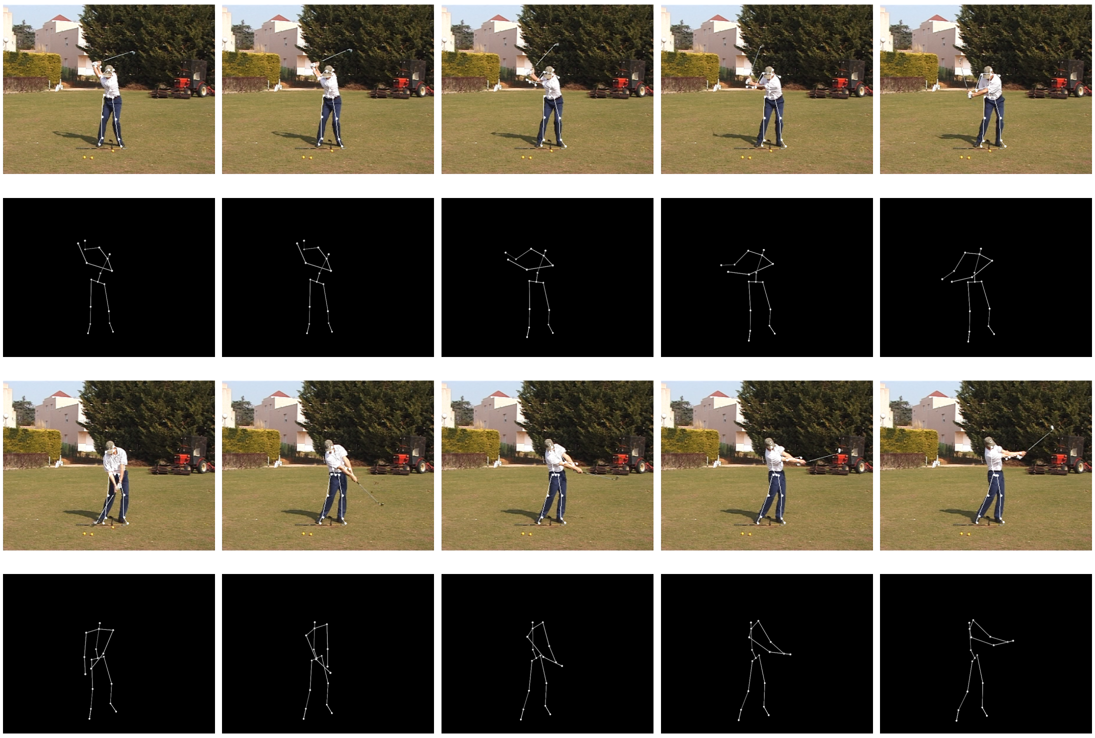


Figure 4.14: Reconstructing a golf swing performed by a professional player. **First and third row:** Frames from the input video with reprojected 3D skeletons. **Second and fourth row:** 3D skeleton seen from a different viewpoint.

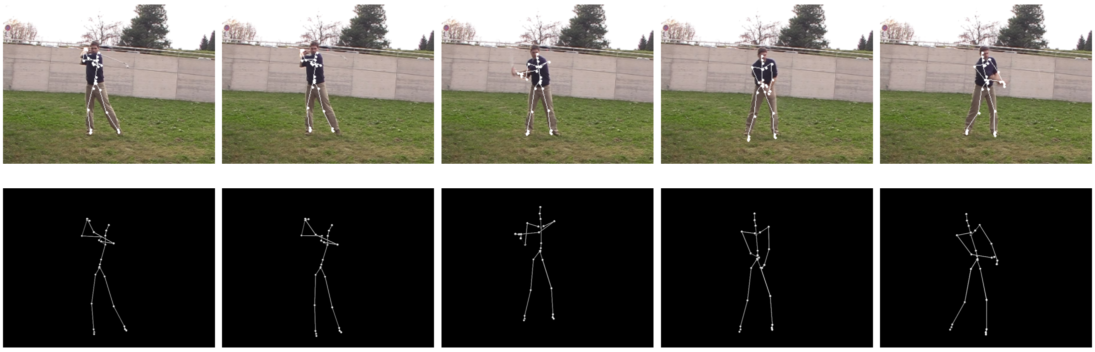


Figure 4.15: Reconstructing a golf swing performed by a novice player. **First row:** Frames from the input video with reprojected 3D skeletons. **Second row:** 3D skeleton seen from a different a different viewpoint.

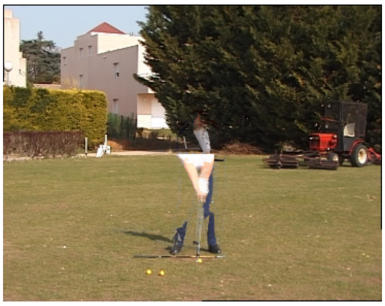


Figure 4.16: Background image used to generate the results of Fig. 4.14. Notice that there are some artifacts for instance in the feet area, which are anyway overcome by our algorithm.

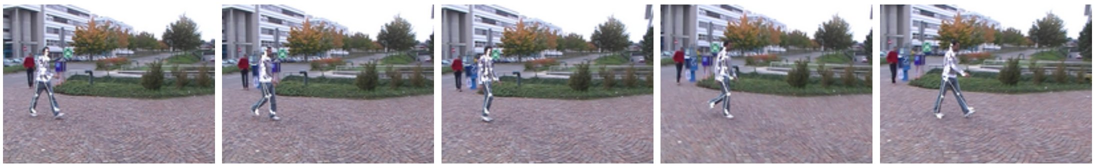


Figure 4.17: Recovered 3D skeletons reprojected into individual images of the sequence of Fig. 4.6, which was acquired by a camera translating to follow the subject.

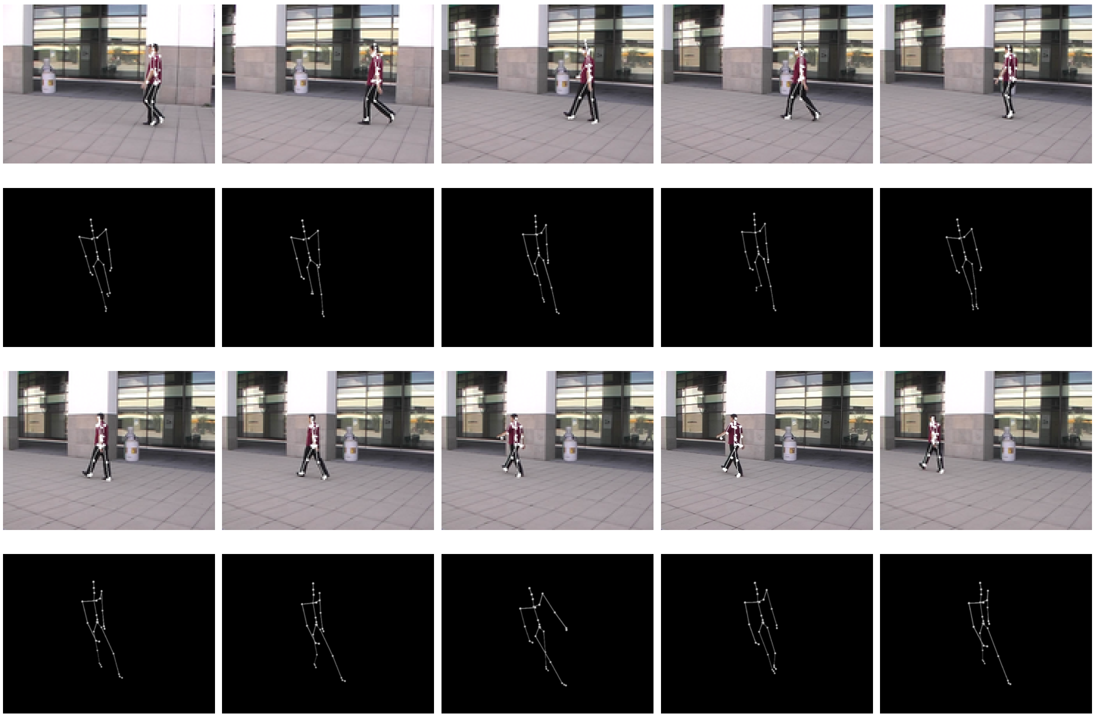


Figure 4.18: Final result for the subject of Fig. 4.8 who moves away from the camera and is eventually seen from behind. **First and third rows:** Frames from the input video with reprojected 3D skeletons. **Second and fourth rows:** 3D skeletons seen from a different viewpoint. The 3–D pose is correctly estimated over the sequence, even when the person goes far away and eventually turns his back to the camera. Note that the extremely similar poses in which it is very hard to distinguish which leg is in front are successfully disambiguated by our algorithm.

4.7.2 Walking

We now demonstrate the performance of our algorithm on walking sequences acquired under common but challenging conditions. In all cases except when we use the HumanEva dataset [Sigal 06] to quantify our results, the subject is seen against a cluttered background and the camera moves to follow him, which precludes the use of simple background subtraction techniques.

In the sequences of Figs. 4.17 and 4.18 the camera translates. Furthermore, in Fig. 4.18, the subject is seen first from the side and progressively from the back as he becomes smaller and smaller. In the sequence of Fig. 4.19, the subject walks along a circular trajectory and the camera follows him from its center. At some point the subject undergoes a total occlusion but the global model allows the algorithm to nevertheless recover both pose and position for the whole sequence. We can also recover the instantaneous speeds and the ground plane trajectory, as shown in Fig. 4.20.

All these results were obtained by minimizing the objective function of Eq. 4.22 that explicitly enforces consistency between the direction the person faces and the direction of motion. We also computed results by minimizing the objective function of Eq. 4.20, which does not take this consistency into account. When shown in projections in the original images, these two sets of results are almost indistinguishable. However, the improvement becomes clear when one compares the two trajectories of Fig. 4.20, one obtained without enforcing the constraint and the other with. To validate these results, we manually marked the subject's feet every 10 frames in the sequence of Fig. 4.19 and used their position with respect to the tiles on the ground plane to estimate their 3D coordinates. We then treated the vector joining the feet as an estimate of the body orientation and the midpoint as an estimate of its location. As can be seen in Table 4.1, linking orientation to motion produces a small improvement in the position estimate and a much more substantial one in the orientation estimate, which is consistent with what can be observed in Fig. 4.20. Obviously these numbers should be only considered in a relative way, and to have an idea of the quantitative performance of our algorithm we refer the reader to the results on the HumanEvaII sequence.

In the sequence of Fig. 4.21 the subject walks along a curvilinear path and the camera follows him, so that the viewpoint undergoes large variations. We are nevertheless able to recover pose and motion in a consistent way, as shown in Fig. 4.22 that depicts the recovered trajectory. Again, linking orientation to motion yields improved results.

Fig. 4.23 demonstrates the robustness of our approach to missed detections. We ran our algorithm on the same sequence as in Fig. 4.1 but ignored one out of every two detections. Note that, even though the subject is now only detected every other step, the algorithm's performance barely degrades.

To further quantify our results, we tracked subject S4 of the HumanEvaII dataset [Sigal 06] over 230 frames acquired by camera C1. Since it is static, we used the same simple approach as in the golf case to synthesize the background image we use to compute our image likelihoods. In Fig. 4.24 we plot the mean 3–D distance between the real position of some reference body joints and those recovered by our algorithm, which are commensurate with the numerical results of Table 4.1 that we obtained using our own sequences. Given that our approach is strictly monocular—we simply ignored the input of the other cameras—the 158mm average error our algorithm produces is within the

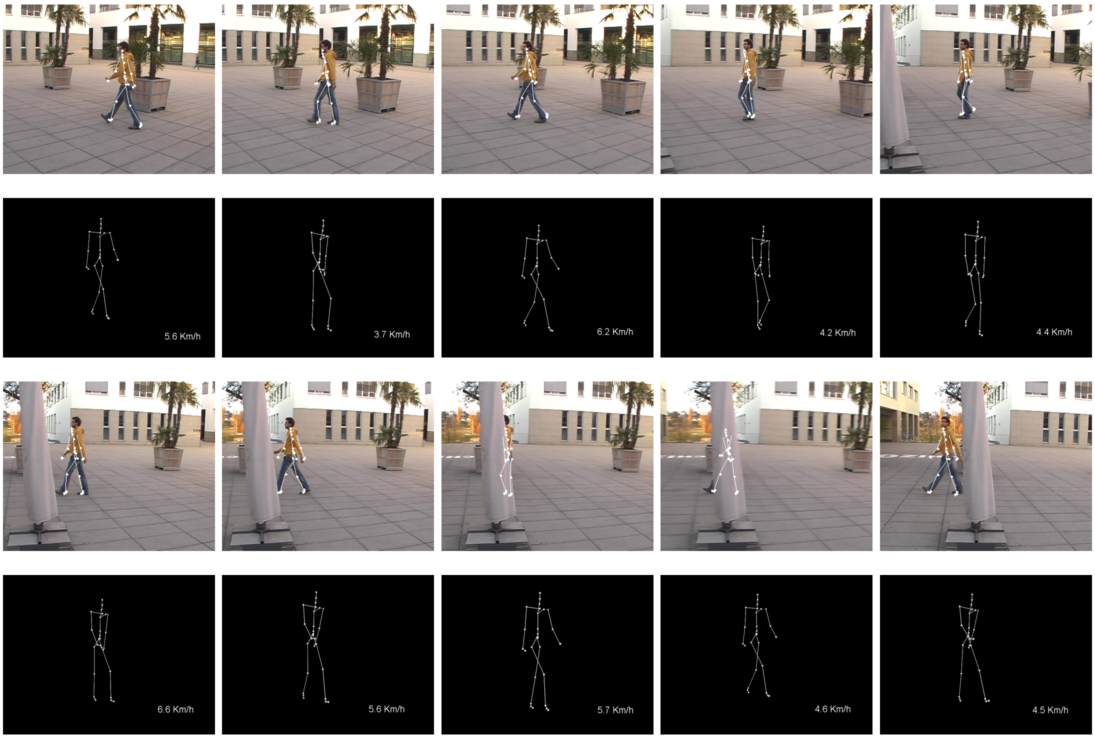
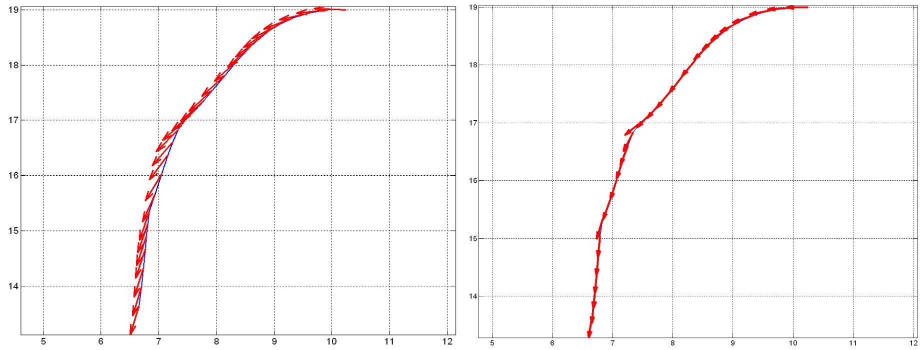


Figure 4.19: Subject walking in a circle. **First and third rows:** Frames from the input video with reprojected 3D skeletons. **Second and fourth rows:** 3D skeletons seen from a different viewpoint. The numbers in the bottom right corner are the instantaneous speeds derived from the recovered motion parameters.



(a) (b)

Figure 4.20: Recovered 2D trajectory of the subject of Fig. 4.19. The underlying grid is made of 1×1 meter squares and the arrows represent the direction he is facing. (a) When orientation and motion are not linked, he appears to walk sideways. (b) When they are, he walks naturally.

	X Error		Y Error		Orientation Error	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
No Link	12.0	7.1	16.8	11.9	11.7	7.6
Link	11.8	7.3	14.9	9.3	6.2	4.9

Table 4.1: Comparing the recovered position and orientation values for the body root node against ground truth data for the sequence of Fig. 4.19 in case we do not link orientation and motion (first line) and in case they are linked (second line). We provide the mean and standard deviation of the absolute positional error in the X and Y coordinates, in centimeters, and the mean and standard deviation of the recovered orientation error, in degrees.

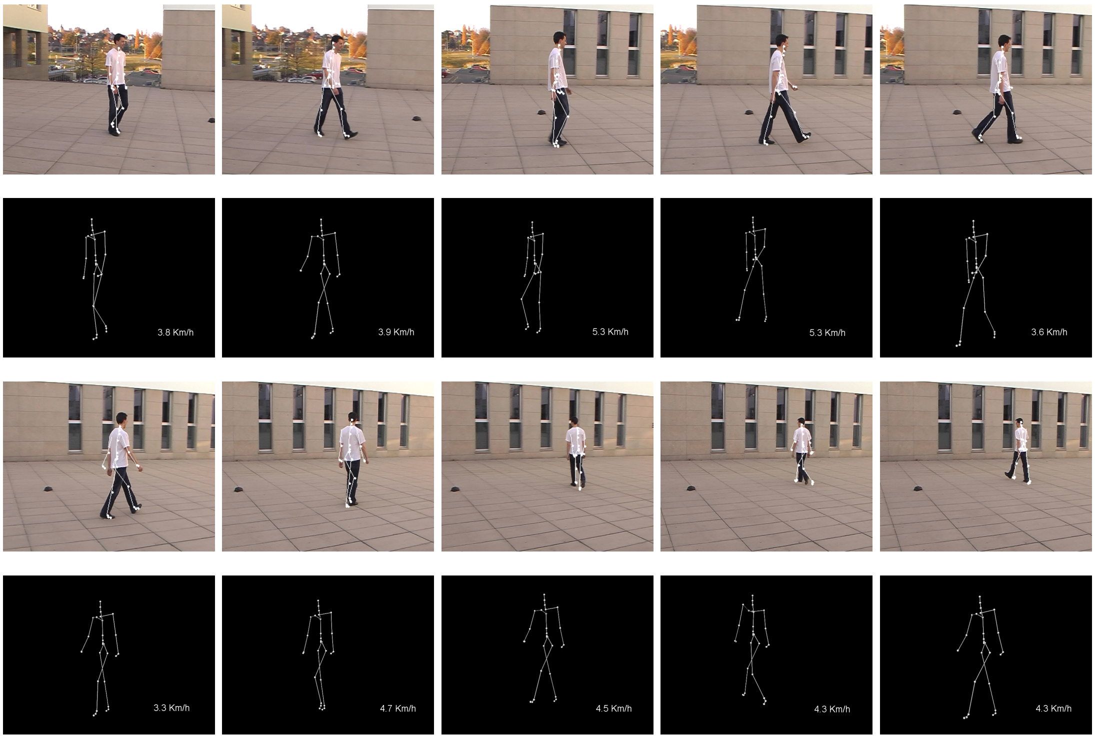
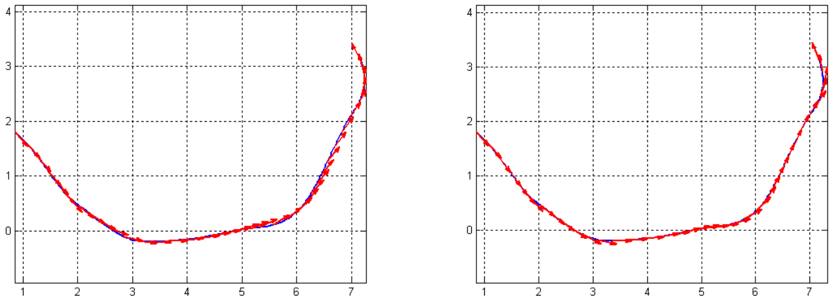


Figure 4.21: Pedestrian tracking and reprojected 3D model in a second sequence. **First and third rows:** Frames from the input video with reprojected 3D skeletons. **Second and fourth rows:** 3D skeletons seen from a different viewpoint. The numbers in the bottom right corner are the instantaneous speeds derived from the recovered motion parameters.



(a) (b)

Figure 4.22: Recovered 2D trajectory of the subject of Fig. 4.21. As in Fig. 4.20, when orientation and motion are not linked, he appears to walk sideway (a) but not when they are (b).

range of methods that make similar assumptions. By comparison, errors around 200mm are reported in [Li 06] and between 100 and 200mm in [Brubaker 06]. This is encouraging given the fact that we only use relatively coarse models and motions described by a reduced number of parameters. In other words, our algorithm is designed more for robustness, moving cameras, and recovery from situations where other algorithms might lose track, such as total occlusions, than for accuracy.

Of course the algorithm, even if it is designed for robustness, can fail. In the walking case, this can happen if the subject performs very sharp turns, thus preventing the Viterbi algorithm to infer the correct trajectory. Similarly, facing the camera for too long can result in loss of track since our detector is designed for people not seen completely frontally. This could be overcome by adding an appropriate detector, which could take advantage of the very reliable frontal head detection algorithms that now exist. In the golfing case, a misdetection of either the initial or the final pose would also cause a failure, but

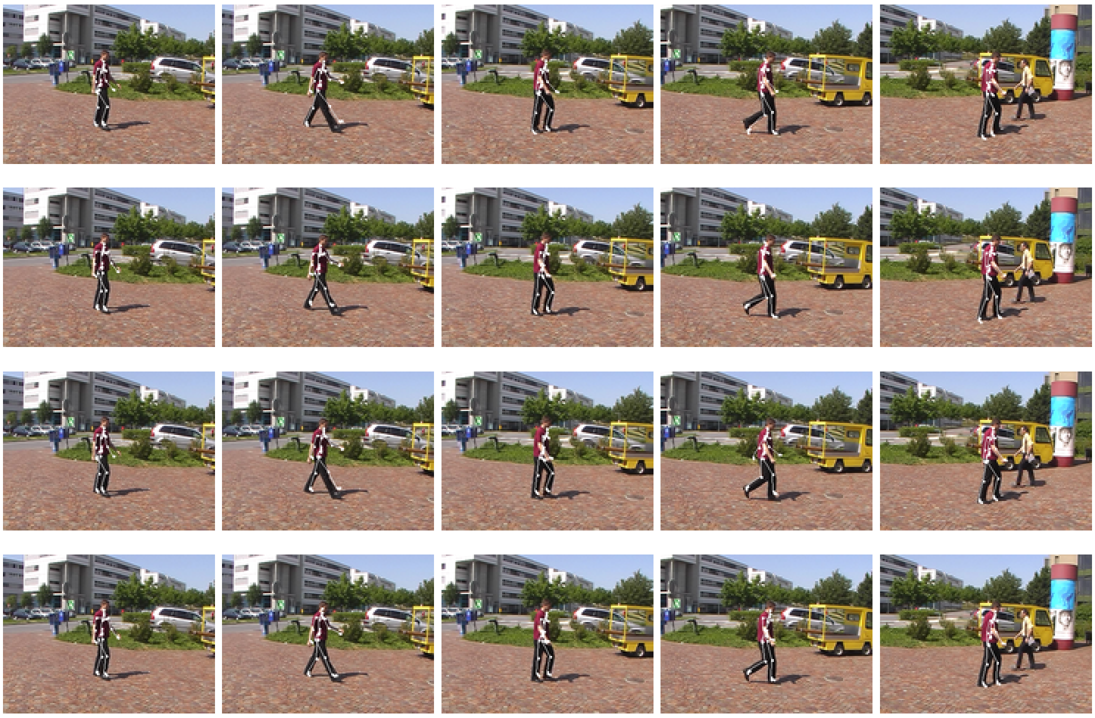


Figure 4.23: Robustness to misdetection. **First two rows:** Initial and refined poses for a sequence in which 3 consecutive key-poses are detected. **Last two rows:** Initial and refined poses for the same sequence when ignoring the central detection and using the other two. The initial poses are less accurate but the refined ones are indistinguishable.

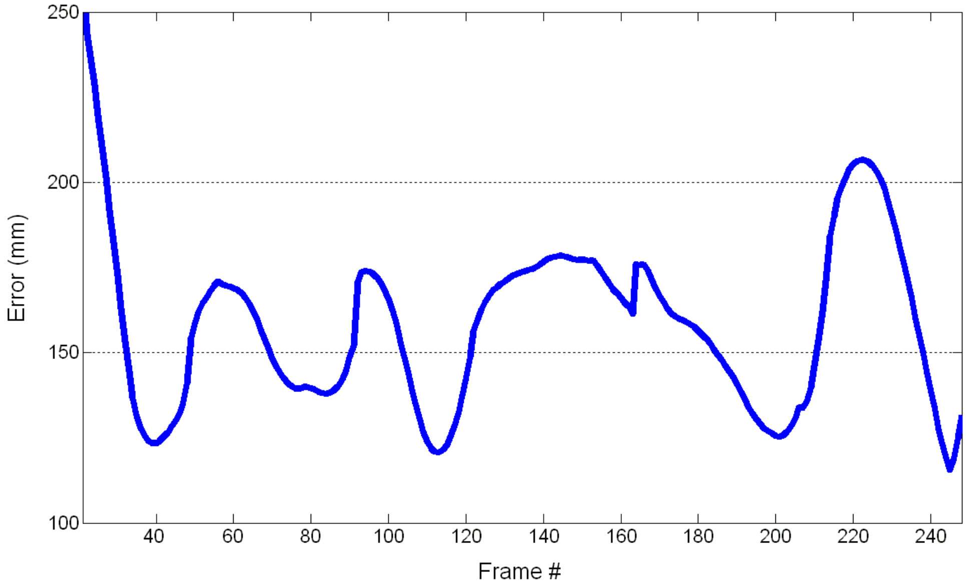


Figure 4.24: Absolute mean 3D error in joint location obtained on frames 21-248 of the HumanEvaII dataset for subject S4 and using only camera C1 as input. It is expressed in millimeters.

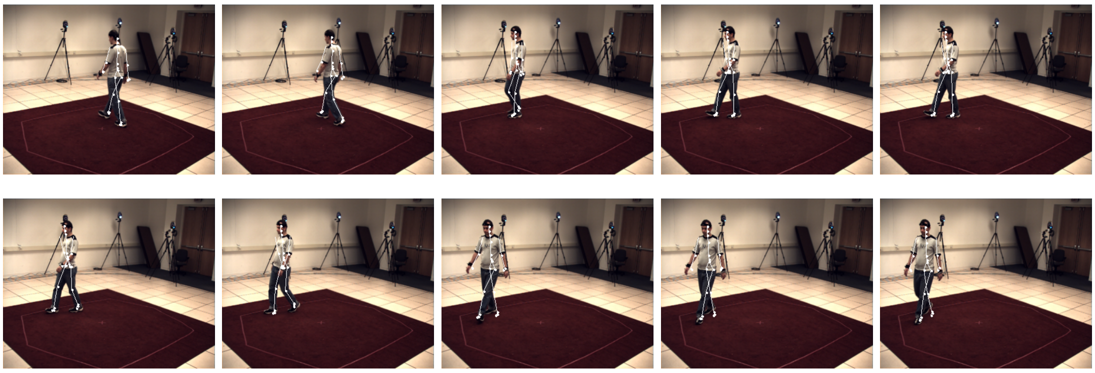


Figure 4.25: Tracking subject S4 from the HumanEvaII dataset using only camera C1. Obtained results projected onto the input frames.

they are infrequent because the pose is very characteristic.

Chapter 5

Observable Subspaces

The articulated body models used to represent human motion typically have many degrees of freedom, usually expressed as joint angles that are highly correlated. The true range of motion can therefore be represented by latent variables that span a low-dimensional space.

As also explained in Chapter 2 and demonstrated in Chapter 4, this has often been used to make motion tracking easier. However, learning the latent space in a problem-independent way makes it non trivial to initialize the tracking process by picking appropriate initial values for the latent variables, and thus for the pose, specially in the types of motion in which key-poses are not available. In this chapter, we show that by directly using observable quantities as our latent variables, we eliminate this problem and achieve full automation given only modest amounts of training data.

More specifically, we exploit the fact that the trajectory of a person's feet or hands strongly constrains body pose in motions such as walking, skating, skiing, or golfing. These trajectories are easy to compute and to parameterize using a few variables. We treat these as our latent variables and learn a mapping between them and sequences of body poses. In this manner, by simply tracking the feet or the hands, we can reliably guess initial poses over whole sequences and, then, refine them.

5.1 Framework

Our goal is to relate 3D motions to image trajectories of the hands or feet so that we can predict the former from the latter. Here, we propose to learn a Gaussian Process mapping [Rasmussen 06] from the space of image trajectories to that of human motions represented as sequences of 3D poses, which can be done with a relatively small training database. Given this mapping, we can track the hands or feet of subjects in video sequences, infer plausible motions, and refine them to obtain accurate 3D pose estimates by minimizing an image-based objective function. In practice, however, the space of 3D pose sequences is too high-dimensional to be directly used for optimization purposes. Therefore, to reduce the dimensionality of our problem and the complexity of optimization, we use a linear subspace motion model [Urtasun 04, Sidenbladh 00] to represent 3D pose sequences with a manageable number of parameters, and learn a mapping from trajectory curvatures to these parameters.

In this section, we first introduce the motion representation we use. We then show how a Gaussian Process mapping can be learned between such motions and image trajectories from training data, and used to initialize poses in input video sequences. Finally, to make optimization practical, we introduce our linear subspace motion model.

5.1.1 Motion Representation

As described in Chapter 3, we rely on a coarse body model in which individual limbs are modeled as cylinders. A *motion* can be viewed as a time-varying pose. While pose varies continuously over time, we assume a discrete representation in which pose is sampled at N_t distinct time instants. In this way, a motion $\mathbf{y}$ is just a sequence of N_t discrete poses, and can be written as the $D = (N_j N_t + 6N_t)$-dimensional vector

$$\mathbf{y} = [\psi_1^T, \cdots, \psi_{N_t}^T, \mathbf{g}_1^T, \cdots, \mathbf{g}_{N_t}^T]^T \ , \tag{5.1}$$

where ψ_t is a set of N_j joint angles and $\mathbf{g}_t$ a 6D vector that defines the position and orientation of a reference body joint in a global coordinate system, as introduced in Chapter 3. Naturally, we assume that the temporal sampling rate is sufficiently high to interpolate the continuous pose signal. In our examples we split activities into short and temporally smooth motions. Therefore we simply consider poses as equally-spaced in time between the beginning and the end of a motion. This avoids the need to explicitly account for differences in speed between motions.

5.1.2 Gaussian Processes

Let $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_N]^T$ be the $N \times D$ matrix of N training motions from which the mean motion $\mathbf{y}_0$ was subtracted, and $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_N]^T$ the $N \times d$ matrix of corresponding d-dimensional image trajectories parameters. $\mathbf{Y}$ and $\mathbf{X}$ are said to be related through a Gaussian Process (GP) mapping [Rasmussen 06] if

$$\mathbf{y}_i = f(\mathbf{x}_i) + \epsilon_i \ , \tag{5.2}$$

where ϵ_i is zero-mean Gaussian noise, with a prior over f defined as

$$p(\mathbf{f}|\mathbf{X}) = \mathcal{N}(0, \mathbf{K}) \ , \tag{5.3}$$

where $\mathbf{f} = [f(\mathbf{x}_1)^T, \cdots, f(\mathbf{x}_N)^T]^T$, and $\mathbf{K}$ is a kernel matrix whose elements are defined by a covariance function, k, such that $\mathbf{K}_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$. This matrix entirely defines the GP, and only depends on hyperparameters Θ. In practice, we take a covariance function that is the sum of an RBF, a bias, and a noise term:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \theta_1 \exp(-\frac{\theta_2}{2} \|\mathbf{x}_i - \mathbf{x}_j\|^2) + \theta_3 + \frac{\delta_{\mathbf{x}_i,\mathbf{x}_j}}{\theta_4}, \tag{5.4}$$

where $\Theta = \theta_1, \theta_2, \theta_3, \theta_4$ are the hyperparameters that govern the output variance, the RBF support width, the bias, and the variance of the additive noise, respectively.

Learning a GP is then done by maximizing $p(\mathbf{Y} \mid \mathbf{X}, \Theta)\, p(\Theta)$ with respect to Θ, where

$$\begin{aligned} p(\mathbf{Y} \mid \mathbf{X}, \Theta) \ &= \\ & \frac{1}{\sqrt{(2\pi)^{ND}|\mathbf{K}|^D}} \exp\left(-\frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^T\right)\right) \ , \end{aligned} \tag{5.5}$$

and $p(\Theta)$ is a simple prior on the kernel parameters.

Given an input video sequence from which we can extract trajectory parameters $\mathbf{x}'$, the function $f(\mathbf{x}')$ follows a Gaussian distribution $p(f(\mathbf{x}')|\mathbf{X}, \mathbf{Y}, \Theta) = \mathcal{N}(\mu, \sigma)$, with

$$\mu(\mathbf{x}') \ = \ \mathbf{y}_0 + \mathbf{Y}^T\mathbf{K}^{-1}\mathbf{k}(\mathbf{x}') \ , \tag{5.6}$$

$$\sigma^2(\mathbf{x}') \ = \ k(\mathbf{x}', \mathbf{x}') - \mathbf{k}(\mathbf{x}')^T\mathbf{K}^{-1}\mathbf{k}(\mathbf{x}') \ , \tag{5.7}$$

where $\mathbf{k}(\mathbf{x}')$ is the vector with elements $k(\mathbf{x}', \mathbf{x}_j)$ for latent positions $\mathbf{x}_j \in \mathbf{X}$. We can therefore simply use the mean prediction of the model $\mu(\mathbf{x}')$ to initialize the motion in the new sequence, and refine it via optimization of an image-based objective function, as will be explained in Section 5.1.4.

5.1.3 Linear Subspace Motion Model

Since, in practice, optimizing an image-based criterion with respect to the $N_j N_t + 6N_t$ degrees of freedom of a sequence of poses is intractable, we first reduce the dimensionality of this space. To this end, we perform Principal Component Analysis on the dataset $\mathbf{Y}$ to find a low-dimensional basis with which we can effectively model the motion, as described in Chapter 3, specifically by Equation 3.5.

In this case the scalar coefficients $\{\alpha_i\}$ characterize the motion, and the dimensionality $N_m \leq N_j N_t + 6N_t$ controls the fraction of the total variance of the training data that is captured by the subspace. This is measured by

$$Q(N_m) \;=\; \frac{\sum_{i=1}^{N_m} \lambda_i}{\sum_{i=1}^{N_j N_t + 6N_t} \lambda_i} \;, \tag{5.8}$$

where λ_i are the eigenvalues of the data covariance matrix, ordered such that $\lambda_i \geq \lambda_{i+1}$. In practice, we choose N_m such that $Q(N_m) > 0.9$. Finally, the GP mapping is learned from the trajectory curvatures to the parameters α_i of our training data rather than to the sequence of poses directly. Since we ensure that 90 $\%$ of the training data is modeled by the linear subspace, this only yields a negligible loss of accuracy.

5.1.4 Fitting the Model to Image Data

We now describe the process of estimating the body pose for an input video sequence. The whole runtime procedure is summarized in Figure 5.1. Given an input sequence, we can easily extract the trajectory parameters $\mathbf{x}'$ as will be described in Section 5.2.2. From these, we compute the mean prediction $\mu(\mathbf{x}')$ through our GP model and use it to initialize the eigen-motion coefficients α'. Because, in our examples, most global motion parameters $\mathbf{g}_1, \ldots, \mathbf{g}_{N_t}$ either can be computed from the feet trajectories or remain constant, the linear subspace decomposition is only performed on the joint angles. Only two global orientations need to be estimated from the images, which we do by considering them as unknowns in the first and last frames, and linearly interpolating them in between.

The objective function is then computed as a binary *AND* between the silhouette obtained by background subtraction performed on the input images and the reprojection of our cylinder-based body model in the estimated poses. Our method is robust to very low-quality silhouettes, as the ones shown in Figure 5.2.

At this point the objective function needs to be optimized over the α' parameters that, through the PCA mapping, define the body pose in the full sequence. Since the motion in the images is of arbitrary length, we just warp the one we obtain with the eigen-motion coefficients to fit the correct number of frames through a simple linear interpolation of the N_j joint angles defining a pose. Such α' parameters are much lower dimensional than the full pose sequence space which, as reminded before, has $N_j N_t + 6N_t$ dimensions. The dimensionality of α' is usually 10 in our experiments, which makes the problem manageable, but still requires an advanced optimization strategy.

Given the complexity of the objective function we had to adopt a discrete optimization technique. A standard particle filtering algorithm [Isard 98] has been tested and has given good results, such as the ones that are provided in the following section. Unfortu-

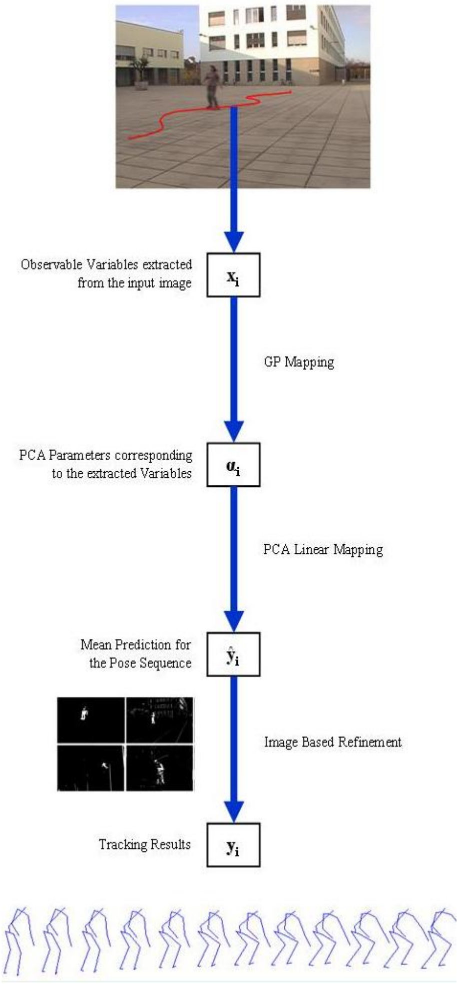

Observable Variables extracted from the input image
$\mathbf{x_i}$
GP Mapping
PCA Parameters corresponding to the extracted Variables
α_i
PCA Linear Mapping
Mean Prediction for the Pose Sequence
$\hat{\mathbf{y}}_i$
Image Based Refinement
Tracking Results
$\mathbf{y_i}$

Figure 5.1: Overview of the runtime procedure to recover the body pose, explained in Section 5.1.4.

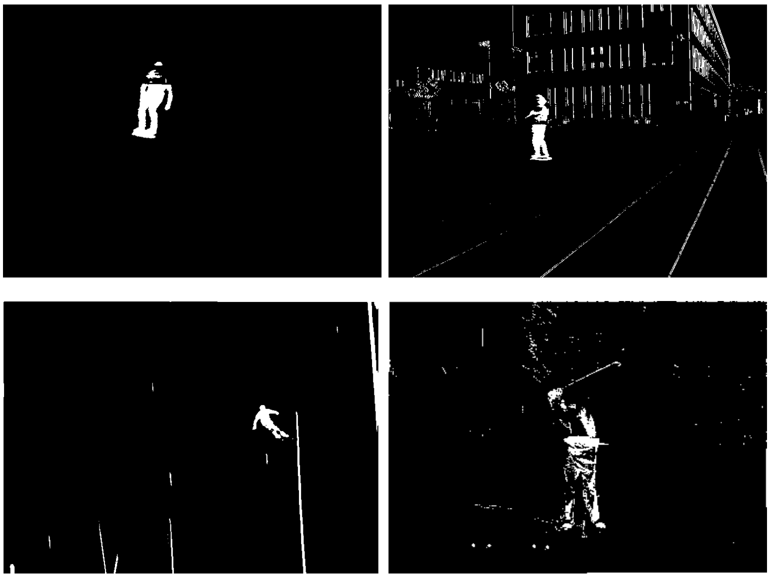


Figure 5.2: Input silhouettes used to compute our results. The silhouettes were extracted using a standard background subtraction technique on skating and golfing examples, while on skiing an intensity threshold was instead applied.

nately, it has the drawback of producing a quite slow optimization, given that the particles are thrown in a 10-dimensional space and due to this are quite spread and dispersed around the initial estimate, according to the eigenvalues λ_i. Note that, for this purpose, we could equivalently have used the variance $\sigma^2(\mathbf{x}')$ of Eq. 5.7. Accordingly, the number of particles needs to be very large, in order to cover well such a high-dimensional space, but given the fact that one function evaluation requires the rendering of the whole sequence, this has a strong influence on the speed of the global optimization.

For this reasons we have decided to also do some experiments with a different discrete maximization strategy, namely Powell's Method [Powell 64]. It is a well-known and powerful technique that allows to minimize (or maximize, as in our case) an objective function without the need of computing its partial derivatives. As shown in Fig. 5.3,

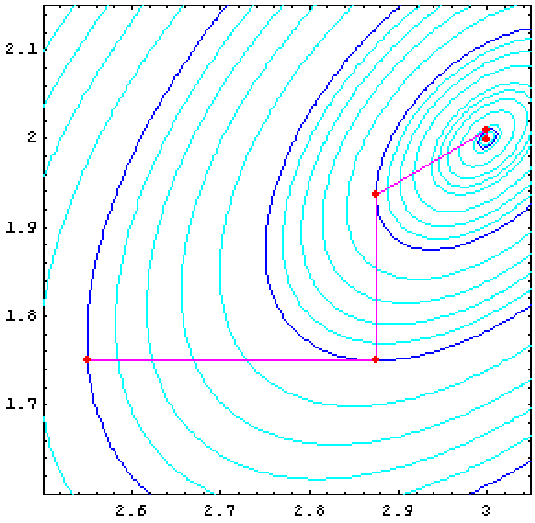


Figure 5.3: Powell's Method. The figure shows one iteration of Powell's Method on a sample 2-dimensional function.

it basically consists of several iterations: In each of these iterations the function is first optimized one dimension at a time through direct search, thus obtaining a partial solution point. Then a second optimization is performed along the vector that connects the initialization point and this partial solution. In the end the adoption of the Powell's Method for optimization brought a relevant speed-up in the computation time needed to obtain a solution ($\sim 50\%$ of improvement, from ~ 3 minutes to ~ 2 minutes for a sequence) of the same quality as with the standard particle filtering technique.

5.2 Experimental Results

To demonstrate the effectiveness of our approach, we applied it to three very different kinds of motion: Walking, roller skating and golfing. Furthermore, to show that our models generalize over the training data, we used the skating model to recover the motion of a skier.

In this section, we first describe our training data, next we explain how we obtain the values of latent variables x from sequences of images and finally we present our tracking results.

5.2.1 Obtaining Training Data

To obtain the training sequences of 3D poses, we used a commercial optical motion capture system that recovers the positions of reflective markers placed on the joints of a person using six infrared cameras [h]. The body model we used has $N_j = 51$ degrees of freedom corresponding to the joint angles, and the length N_t of the normalized motions varies from 5 for walking steps, to 15 for golf and skating sequences.

- In the walking case, we asked a subject to walk around for a few minutes, making smooth and sharp changes of direction at varying speeds. We then split the collected data into single steps, each one normalized to N_t poses, thus building vectors of size $N_j N_t$. Moreover we inferred the ground trajectory from all steps, by taking the average between the ground projections of the feet. These trajectories were finally used to compute the $\mathbf{x}$ trajectory parameters, every time using as reference the starting point and tangent of the step trajectory depicted by 5.4(a). The 4-dimensional embedded representation includes the curvature of the trajectory, the latitudinal and longitudinal displacement of the subject, and a binary variable indicating which foot is moving. Globally, we used 200 training points, corresponding to a step each.

- In the skating case, we captured a subject performing turns of varying radii. We then split the reconstructed sequences into small motions representing half a turn each and time-normalized these subsequences to build vectors of length $N_j N_t$ by concatenating N_t poses of N_j joint angles. For each one of these vectors, we computed the trajectory of the feet on the ground plane to which we fitted a second order polynomial, which was enough to optain a simple parameterization. In fact this yielded a 2-dimensional vector $\mathbf{x}$ of trajectory parameters containing the curvature of the half-turns and a parameter discriminating between the two halves of a turn, as depicted by 5.4(a). In total we used 84 training points, each of them representing a half-turn.

- In the golf case, the database contained several golf swings, each of which was normalized to a standard length N_t, thus yielding similar training example vectors of length $N_j N_t$. We used the hands' trajectory to compute our trajectory parameters $\mathbf{x}$. Since the 3D hands' trajectory cannot easily be retrieved from single-view sequences, we considered the trajectory in the image plane. Therefore, for each new sequence, we built the set of 2D hand trajectories corresponding to all the motions in our database projected to the same viewpoint as the sequence, which is straightforward given the camera calibration. We then fitted piecewise polynomials to the 2D trajectories, which yielded a 3-dimensional latent representation $\mathbf{x}$. The total number of training points, and therefore swings, in this case was 40.

5.2.2 Retrieving Trajectory Parameters

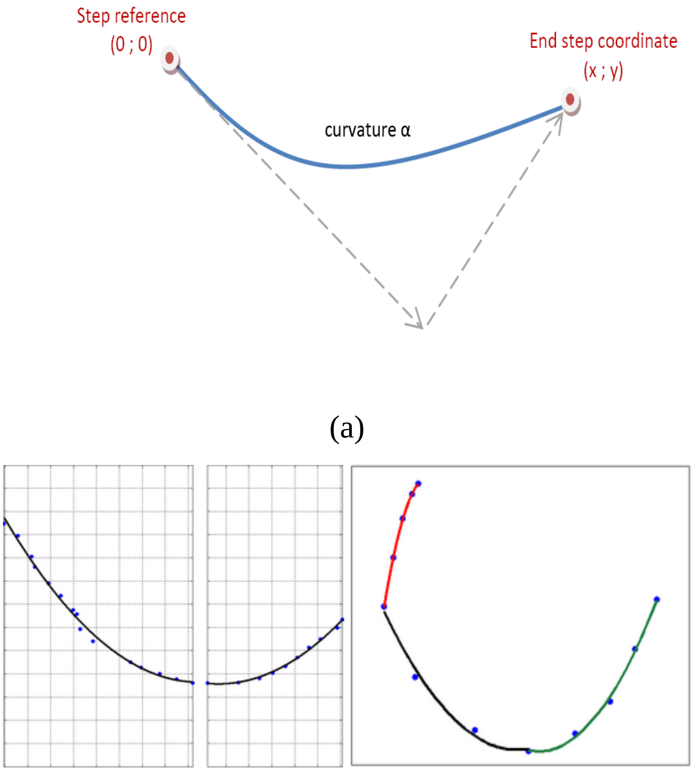
For new sequences in which we want to infer the poses, we first need to recover the trajectory parameters $\mathbf{x}'$. To this end, we track either the feet or the hands using a standard image correlation measure, after a manual initialization of the chosen target in the first frame of the sequence.

In the case walking and skating, this is made more robust by introducing knowledge of where the ground plane is. This yields feet trajectories on a 2D rectified plane, which can be automatically split into steps or half-turns, depending on the type of motion. We then obtain the latent parameters corresponding to the half-turns and steps by fitting a polynomial to the trajectories in the same way as for the training data, as depicted in Figures 5.4(a) and (b).

For golf swings, tracking the hands can be made robust by also tracking the golf club, as proposed in [Gehrig 03]. Since the trajectory parameters for the training sequences are estimated in the image plane using the same camera as in the test sequence, we can directly fit the piecewise polynomial to the hand trajectories to obtain $\mathbf{x}'$, as shown in Figure 5.4(c).

5.2.3 Motion Recovery

We present our tracking results obtained from real sequences in which we initialized the motion with the mean prediction of the Gaussian Process model given the trajectory parameters computed as mentioned above. We show results obtained for skating, skiing, walking and golfing.



(a)

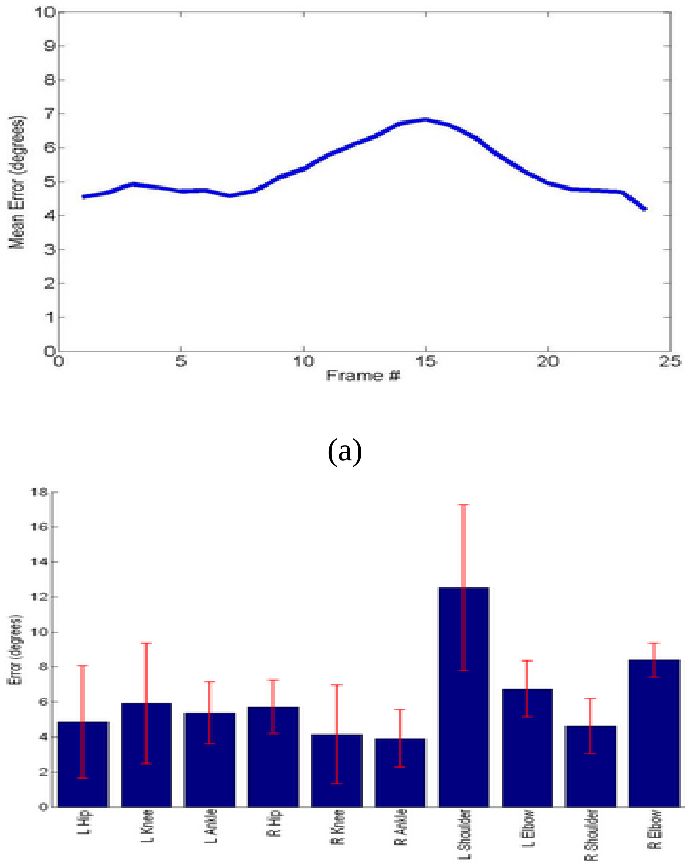
(b) (c)

Figure 5.4: (a) For walking, we use a step low dimensional parameterization, including curvature of the trajectory, x and y displacement and a binary variable indicating which foot is moving. (b) For skating, we show two consecutive skating motions that correspond to the test sequence of Figure 5.6. The blue dots represent the tracked average feet locations on the ground plane and the black line is a second order polynomial fitted to them. The underlying grid is composed of 20cm×20cm squares. The first latent parameter is the curvature of the polynomial, whose sign changes if the subject is turning left or right. The second one is a binary variable indicating if the subject is in the first or second half of the turn. (c) For golfing, we use hands' motion, corresponding to the second golfing sequence of Figure 5.11. The blue dots depict the tracked hand locations, while the 3 lines show the polynomials fitted to the different phases of the swing, whose 3 curvatures are the latent parameters.

To obtain a quantitative evaluation of our results, we filmed some of the motion captured skating sequences. We then removed one sequence from the training data to adopt a leave-one-out validation scheme. We applied our algorithm to this sequence, and measured the reconstruction error as the average of the absolute error over the N_j joint angles that define a pose. This error is plotted frame-wise in Figure 5.5(a), and has an average value of 5.3 degrees over 24 frames, with a standard deviation of 0.8 degrees. This number of frames corresponds to the time during which the subject was within the capture volume of the Vicon system. In Figure 5.5(b) we plot the errors for different joint angles, averaged throughout the sequence. We achieve better accuracy on the lower part of the body than on the upper part, because it is much better constrained by the feet trajectory. We show the retrieved pose, both reprojected in the input image and seen from a different viewpoint, in Figure 5.6.

This ground-truth data also helped us in computing how much accuracy is brought by the refinement step: Without it, the above mean error would have been 6.4 degrees, with a standard deviation of 1.1 degree. Moreover we also made some experiments without using the observable variables to initialize the PCA motion model, in order to compare our approach to [Urtasun 04]. In such paper the PCA weights were all initialized to zero, and doing this on our skating sequence would lead to a mean error of 10.7 degrees with a standard deviation of 1.8 degrees.

To demonstrate that our approach also works in non studio-like environments, we filmed the outdoor sequence of Figure 5.7 in which the skater is not the one we captured to train the GP. The viewpoint is also very different to show that our approach, being fully 3D, is totally view-independent. Note that the reprojections of our skeleton model



(a)

(b)

Figure 5.5: (a) Average frame-wise error for the sequence of Figure 5.6, in degrees. (b) Mean errors for different joint angles, in degrees, averaged throughout the sequence. The bars represent the standard deviations of the errors.

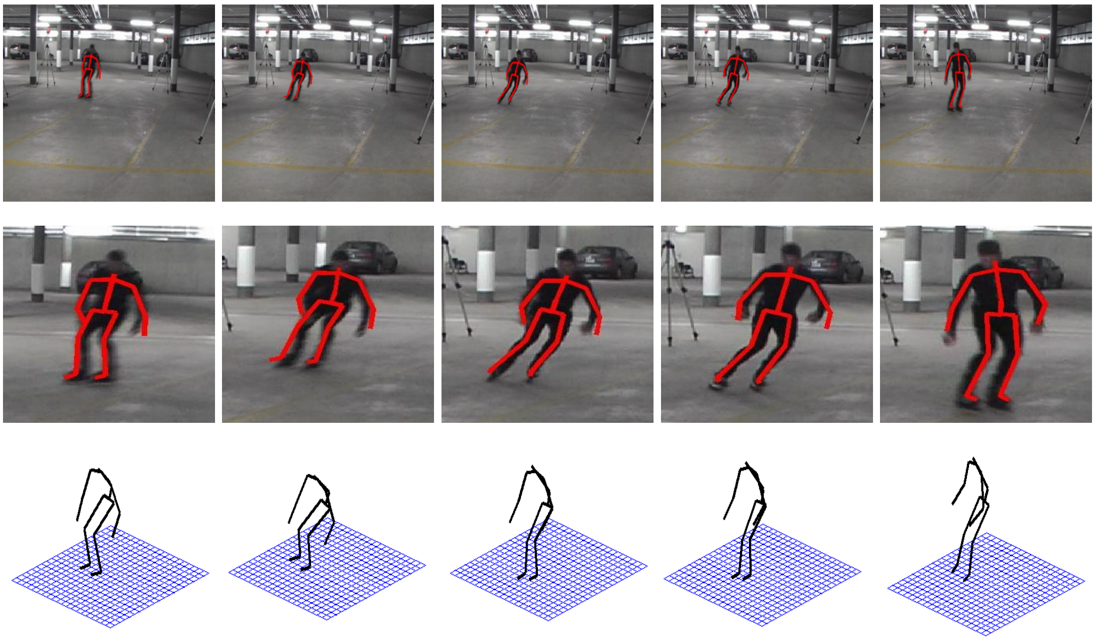


Figure 5.6: Roller skating in a studio setup. **First row:** We reprojected the recovered body poses in the input images. **Second row:** Zoomed version of the first row. **Third row:** To highlight the 3D nature of the results, we display the 3D skeleton seen from a different viewpoint.

correspond well to the underlying images.

Finally, since the skiing motion is very similar to the skating one, we applied our GP trained for skating on the skiing sequence of Figure 5.8 in which a subject is slaloming between gates. Of course modeling the ground plane on a ski slope is not straightforward. We therefore selected a part of the slalom track that could be roughly approximated by a plane. We then used the GPS coordinates of the gates to warp the 2D trajectory to an orthogonally rectified one, in which we could compute the latent parameters. To this end it would have been enough to have a 3D reference on the ground plane. The results we have obtained are encouraging, but can only be evaluated qualitatively. Nevertheless, they highlight our method's ability to generalize over the learned motion.

For the walking case, we present some results obtained on a subject walking along a curvilinear trajectory in Figures 5.9 and 5.10. To demonstrate the ability of the algorithm to generalize over the body shape, the tracked subject is different from the subject that was used to generate the training database. These results were computed to show that the whole framework can manage very different types of motions, while we still believe that the technique described in Chapter 4 works better in terms of robustness and accuracy in cases where key poses are frequent and relatively easy to detect, as is the case during walking.

In the case of golf, we show the results obtained when tracking two different subjects, whose motion was *not* captured to build our database, performing a swing. These results are depicted by Figure 5.11, both overlaid on the input images and seen from a different viewpoint. As can be noticed, the arms motion constrains better the upper body than the legs, which is intuitive.

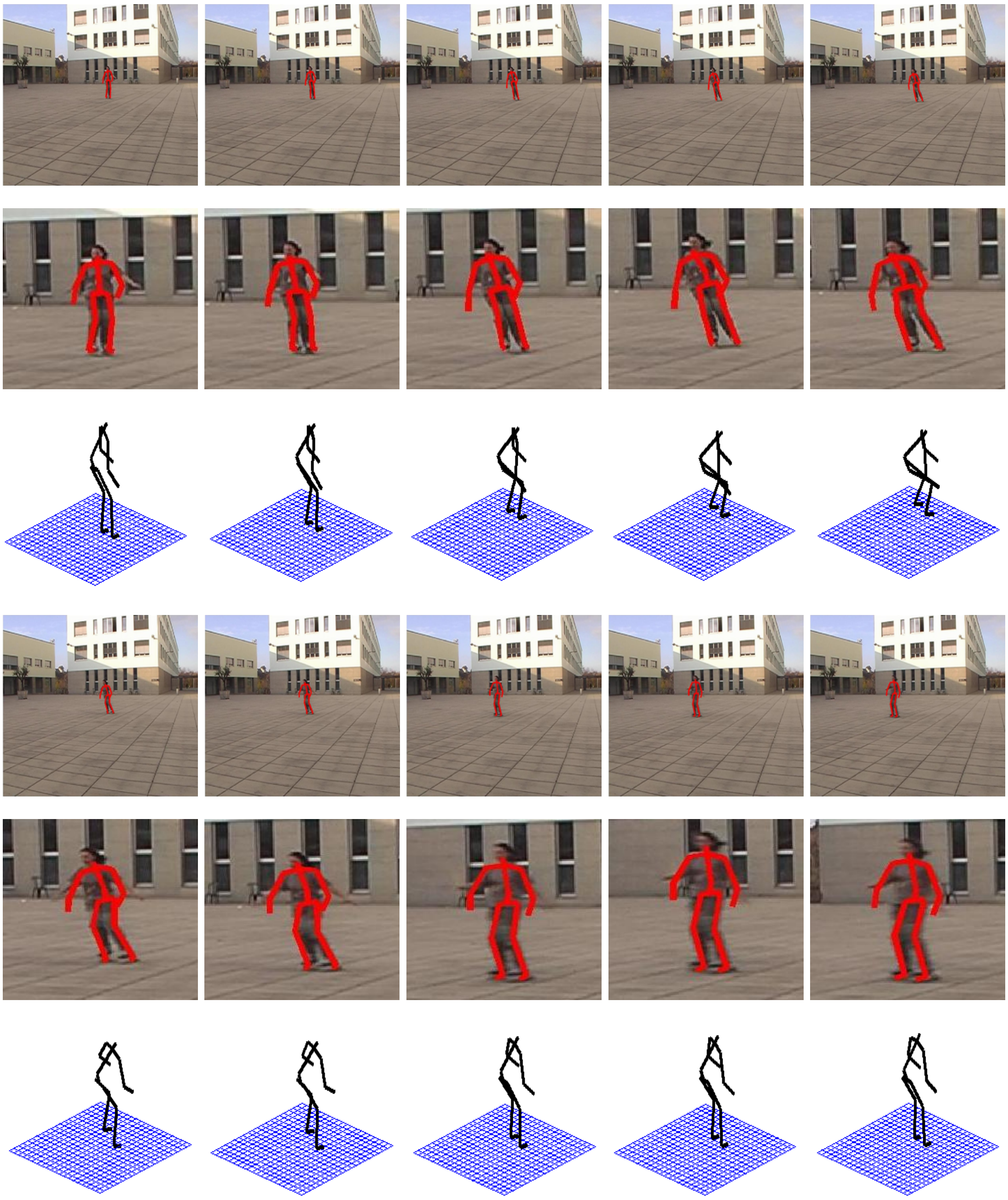


Figure 5.7: Roller skating. **First and fourth rows:** Recovered body pose reprojected in the input image. **Second and fifth rows:** Zoomed versions of the first and fourth rows, respectively. **Third and sixth rows:** 3D skeleton of the subject seen from a different viewpoint.

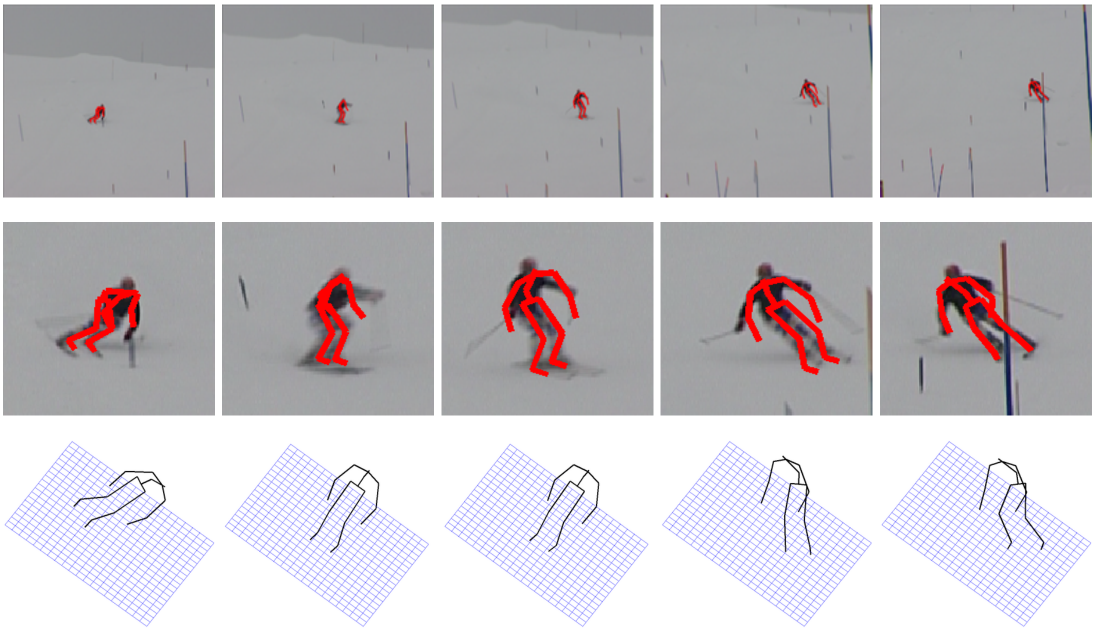


Figure 5.8: We used the model trained on skating motions to recover a skiing one. **First row:** Recovered body pose reprojected in the input image. **Second row:** Zoomed versions of the first row. **Third row:** 3D skeleton of the subject seen from a different viewpoint.

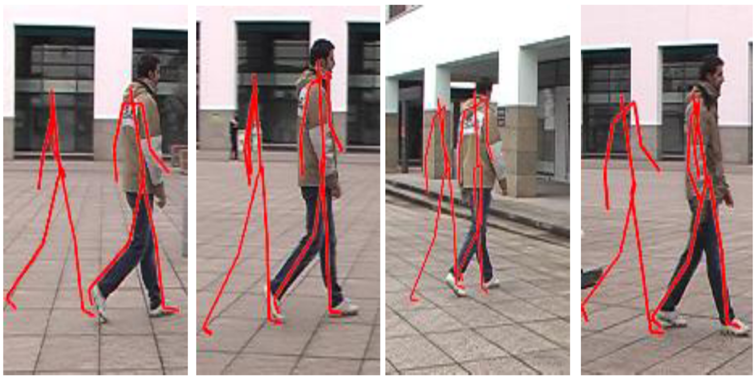


Figure 5.9: Recovered pose in the walking case for 4 steps. The resulting estimated pose is shown in red for the first and the last frame of each step. In the case of the last step it is projected over the walking subject.

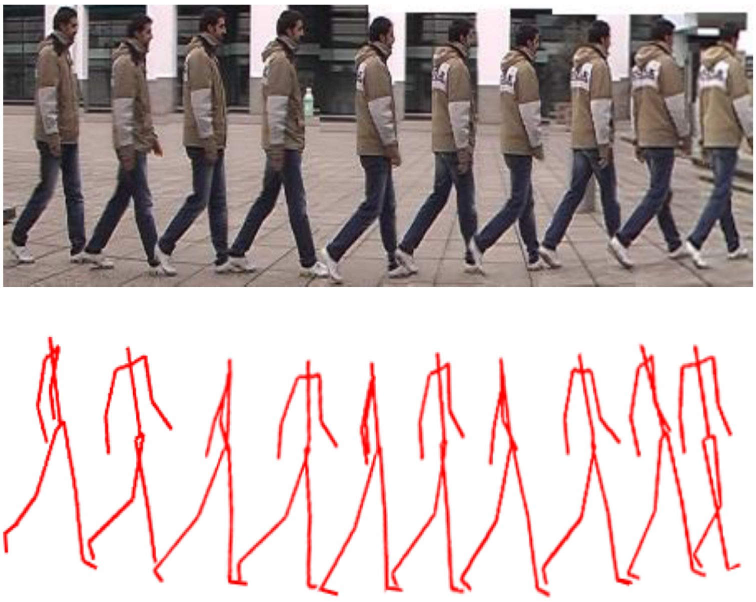


Figure 5.10: Estimated pose for the whole curvilinear walking trajectory. The top image shows the input sequence while the bottom image displays the corresponding body poses.

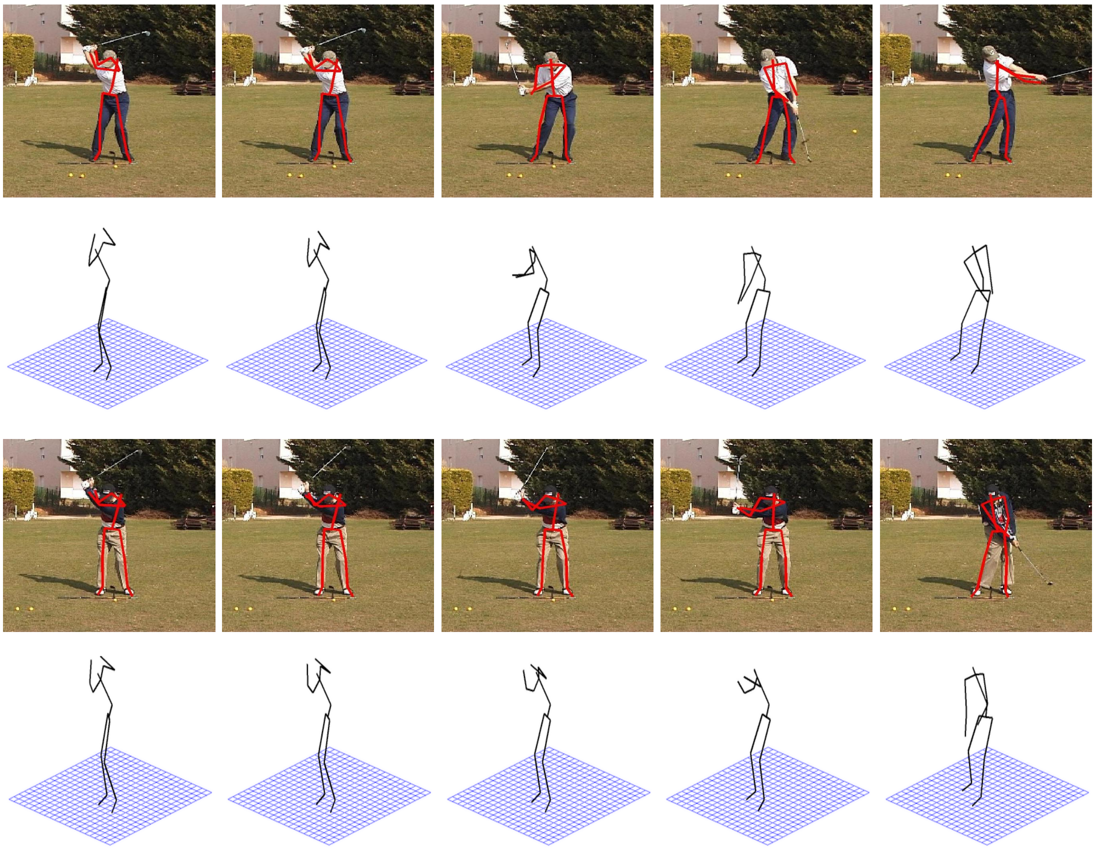


Figure 5.11: Golf swing tracking. **First and third rows:** Two different subjects performing a golf swing. The recovered body poses have been reprojected in the input images. **Second and fourth rows:** The 3D skeleton of the person is seen from a different viewpoint.

Chapter 6

Conclusions

In this thesis we have demonstrated that retrieving the 3–D human body configuration starting from monocular input is a very challenging task, but nonetheless it can be efficiently tackled with the help of prior information and adequate techniques.

Two related approaches have been presented, that can handle two different types of motions: For motions that contain characteristic postures that are relatively easy to detect, such as walking and golfing, the algorithm presented in Chapter 4 exploits this fact to formulate 3–D motion recovery from a single video sequence as an interpolation problem. This is much easier to achieve than open-ended tracking and we have shown that it can be solved using straightforward minimization. This approach is generic because most human motions also feature canonical poses that can be easily detected. This is significant because it means that we can focus our future efforts on developing methods to reliably detect these canonical poses instead of all poses, which is much harder. A limitation of this approach is that it does not handle transitions from one activity to another, as Markovian motion models could. However, since transitions typically also involve key-poses, the approach could potentially be extended to this much more demanding context given a sufficiently rich training database. This would involve choosing which motion model to use to connect these key poses and modeling the transition probabilities between activities, and is a topic for future research.

Furthermore, the approach proposed in Chapter 5 can handle motions in which such key-poses are not defined, but there is still a clear relation between some easily measurable image quantities and the body configuration, as for example skating where the trajectory followed by a subject is highly correlated to how the subject articulates. Our technique uses these easily retrievable image measurements as latent variables from which we can recover 3D human body motion via a Gaussian Process mapping. By contrast with state-of-the-art approaches that consider the latent variables as unknowns, learning our mapping involves very few parameters and is therefore much easier to do. It allows us to recover 3D motion from monocular video sequences without having to manually initialize either the poses or the latent variables. We have demonstrated this approach on challenging activities such as roller skating, skiing, and golfing. A potential extension would be to look into more complex activities for which some of the latent variables are indeed observable and others not. In these cases, such as when the person's individual style truly matters, we will look at hybrid approaches where we will establish a first mapping using the approach presented here and then learn a second mapping modeling deviations from what the first predicts. Because the first mapping will have captured much of the complexity, it is hoped that the second will be easy to learn, even in these difficult cases.

A general drawback of the proposed approaches is that they rely on the underlying motion model, and therefore the reconstructed motions cannot be too different from the ones that build the training dataset. This is also due to the fact that the chosen dimensionality reduction technique, namely PCA, is intrinsically linear and not suitable to handle motions that are too complex or that differ too much from the training ones. As explained in Chapter 3, the choice of such technique comes from the trade-off between complexity and model elasticity, and has anyway proved itself enough to cope with the types of motion proposed in this thesis, even in a very ill-constrained setup. Another issue that affects the proposed techniques is related to optimization and computation time. Given the high nonlinearity of the studied objective function, standard gradient-based optimization strategies could not be used. For this reason we had to adopt different algorithms like stochastic optimization and Powell's method, which were still able to obtain good results but on the other hand required a lot of computation time, also due to the high-dimensionality of the analyzed functions.

Bibliography

[a] *Ascension MotionStar magnetic Motion Capture system.* Burlington, VT, USA. http://www.ascension-tech.com/.

[b] *Meta Motion electro-mechanical motion capture.* Meta Motion, San Francisco, CA, USA. http://www.metamotion.com/.

[c] *Motion analysis Motion Capture Systems.* Santa Rosa, CA, USA. http://www.motionanalysis.com/.

[d] *Xsens MVN Interial Motion Capture System.* Xsens, Enschede, The Netherlands. http://www.xsens.com/.

[e] *IMPULSE Motion Capture System, Phase Space.* San Leandro, CA, USA. http://www.phasespace.com/.

[f] *Physilog, a portable datalogger for long term recording. LMAM, EPFL, Switzerland.* http://lmam.epfl.ch/page2842.html.

[g] *Polhemus Liberty Electromagnetic Tracker*. Polhemus, Colchester, VT, USA. http://www.polhemus.com/.

[h] *Vicon Motion Capture Systems.* http://www.vicon.com/.

[Agarwal 04] A. Agarwal & B. Triggs. *3D Human Pose from Silhouettes by Relevance Vector Regression.* In Conference on Computer Vision and Pattern Recognition, 2004.

[Andriluka 08] M. Andriluka, S. Roth & B. Schiele. *People-Tracking-by-Detection and People-Detection-by-Tracking*. In Conference on Computer Vision and Pattern Recognition, Anchorage, Alaska, June 2008.

[Andriluka 09] M. Andriluka, S. Roth & B. Schiele. *Pictorial Structures Revisited: People Detection and Articulated Pose Estimation.* In Conference on Computer Vision and Pattern Recognition, Miami, FL, June 2009.

[Anguelov 05] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers & James Davis. *SCAPE: shape completion and animation of people*. ACM SIGGRAPH, vol. 24, pages 408–416, 2005.

[Balan 08] A.O. Balan & M.J. Black. *The Naked Truth: Estimating Body Shape Under Clothing*. In European Conference on Computer Vision, pages II: 15–29, 2008.

[Barr 84] A. H. Barr. *Local and global deformations of solid primitives*. ACM SIGGRAPH, pages 21–30, September 1984.

[Bo 08] L. Bo, C. Sminchisescu, A. Kanaujia & D. Metaxas. *Fast Algorithms for Large Scale Conditional 3D Prediction*. In Conference on Computer Vision and Pattern Recognition, Anchorage, AK, June 2008.

[Bregler 98] C. Bregler & J. Malik. *Tracking People with Twists and Exponential Maps*. In Conference on Computer Vision and Pattern Recognition, Santa Barbara, CA, June 1998.

[Brubaker 06] M. Brubaker, A. Hertzmann & D. Fleet. *Physics-Based Human Pose Trackng*. In NIPS Workshop on Evaluation of Articulated Human Motion and Pose Estimation (EHuM), 2006.

[Brubaker 07] M.A. Brubaker, D.J. Fleet & A. Hertzmann. *Physics-Based Person Tracking Using Simplified Lower-Body Dynamics*. In Conference on Computer Vision and Pattern Recognition, pages 1–8, 2007.

[Brubaker 08] M.A. Brubaker & D.J. Fleet. *The Kneed Walker for Human Pose Tracking*. In Conference on Computer Vision and Pattern Recognition, 2008.

[Carranza 03] J. Carranza, C. Theobald, C. Magnor & H. P. Seidel. *Free-viewpoint video of human actors*. In ACM SIGGRAPH, 2003.

[Cheung 03] G. Cheung, S. Baker & T. Kanade. *Shape-From-Silhouette of Articulated Objects and its Use for Human Body Kinematis Estimation and Motion Capture*. In Conference on Computer Vision and Pattern Recognition, pages 569–577, Madison, WI, July 2003.

[Choo 01] K. Choo & D.J. Fleet. *People Tracking Using Hybrid Monte Carlo Filtering*. In International Conference on Computer Vision, Vancouver, Canada, July 2001.

[Chu 03] C. W. Chu, O. C. Jenkins & M. J. Mataric. *Markerless kinematic model capture from volume sequences.* In Conference on Computer Vision and Pattern Recognition, pages 475–482, Madison, WI, July 2003.

[Davison 01] A. J. Davison, J. Deutscher & I. D. Reid. *Markerless Motion Capture of Complex Full-Body Movement for Character Animation.* In Eurographics Workshop on Computer Animation and Simulation. Springer-Verlag LNCS, 2001.

[Delamarre 99] Q. Delamarre & O. Faugeras. *3D Articulated Models and Multi-View Tracking with Silhouettes*. In International Conference on Computer Vision, Corfu, Greece, September 1999.

[Deutscher 00] J. Deutscher, A. Blake & I. Reid. *Articulated Body Motion Capture by Annealed Particle Filtering.* In Conference on Computer Vision and Pattern Recognition, pages 2126–2133, Hilton Head Island, SC, 2000.

[Dewaele 04] G. Dewaele, F. Devernay & R. Horaud. *Hand Motion from 3D Point Trajectories and a Smooth Surface Model*. In European Conference on Computer Vision, pages 495–507, May 2004.

[DiFranco 01] D.E. DiFranco, T.J. Cham & J.M. Rehg. *Reconstruction of 3-D Figure Motion from 2-D Correspondences.* In Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii, December 2001.

[Dimitrijevic 06] M. Dimitrijevic, V. Lepetit & P. Fua. *Human Body Pose Detection Using Bayesian Spatio-Temporal Templates*. Computer Vision and Image Understanding, vol. 104, no. 2-3, pages 127–139, 2006.

[E.-J-Ong 06] E.-J-Ong, A. S. Micilotta, R. Bowden & A. Hilton. *Viewpoint Invariant Exemplar-Based 3–D Human Tracking*. Computer Vision and Image Understanding, vol. 104, no. 2–3, pages 178–189, 2006.

[Elgammal 04] A. Elgammal & C.S. Lee. *Inferring 3D Body Pose from Silhouettes using Activity Manifold Learning*. In Conference on Computer Vision and Pattern Recognition, Washington, DC, June 2004.

[Enzweiler 08] M. Enzweiler & D.M. Gavrila. *A Mixed Generative-Discriminative Framework for Pedestrian Classification.* In Conference on Computer Vision and Pattern Recognition, Anchorage, AK, June 2008.

[Ess 08] A. Ess, B. Leibe, K. Schindler & L. van Gool. *A mobile vision system for robust multi-person tracking.* In Conference on Computer Vision and Pattern Recognition, 2008.

[Felzenszwalb 00] P.F. Felzenszwalb & D.P. Huttenlocher. *Efficient Matching of Pictorial Structures*. In Conference on Computer Vision and Pattern Recognition, 2000.

[Forsyth 97] D.A. Forsyth & M.M. Fleck. *Body plans*. In Conference on Computer Vision and Pattern Recognition, June 1997.

[Fossati 07] A. Fossati, M. Dimitrijevic, V. Lepetit & P. Fua. *Bridging the Gap between Detection and Tracking for 3D Monocular Video-Based Motion Capture*. In Conference on Computer Vision and Pattern Recognition, Minneapolis, MI, June 2007.

[Fossati 08] A. Fossati & P. Fua. *Linking Pose and Motion*. In European Conference on Computer Vision, Marseille, France, October 2008.

[Franco 05] J. S. Franco & E. Boyer. *Fusion of multi-view silhouette cues using a space occupancy grid*. In International Conference on Computer Vision, Beijing, China, October 2005.

[Gall 10] J. Gall, B. Rosenhahn, T. Brox & H. P. Seidel. *Optimization and Filtering for Human Motion Capture*. International Journal of Computer Vision, vol. 87, no. 1, pages 75–92, March 2010.

[Gammeter 08] S. Gammeter, A. Ess, T. Jaeggli, K. Schindler, B. Leibe & L. Van Gool. *Articulated Multi-Body Tracking under Egomotion*. In European Conference on Computer Vision, Marseille, France, 2008.

[Gavrila 96] D.M. Gavrila & L. Davis. *3D Model-based Tracking of Humans in Action : A Multi-View Approach*. In Conference on Computer Vision and Pattern Recognition, pages 73–80, San Francisco, CA, June 1996.

[Gavrila 99] D. Gavrila & V. Philomin. *Real-Time Object Detection for "Smart" Vehicles*. In International Conference on Computer Vision, pages 87–93, 1999.

[Gehrig 03] N. Gehrig, V. Lepetit & P. Fua. *Golf Club Visual Tracking for Enhanced Swing Analysis Tools*. In British Machine Vision Conference, Norwich, UK, September 2003.

[Giebel 04] J. Giebel, D.M. Gavrila & C. Schnorr. *A Bayesian Framework for Multi-Cue 3D Object Tracking*. In European Conference on Computer Vision, 2004.

[Ham 06] Jihun Ham, Ikkjin Ahn & D. Lee. *Learning a manifold-constrained map between image sets: applications to matching and pose estimation*. Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, vol. 1, pages 817–824, June 2006.

[Hartley 00] R. Hartley & A. Zisserman. Multiple View Geometry in Computer Vision. Cambridge University Press, 2000.

[Herda 04] L. Herda, R. Urtasun & P. Fua. *Hierarchical Implicit Surface Joint Limits to Constrain Video-Based Motion Capture*. In European Conference on Computer Vision, Prague, Czech Republic, May 2004.

[Horaud 09] R. Horaud, M. Niskanen, G. Dewaele & E. Boyer. *Human Motion Tracking by Registering an Articulated Surface to 3D Points and*

Normals. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 31, no. 1, pages 158–163, Jan. 2009.

[Howe 04] N. R. Howe. *Silhouette lookup for automatic pose tracking*. In Workshop on Articulated and Non-Rigid Motion, 2004.

[Isard 98] M. Isard & A. Blake. *CONDENSATION – Conditional Density Propagation for Visual Tracking*. International Journal of Computer Vision, vol. 1, pages 5–28, 1998.

[Isard 01] M. Isard & J. MacCormick. *BraMBLe: a Bayesian multiple-blob tracker*. In Conference on Computer Vision and Pattern Recognition, volume 2, pages 34–41, July 2001.

[Kakadiaris 00] I.A. Kakadiaris & D. Metaxas. *Model-Based Estimation of 3D Human Motion*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 12, December 2000.

[L. Bourdev 09] L. Bourdev & J. Malik. *Poselets: Body Part Detectors Trained Using 3D Human Pose Annotations*. In International Conference on Computer Vision, 2009.

[Lawrence 04] N. D. Lawrence. *Gaussian Process Models for Visualisation of High Dimensional Data*. In Neural Information Processing Systems. MIT Press, Cambridge, MA, 2004.

[Leibe 05] B. Leibe, E. Seemann & B. Schiele. *Pedestrian Detection in Crowded Scenes.* In Conference on Computer Vision and Pattern Recognition, volume 1, San Diego, CA, June 2005.

[Li 06] R. Li, M. Yang, S. Sclaroff & T. Tian. *Evaluation of 3D Human Motion Tracking with a Coordinated Mixture of Factor Analyzers.* In NIPS Workshop on Evaluation of Articulated Human Motion and Pose Estimation (EHuM), 2006.

[Loy 04] G. Loy, M. Eriksson, J. Sullivan & S. Carlsson. *Monocular 3D Reconstruction of Human Motion in Long Action Sequences*. In European Conference on Computer Vision, 2004.

[Mikic 03] I. Mikic, M. Trivedi, E. Hunter & P. Cosman. *Human Body model acquisition and tracking using voxel data*. International Journal of Computer Vision, vol. 53, no. 3, pages 199–223, 2003.

[Mitchelson 03] J Mitchelson & A Hilton. *Hierarchical Tracking of Multiple People*. In Proceedings of the British Machine Vision Conference, 2003.

[Moeslund 00] T. Moeslund & E. Granum. *Multiple Cues used in Model-Based Human Motion Capture*. In Automated Face and Gesture Recognition, Grenoble, France, 2000.

[Moeslund 06] Thomas B. Moeslund, Adrian Hilton & Volker Krüger. *A survey of advances in vision-based human motion capture and analysis*. Comput. Vis. Image Underst., vol. 104, no. 2, pages 90–126, 2006.

[Mori 04] G. Mori, X. Ren, A.A. Efros & J. Malik. *Recovering Human Body Configurations: Combining Segmentation and Recognition*. In Conference on Computer Vision and Pattern Recognition, Washington, DC, 2004.

[Muendermann 07] L. Muendermann, S. Corazza & T. Andriacchi. *Accurately measuring human movement using articulated ICP with soft-joint constraints and a repository of articulated models*. In Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, June 2007.

[Navaratnam 07] R. Navaratnam, A. Fitzgibbon & R. Cipolla. *The Joint Manifold Model for Semi-supervised Multi-valued Regression*. In International Conference on Computer Vision, Rio, Brazil, October 2007.

[Odobez 97] J. Odobez & P. Bouthemy. *Separation of moving regions from background in an image sequence acquired with a mobile camera*. Video Data Compression for Multimedia Computing, pages 283–311, 1997.

[Olson 97] C. F. Olson & D. P. Huttenlocher. *Automatic target recognition by matching oriented edge pixels*. IEEE Transactions on Image Processing, vol. 6, pages 103–113, January 1997.

[Ormoneit 01] D. Ormoneit, H. Sidenbladh, M.J. Black & T. Hastie. *Learning and Tracking Cyclic Human Motion*. In Neural Information Processing

Systems, pages 894–900, 2001.

[Plänkers 03] R. Plänkers & P. Fua. *Articulated Soft Objects for Multi-View Shape and Motion Capture*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 10, 2003.

[Powell 64] M. J. D. Powell. *An efficient method for finding the minimum of a function of several variables without calculating derivations*. In Computer Journal, volume 7, pages 155 – 162, 1964.

[Ramanan 06] D. Ramanan, A. Forsyth & A. Zisserman. *Tracking People by Learning their Appearance*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006. In press.

[Rasmussen 06] C. E. Rasmussen & C. K. Williams. Gaussian process for machine learning. MIT Press, 2006.

[Rosales 01] R. Rosales, M. Siddiqui, J. Alon & S. Sclaroff. *Estimating 3D Body pose using uncalibrated cameras*. In Conference on Computer Vision and Pattern Recognition, December 2001.

[Rosenhahn 07] B. Rosenhahn, T. Brox & H.P. Seidel. *Scaled Motion Dynamics for Markerless Motion Capture.* In Conference on Computer Vision and Pattern Recognition, pages 1–8, 2007.

[Rosenhahn 09] A. Baak B. Rosenhahn, M. Mueller & H.-P. Seidel. *Stabilizing Motion Tracking Using Retrieved Motion Priors.* In International Conference on Computer Vision, Kyoto, Japan, 2009.

[Shakhnarovich 03] G. Shakhnarovich, P. Viola & T. Darrell. *Fast pose estimation with parameter-sensitive hashing*. In International Conference on Computer Vision, Nice, France, 2003.

[Shon 06] Aaron P. Shon, Keith Grochow, Aaron Hertzmann & Rajesh P. N. Rao. *Learning shared latent structure for image synthesis and robotic imitation*. In Neural Information Processing Systems, pages 1233–1240, 2006.

[Sidenbladh 00] H. Sidenbladh, M. J. Black & D. J. Fleet. *Stochastic Tracking of 3D human Figures using 2D Image Motion*. In European Conference on Computer Vision, June 2000.

[Sidenbladh 02] H. Sidenbladh, M. J. Black & L. Sigal. *Implicit Probabilistic Models of Human Motion for Synthesis and Tracking*. In European Conference on Computer Vision, Copenhagen, Denmark, May 2002.

[Sidenbladh 03] H. Sidenbladh & M. J. Black. *Learning the statistics of people in images and video*. IJCV, vol. 54, pages 54–1, 2003.

[Sigal 03] L. Sigal, M. Isard, B. H. Sigelman & M. J. Black. *Attractive people: Assembling loose-limbed models using non-parametric belief propagation*. In Neural Information Processing Systems, Vancouver, BC, Canada, 2003.

[Sigal 04] L. Sigal, S. Bhatia, S. Roth, M. J. Black & M. Isard. *Tracking loose-limbed people*. In Conference on Computer Vision and Pattern Recognition, Washington DC, June 2004.

[Sigal 06] L. Sigal & M. J. Black. *HumanEva: Synchronized Video and Motion Capture Dataset for Evaluation of Articulated Human Motion*. Rapport technique, Department of Computer Science, Brown University, 2006.

[Simon 00] G. Simon, A. Fitzgibbon & A. Zisserman. *Markerless Tracking using Planar Structures in the Scene*. In International Symposium on Mixed and Augmented Reality, pages 120–128, October 2000.

[Sminchisescu 03a] C. Sminchisescu & B. Triggs. *Estimating Articulated Human Motion With Covariance Scaled Sampling*. International Journal of Robotics Research, vol. 22, no. 6, pages 371–391, June 2003.

[Sminchisescu 03b] C. Sminchisescu & B. Triggs. *Kinematic Jump Processes for Monocular 3D Human Tracking*. In Conference on Computer Vision and Pattern Recognition, volume I, page 69, Madison, WI, June 2003.

[Sminchisescu 04] C. Sminchisescu & A. Jepson. *Generative Modeling for Continuous Non-Linearly Embedded Visual Inference*. In International Conference in Machine Learning, Banff, Alberta, Canada, July 2004.

[Sminchisescu 05] C. Sminchisescu, A. Kanaujia, Z. Li & D. Metaxas. *Discriminative Density Propagation for 3–D Human Motion Estimation*. In Conference on Computer Vision and Pattern Recognition, San Diego, CA, June 2005.

[Starck 03] J. Starck & A. Hilton. *Model-Based Multiple View Reconstruction of People*. In International Conference on Computer Vision, 2003.

[Starck 05] J. Starck & A. Hilton. *Spherical Matching for Temporal Correspondence of Non-Rigid Surfaces*. In International Conference on Computer Vision, pages 1387–1394, 2005.

[Sundaresan 08] A. Sundaresan & R. Chellappa. *Model driven segmentation and registration of articulating humans in Laplacian Eigenspace*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 10, pages 1771–1785, 2008.

[Taycher 06] L. Taycher, G. Shakhnarovich, D. Demirdjian & T. Darrell. *Conditional Random People: Tracking Humans with CRFs and Grid Filters*. In Conference on Computer Vision and Pattern Recognition, 2006.

[Thayananthan 03] A. Thayananthan, B. Stenger, P.H.S. Torr & R. Cipolla. *Tracking Articulated Hand Motion using a Kinematic Prior*. In British Machine Vision Conference, pages 589–598, Norwich, UK, 2003.

[Tomasi 03] C. Tomasi, S. Petrov & A. Sastry. *3D Tracking = classification + interpolation*. In International Conference on Computer Vision, pages 1441–1448, 2003.

[Urtasun 04] R. Urtasun & P. Fua. *3D Human Body Tracking using Deterministic Temporal Motion Models*. In European Conference on Computer Vision, Prague, Czech Republic, May 2004.

[Urtasun 05] R. Urtasun, D. Fleet, A. Hertzman & P. Fua. *Priors for People Tracking from Small Training Sets*. In International Conference on Computer Vision, Beijing, China, October 2005.

[Urtasun 06] R. Urtasun, D. Fleet & P. Fua. *3D People Tracking with Gaussian Process Dynamical Models.* In Conference on Computer Vision and Pattern Recognition, New York, 2006.

[Urtasun 08a] R. Urtasun & T. Darrell. *Sparse Probabilistic Regression for Activity-independent Human Pose Inference*. In Conference on Computer Vision and Pattern Recognition, Anchorage, AK, 2008.

[Urtasun 08b] R. Urtasun, D.J. Fleet, A. Geiger, J. Popović, T. Darrell & N.D. Lawrence. *Topologically-constrained latent variable models*. In ICML ’08: Proceedings of the 25th international conference on Machine learning, pages 1080–1087, 2008.

[Vondrak 08] M. Vondrak, L. Sigal & O.C. Jenkins. *Physical Simulation for Probabilistic Motion Tracking*. In Conference on Computer Vision and Pattern Recognition, Anchorage, USA, 2008.

[Wu 03] Y. Wu, G. Hua & T. Yu. *Tracking Articulated Body by Dynamic Markov Network.* In International Conference on Computer Vision, 2003.

[Yamamoto 98] M. Yamamoto, A. Sato, S. Kawada, T. Kondo & Y. Osaki. *Incremental tracking of human actions from multiple views*. In Conference on Computer Vision and Pattern Recognition, Santa Barbara, CA, 1998.

Europass Curriculum Vitae

Personal information

Surname(s) / First name(s)	**Fossati, Andrea**
Email(s)	andrea.fossati@epfl.ch
Nationality(-ies)	Italian
Date of birth	May, 12th 1981
Gender	Male
Webpage	http://cvlab.epfl.ch/~fossati/
Desired employment/ Occupational field	Research and Development

Education and training

Date	2005 - 2010
Title of qualification awarded	PhD Student in Computer, Communication and Information Sciences
Principal subjects/Occupational skills covered	Computer Vision, 3D Human Body Tracking from Monocular Sequences
Name and type of organization providing education and training	CVLab, Ecole Polytechnique Fédérale de Lausanne
Supervisor	Prof. Pascal Fua
Teaching Assistant Activity	Introduction to Computer Science (Master Course, Spring 2007, Spring 2008, Spring 2009), Computer Science 1 (Bachelor Course, Fall 2007)

Date	July 2008
Title of qualification awarded	Visiting PhD Student
Principal subjects/Occupational skills covered	Computer Vision, 3D Human Body Tracking from Monocular Sequences
Name and type of organization providing education and training	Visual Geometry Group, University of Oxford
Supervisor	Prof. Andrew Zisserman

Date	July 2007
Title of qualification awarded	Visiting PhD Student
Principal subjects/Occupational skills covered	Computer Vision, 3D Human Body Tracking from Monocular Sequences
Name and type of organization providing education and training	PERCEPTION Group, INRIA Grenoble - Rhone Alpes
Supervisor	Prof. Radu Horaud

Date	2004 - 2006

Title of qualification awarded	Master of Science
Principal subjects/Occupational skills covered	Computer Science
Name and type of organization providing education and training	University of Illinois at Chicago
Thesis Supervisor	Prof. Milos Zefran
Thesis Title	Cooperative and distributed localization in multi-robot systems
GPA	4.0 / 4.0
Courses	Advanced Computer Vision, Computer Graphics I, High Performance Processors and Systems, Distributed Computing Systems, Virtual Reality, Numerical Analysis, Formal Methods in Concurrent and Distributed Systems

Date	2003 - 2005
Title of qualification awarded	Laurea Specialistica (Master of Science Equivalent)
Principal subjects/Occupational skills covered	Computer Engineering
Name and type of organization providing education and training	Politecnico di Milano
Thesis Supervisor	Prof. Vincenzo Caglioti
Thesis Title	Cooperative Localization in Multi-Robot Systems
Final Grade	110 summa cum laude / 110
GPA	29.4 / 30
Courses	DataBases II, Software Engineering II, Operating Systems Design, Formal Languages and Compilers, Artificial Intelligence, Operation Research, Robotics, High Performance Processors and Systems, Formal Methods in Concurrent and Distributed Systems, Distributed Computing Systems, Statistics, Mechanics, Computer Vision, Computer Graphics, Numerical Calculus, Advanced Topics in Image Analysis, Advanced Topics in Image Synthesis, Internet Infrastructures and Protocols, Computer Graphics, Artificial Intelligence and Robotics Lab, Soft Computing, Data Mining

Date	2000 - 2003
Title of qualification awarded	Laurea (Bachelor Equivalent)
Principal subjects/Occupational skills covered	Computer Engineering
Name and type of organization providing education and training	Politecnico di Milano
Thesis Supervisor	Prof. Luca Ferrarini
Thesis Title	Analysis of Profibus fieldbus and evaluation of its wireless applications
Final Grade	110 summa cum laude / 110
GPA	29.5 / 30

Courses	Mathematical Analysis A and Geometry, Mathematical Analysis B, Experimental Physics I, Experimental Physics II, Computer Science I, Computer Science II, Circuit Theory, Fundamentals of Automatic Controls, Fundamentals of Telecommunication Systems, Fundamentals of Electronics, Software Engineering I, Computer Science III, Economics and Business Administration, Ordinary Differential Equations, Chemistry A, Software Engineering Project, Probability Theory, Digital Logic Design A, Business Management, Industrial Automation, Thermodynamics and Heat Transfer, Databases I, Complementary Project Activity in CS Engineering, Model Identification and Data Analysis, Algebra and Mathematical Logic, Industrial Application of Computer Systems, Theoretical Computer Science

Date	1995 - 2000
Title of qualification awarded	High School Diploma
Name and type of organization providing education and training	Liceo Scientifico Statale Galileo Galilei, Legnano (MI), Italy
Final Grade	100 / 100

Personal skills and competences

Mother tongue(s)	**Italian**
Other language(s)	English, French

Self-assessment *European level*$^{(*)}$	**Understanding**		**Speaking**		**Writing**
	Listening	Reading	Spoken interaction	Spoken production	
English	C1	C2	C2	C1	C2
French	B1	B2	B1	B1	B1

$^{(*)}$*Common European Framework of Reference (CEF) level*

Computer skills and competences

Operating systems: Windows, GNU/Linux
Programming Languages: C, C++, Java, LISP (basic)
Graphic Libraries: OpenGL
Graphic Applications: Blitz3D, 3D Studio Max (basic)
Development Software: Matlab, Maple, Mosel, JBuilder, MS Visual Studio, Vicon IQ

Publications

Real-Time Vehicle Tracking for Driving Assistance - A. Fossati, P. Schönmann and P. Fua - Accepted to Machine Vision and Applications

From Canonical Poses to 3-D Motion Capture using a Single Camera - A. Fossati, M. Dimitrijevic, V. Lepetit and P. Fua - Accepted to IEEE Transactions on Pattern Analysis and Machine Intelligence

Observable Subspaces for 3D Human Motion Recovery - A. Fossati, M. Salzmann and P. Fua - IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, June 2009

Linking Pose and Motion - A. Fossati and P. Fua - European Conference on Computer Vision, Marseille, France, October 2008

Tracking Articulated Bodies using Generalized Expectation Maximization - A. Fossati, E. Arnaud, R. Horaud and P. Fua - CVPR Workshop on Non-Rigid Shape Analysis and Deformable Image Alignment, Anchorage, AK, USA, June 2008

Bridging the Gap between Detection and Tracking for 3D Monocular Video-Based Motion Capture - A. Fossati, M. Dimitrijevic, V. Lepetit and P. Fua - IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MI, June 2007

Cooperative, distributed localization in multi-robot systems: a minimum-entropy approach - V. Caglioti, A. Citterio and A. Fossati - IEEE Workshop on Distributed and Intelligent Systems, Prague, Czech Republic, June 2006

Reviewing Activity

Reviewer for IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) in 2006 and 2007.

Additional information

Interesting Facts

Took part to the Italian national final of the Math Olympics ($\sim$ 300 students out of all the Italian High Schools) both in 1999 and 2000.
Took part to the Italian national final of the Physics Olympics ($\sim$ 80 students out of all the Italian High Schools) in 2000.
Admitted to the Excellence School 'Collegio di Milano' in the semester of its foundation (Spring 2004).

Hobbies and Interests

Sports: Basketball (currently playing in Renens Basket, Swiss 1ère Ligue Nationale), Soccer, Volleyball.
Travels, Music, Photography.