MANCHESTER METROPOLITAN UNIVERSITY

3D Human Motion Tracking and Pose Estimation using Probabilistic Activity Models

John Darby

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

Faculty of Science and Engineering
The Department of Computing and Mathematics

October 2010

Abstract

This thesis presents work on *generative* approaches to human motion tracking and pose estimation where a geometric model of the human body is used for comparison with observations. The existing generative tracking literature can be quite clearly divided between two groups. First, approaches that attempt to solve a difficult high-dimensional inference problem in the body model's full or *ambient* pose space, recovering freeform or *unknown activity*. Second, approaches that restrict inference to a low-dimensional *latent* embedding of the full pose space, recovering activity for which training data is available or *known activity*.

Significant advances have been made in each of these subgroups. Given sufficiently rich multiocular observations and plentiful computational resources, high-dimensional approaches have been proven to track fast and complex unknown activities robustly. Conversely, low-dimensional approaches have been able to support monocular tracking and to significantly reduce computational costs for the recovery of known activity. However, their competing advantages have – although complementary – remained disjoint. The central aim of this thesis is to combine low- and high-dimensional generative tracking techniques to benefit from the best of both approaches.

First, a simple generative tracking approach is proposed for tracking known activities in a latent pose space using only monocular or binocular observations. A hidden Markov model (HMM) is used to provide dynamics and constrain a particle-based search for poses. The ability of the HMM to *classify* as well as synthesise poses means that the approach naturally extends to the modelling of a number of different known activities in a single joint-activity latent space.

Second, an additional low-dimensional approach is introduced to permit transitions between *segmented* known activity training data by allowing particles to move between activity manifolds. Both low-dimensional approaches are then fairly and efficiently combined with a simultaneous high-dimensional generative tracking task in the ambient pose space. This combination allows for the recovery of sequences containing multiple known and unknown human activities at an appropriate (dynamic) computational cost.

Finally, a rich hierarchical embedding of the ambient pose space is investigated. This representation allows inference to progress from a single full-body or *global* non-linear latent pose space, through a number of gradually smaller *part-based* latent models, to the full ambient pose space. By preserving long-range correlations present in training data, the positions of occluded limbs can be inferred during tracking. Alternatively, by breaking the implied coordination between part-based models novel activity combinations, or *composite activity*, may be recovered.

Acknowledgements

I would like to express my gratitude to my supervisors, Baihua Li and Nick Costen, for their patient guidance and support during this work. I am thankful to them for giving me this opportunity to study Computer Vision, and to the Dalton Research Institute for funding my studentship.

A number of other members of the Image and Sensory Computation Group have been generous in their help. Particularly, Hui Fang has explained many new concepts and tirelessly assisted me with laborious data collection tasks, and Edmond Prakash has offered advice and encouragement throughout.

A number of researchers at other universities have also been kind enough to help me with my work at various stages. Shaobo Hou has patiently and succinctly answered many questions over the past three years. I am particularly indebted to Neil Lawrence for helping me to use the H-GPLVM, engaging in really reproducible research, and reading draft papers and thesis chapters; and David Fleet for very kindly processing data and helping me to improve my writing style.

Thanks are also owed to the team of researchers at Brown University for creating and maintaining the *HumanEva* datasets, baseline, and evaluation system. Without these I would have understood and accomplished far less in these years. In particular, Alexandru Bălan and Leonid Sigal have gone well beyond the call of duty in answering my numerous emails.

I am grateful to a number of other researchers and groups who have chosen to share their data and code. Foremost amongst these, Kevin Murphy for creating the HMM Toolbox and Ian Nabney for providing the Netlab machine learning software. The motion capture data in Chapter 7 was obtained from the Carnegie Mellon University database at http://mocap.cs.cmu.edu/. The database was created with funding from NSF EIA-0196217.

I have received much help and advice from a succession of office mates in the Department of Computing and Mathematics. In the early days, Tim Jackson, Naresh Subramaniam and Darren Dancey all provided me with useful advice on getting started in research, and on local public houses. More recently, Adeel Hashmi and Peter Harding have offered valuable advice in matters of caffeine-based beverages and Greggs pasties.

Finally, I thank my parents for their constant encouragement and Becca for her companionship and unwavering support.

Contents

List of Figures

Publications

This thesis is based on material from the following publications:

1. J. Darby, B. Li and N. P. Costen. Tracking human pose with multiple activity models. Pattern Recognition 43(2010), pp. 3042–3058.

2. J. Darby, B. Li and N. P. Costen. Tracking a walking person using activity-guided annealed particle filtering. In IEEE International Conference on Face and Gesture Recognition, pp. 1–6, 2008.

3. J. Darby, B. Li and N. P. Costen. Behaviour based particle filtering for human articulated motion tracking. In IAPR International Conference on Pattern Recognition, pp. 1–4, 2008.

4. J. Darby, B. Li, N. P. Costen, D. J. Fleet and N. D. Lawrence. Backing off: hierarchical decomposition of activity for 3D novel pose recovery. In British Machine Vision Conference, pp. 1-11, 2009.

5. J. Darby, B. Li and N. P. Costen. Human activity tracking from moving camera stereo data. In British Machine Vision Conference, pp. 865–874, 2008.

6. J. Darby, B. Li and N. P. Costen. Activity classification for interactive game interfaces. In International Journal of Computer Games Technology, pp. 1–7, 2007.

Abbreviations

ADABOOST	**ADA**ptive **BOOST**ing
ASOM	**A**rticulated **S**oft **O**bject **M**odel
ARP	**A**uto **R**egressive **P**rocess
APF	**A**nnealed **P**article **F**ilter
BME	**B**ayesian **M**ixture of **E**xperts
BC-GPLVM	**B**ack **C**onstrained **GP-LVM**
B-GPDM	**B**alanced **GPDM**
CONDENSATION	**CON**ditional **DENS**ity Propag**ATION**
CSS	**C**ovariance **S**caled **S**ampling
DP	**D**ynamic **P**rogramming
EKF	**E**xtended **K**alman **F**ilter
GP	**G**aussian **P**rocess
GP-LVM	**G**aussian **P**rocess **L**atent **V**ariable **M**odel
GPDM	**G**aussian **P**rocess **D**ynamical **M**odel
HMM	**H**idden **M**arkov **M**odel
H-GPLVM	**H**ierarchical **GP-LVM**
HS	**H**yperdynamic **S**ampling
HOG	**H**istogram of **O**riented **G**radients
ICP	**I**terative **C**losest **P**oint
KJS	**K**inematic **J**ump **S**ampling
LELVM	**L**aplacian **E**igenmaps **L**atent **V**ariable **M**odel
LLE	**L**ocally **L**inear **C**oordination
MoCap	**Mo**tion **Cap**ture
MAM	**M**ultiple **A**ctivity **M**odels
MHT	**M**ultiple **H**ypothesis **T**racker
NPBP	**N**on-**P**arametric **B**elief **P**ropagation
NN	**N**eural **N**etwork
PaMPas	**Pa**rticle **M**essage **Pas**sing
PSM	**P**ictorial **S**tructure **M**odels
PS	**P**artitioned **S**ampling
PSH	**P**arameter **S**ensitive **H**ashing
PCA	**P**rincipal **C**components **A**nalysis
PPCA	**P**robabilistic **PCA**
P-R	**P**rojection-**R**econstruction (Error)
RANSAC	**RAN**dom **SA**mple **C**onsensus
RVM	**R**elevance **V**ector **M**achine
RBF	**R**adial **B**asis **F**unction

S-GPLVM	**S**caled **GP-LVM**
SCG	**S**caled **C**onjugate **G**radient
SSD	**S**um of **S**quared **D**ifferences
SLDS	**S**witching **L**inear **D**ynamical **S**ystems
SIR	**S**equential **I**mportance **S**ampling
VLMM	**V**ariable **L**ength **M**arkov **M**odel
WSL	**W**andering **S**table **L**ost (Tracker)

Chapter 1

Introduction

In this chapter a brief introduction to the field of human motion tracking is given (expanded further in Chapter 2). A number of important terms are defined and the thesis statement and outline are given.

1.1 Background

There is a rich body of literature on the analysis of human motion from non-invasive visual cues, driven by applications in diverse areas such as human-computer interaction, visual surveillance and medicine [RKM08]. The topic is a broad one that has grown considerably in recent years [MHK06] and features many quite distinct sub-branches. The taxonomy of Poppe [Pop07b] is adopted to define the area in which this thesis attempts to contribute. The literature may be broadly divided into two groups: *generative* approaches that optimise the configuration of a volumetric *body model* to coincide with observations, and *discriminative* approaches that predict pose configurations directly from observations. In this work novel generative tracking approaches are developed, using a fixed geometric body model based on the dimensions of the tracking subject to synthesise poses for comparison with image observations. The generative approach entails both a *modelling* and *estimation* stage. Modelling requires the

specification of an *objective function* for comparison of the body model with observations, and estimation requires the recovery of the optimal pose given the objective function.

Defining a single pose given a simple body model consisting of a kinematic tree requires around 30 parameters. Even with carefully formulated objective functions e.g. [ST03a, SB01], the estimation problem over a 30D space given a single observation contains a large number of local optima [ST02a]. Simple approaches like gradient descent are therefore unlikely to find or maintain globally optimal solutions. For this reason probabilistic inference has been favoured and particle filtering methods [AMGC02] have become perhaps the most widely adopted approach to estimation in generative tracking. By maintaining multiple hypotheses about the true pose configuration particle filters are, in theory at least, capable of supporting a multimodal objective surface during estimation.

By deploying a particle filter in a body model's full or *ambient pose space* and permitting each configuration parameter to vary independently, no restrictions are placed on pose and freeform or *unknown activity* can be tracked. However, in practice such approaches have relied upon: large particle numbers to sample the pose space with sufficient density [BEB08]; carefully constrained dynamical models [CGH05]; and observations from a minimum of four synchronised cameras to minimise ambiguity in the modelling step [SBB10]. Reducing reliance on any one of these factors is desirable, but tends to come at the expense of increased dependence on another.

Learning a low-dimensional *latent pose space* from training data is an effective method for constraining the estimation task. Projection of training poses onto a low-dimensional manifold encodes correlations between body model parameters and particle filtering in the resulting subspace has permitted reductions of both particle numbers and camera numbers, e.g. [TLS05, LPS07, RRR08a]. The central limitation of such approaches is that they constrain the classes of activity that can be tracked to *known activities* – that is, those present in the training set.

As an example, take a latent pose space learned from *walk* activity data. This space contains only walking poses and has no capacity to generalise to a new activity regardless of its simplicity. A novel activity such as *wave* will inevitably cause tracking to fail, see also Section 3.3.6 for an experimental proof. To improve robustness, one might therefore construct a latent pose space from both *walk* data and *wave* data. However, even a subtly different combined or *composite activity* such as *walk whilst waving* remains beyond the scope of the new model: extra training data is necessary for *every* activity of interest.

This thesis looks at a number of ways in which the constraints of a latent pose space can be relaxed to permit the recovery of hitherto unseen poses. The aim is to retain the efficiency and robustness of latent pose space estimation while introducing the potential for generalisation to events such as known activity transitions, composite activity and unknown activity. A number of novel solutions are put forward (these are listed in Section 1.3), but each involves permitting particles to move away from latent variables in a controlled manner. This may be by moving between different manifolds in a joint-activity pose space, by breaking the temporal correlations between individual body parts, or by flowing out of the latent pose space and into the unconstrained ambient pose space. By carefully integrating each of these possibilities into a particle-based approach to estimation, the robust tracking of multiple classes of activity at an appropriate (dynamic) computational cost is demonstrated.

The field of human motion tracking and pose estimation has recently benefitted from the introduction of freely available datasets that include ground truth and allow for quantitative evaluation and comparison of techniques. In particular, the *HumanEva-I* and *HumanEva-II* datasets provide observations of human motions from multiple cameras and a synchronised motion capture (MoCap) record of ground truth [SBB10]. These tools have transformed the presentation of results within the field, making possible the quantitative cross-comparison of a range of

existing techniques[1]. In the remainder of this thesis each contribution is thoroughly tested on either the *HumanEva* datasets or other freely available human motion datasets, quantitative results are presented and comparisons drawn with existing state of the art approaches. In addition, software used to create many of the main results is made available to other researchers via the author's website.

1.2 Problem Statement

In Chapter 2 a number of high-dimensional and low-dimensional tracking approaches from the literature are highlighted. High-dimensional approaches are able to track freeform motions where rich observation data and sufficient computational resources are available. Conversely, low-dimensional approaches can recover known activities from limited observation data and at reduced computational cost. However, no attempts to marry the two within a single framework exist. The potential benefit is a generative tracking system that can recover known activities efficiently and robustly (e.g. through occlusions) from a low-dimensional pose space, but upon encountering unknown activity is able to adjust the scope of its inference task to recover unknown poses from a high-dimensional pose space.

The problem statement addressed by this thesis is concisely stated as follows,

> *Low-dimensional generative tracking techniques have brought a number of advantages over their high-dimensional counterparts including reduced computational cost and accurate tracking from limited observation data. However, the requirement that training data be available for an offline learning stage means these advantages come at the expense of flexibility: the ability to track hitherto unseen activities. The strengths of low-dimensional and high-dimensional generative techniques are potentially complementary; this thesis investigates ways in which they might be combined.*

[1] See for example work in the the International Journal of Computer Vision's recent special issue on "Evaluation of Articulated Human Motion and Pose Estimation" [EVA10].

1.3 Contributions of this Thesis

The three main contributions of this thesis are as follows: (i) the specification of a novel low-dimensional generative tracking technique for known activity tracking; (ii) its efficient combination with a high-dimensional generative tracking technique for known and unknown activity tracking; (iii) the use of a hierarchy of part-based latent pose spaces for composite activity tracking. The motivation and context for this work is carefully introduced over the course of the following two chapters, but a concise list of the resulting contributions with relevant sections forward-referenced is given below.

1. Construction of activity models for known activities. PCA is used to recover a latent pose space from MoCap training data (Section 3.3.2), and a *dynamical model* learned by training a hidden Markov model (HMM) from the resulting distribution of latent variables (Section 3.4.2). This combination of pose space and dynamical model is referred to as an *activity model*.

2. Integration of the activity model into an annealed particle filtering (APF) [DBR00] framework for particle dispersion during estimation (Section 4.2). The definition of a number of novel objective functions that permit the resulting tracker – termed HMM-APF – to recover known activity from narrow-baseline stereo (Section 4.3.1), monocular (Section 4.3.2) and wide-baseline stereo observations (Section 4.3.3).

3. Definition of two further complementary activity models. Known *activity transitions* are modelled by permitting particles to flow between activity manifolds in a joint-activity latent pose space (Section 5.3.3). Unknown activities are modelled using Gaussian noise to propagate particles in the high-dimensional ambient pose space (Section 5.3.1).

4. Proposal of a multiple activity model APF (MAM-APF) scheme to unify separate search strategies under the APF framework (Section 5.4). A *particle stacking* approach is described, allowing for the simultaneous consideration of multiple activity models described by different dynamical models

spanning pose spaces of different dimensionality. A variable number of particles are resampled at each annealing layer, allowing for the recovery of known activities using only a small number of particles in latent pose space, and unknown activities using a large number of particles in the ambient pose space (Section 5.4.2).

5. Proposal of an activity model for composite activity tracking based on a hierarchy of latent variables adopted from the machine learning literature on non-linear dimensionality reduction [LM07]. Inference moves gradually between a single low-dimensional known activity latent pose space, through a number of gradually smaller part-based models, to the body model's unconstrained high-dimensional ambient pose space (Section 6.3). This approach permits the recombination of activity to create novel poses (Section 6.5.3) while retaining the ability to solve traditional "global" latent space problems such as tracking through occlusion (Section 6.5.4).

Chapter 2

Literature Review

2.1 Introduction

In this chapter an overview of the field of articulated human motion tracking is presented. Due to the large volume of work in this area, the overview is not intended to be exhaustive but rather to define the areas in the literature where this thesis attempts to contribute, and their wider context. Comprehensive reviews of the literature can be found in a number of review papers e.g. [MG01, MHK06, Pop07b, Smi08].

In the following sections, work on both *pose estimation*, and *tracking* is reviewed. Following the definitions given by Sigal [Sig08], these approaches are defined as follows. Pose estimation problems are concerned with the estimation of a single static human pose at a single instant, given a single sensor observation. Tracking problems are concerned with the estimation of a sequence of static human poses given a sequence of sensor observations and the initial pose estimate corresponding to the first observation. The "sensor" may be multiocular, in which case there are several synchronised images at any instant. Although the focus of this thesis is a solution to the tracking problem, pose estimation is a complimentary (but inherently more challenging) problem that may be used to

initialise tracking and can, in theory, be used for tracking by solving a pose estimation problem at each frame.

Following the taxonomy of Poppe [Pop07b], the discussion of the literature is divided between two central approaches: *discriminative* (or model-free) and *generative* (or model-based). Discriminative approaches attempt to model directly a mapping from sensor observation to pose. Generative approaches use a model of the human body to synthesise pose hypotheses. They then attempt to model the likelihood of resulting pose hypotheses given an observation by constructing a model of the observation likelihood, or *objective function*. Discriminative approaches usually focus on pose estimation and generative approaches on tracking but this is not always the case.

2.2 Discriminative (Model-Free) Approaches

Discriminative approaches to pose estimation and tracking attempt to infer human pose directly from an observation. Although not directly relevant to the work presented in this thesis, discriminative approaches are important for the way they compliment generative approaches (e.g. for initialisation) and for the component techniques they have in common (e.g. dimensionality reduction). The area can be broadly divided between the following two methodologies: *example-based* approaches – that retain a large database of image-pose pairs and given an input observation, search for the most closely matching image to return the associated pose, e.g. [SVD03, PP06, OMBH06, MM06, How07, RRR+08b]; *learning-based* approaches – that learn a continuous mapping between observations and pose allowing the training set to be discarded, e.g. [AT04a, AT06, EL04, RS01, Bra99, GSD03, SKLM05, SKM06b].

2.2.1 Example-Based

For example-based approaches, the number of full-body training examples required to recover general motions is large [MM06]. Even if a database is thought to be "complete", matching success will also depend on the choice of descriptors and the particular search strategy. Nearest neighbour searches e.g. using silhouette [How07] or histogram of oriented gradients (HOG) representations [Pop07a], have been successful but are impractical with large datasets. One solution is to use fast approximations, for example Shakhnarovich *et al.* [SVD03] use parameter sensitive hashing (PSH) to retrieve matching exemplars in a fast nearest neighbour approximation. Learning-based discriminative approaches offer an alternative mechanism for benefitting from the information within large datasets of image-pose pairs.

2.2.2 Learning-Based

Learning-based approaches offer the potential to remove the requirement to store and search large amounts of training data. The alternative problem they address is how best to specify a general *mapping* between image and pose. For example, Agarwal and Triggs [AT04a] use a relevance vector machine (RVM) to characterise a mapping between histograms of shape contexts and pose. This approach is challenging because it is possible for different poses to give rise to the same image features given different camera views and subject orientations; mappings are therefore multi-valued. To overcome this problem Rosales and Sclaroff [RS01] cluster training data in the pose space before using neural networks (NNs) to learn different mapping functions for each cluster. More generally, mixtures of regressors [AT06] have been introduced to cope with the multivalued nature of the problem, e.g. the use of a Bayesian mixture of experts (BME) by Sminchisescu *et al.* [SKLM05]. This multivalued problem is similar to that which arises in generative tracking, where a number of different pose hypotheses may agree well

with an observation. A number of important techniques are therefore common to both approaches.

Perhaps the most important aspect of learning-based approaches in the context of the work presented in this thesis, is the use of dimensionality reduction. In generative work the observation that typical human motions occupy only a small subspace within the full space of kinematically feasible poses has led to the use of dimensionality reduction techniques to learn latent embeddings of the pose space (see also Section 2.3). Similarly, in learning-based discriminative approaches such techniques may be used to learn an embedding in the image space and constrain the recovery of a mapping to pose space. The central issue for dimensionality reduction to overcome is that the data is highly non-linear and therefore a non-linear, twisted data manifold must be recovered. This has led to the adoption of state of the art dimensionality reduction techniques e.g. locally linear embedding [RS00] and the Gaussian process latent variable model [Law05].

Elgammal and Lee [EL04] use local linear embedding (LLE) [RS00] to learn a non-linear manifold embedding from visual input. They then learn a mapping from the embedding space to the pose space with a generalised radial basis function (GRBF), allowing the reconstruction of poses from monocular silhouettes. Learning this mapping is simplified by the recovery of an intermediate low-dimensional visual manifold. Bowden *et al.* [BMS98, BMS00] combine both shape (2D contour, 2D head and hand locations) and structure (3D salient point locations) parameters into a single feature vector and learn a "piecewise linear" model. This is done by performing an initial principal components analysis (PCA) on data and then clustering within the global eigenspace before learning a further set of local linear models by PCA – the approach is sometimes referred to as hierarchical PCA (HPCA) [BMS97, HH97]. By finding 2D estimates for head and hand locations they are able to constrain a search for the optimal contour solution. Likewise by finding the linear model closest to this initial estimate and then the closest allowable training datum within that cluster, they are able to perform 3D pose estimation. With a presumption of small inter-frame changes in

silhouette and head/hand positions, a more efficient constrained tracking mode search is possible.

Ong and Gong [OG99] take a similar approach but using multiocular observations. They note the potential for discontinuous changes in the 2D contour of a 3D shape during tracking and, following [HH98][1], learn a Markov transition matrix to account for large "jumps" between locally linear clusters within the global eigenspace. Furthermore, they account for uncertainty in the current estimate by maintaining multiple pose hypotheses using a particle filter [IB98a, AMGC02]. The work is interesting for its use of artificially inflated dynamics in particle dispersion for robust tracking. This is in common with a number of generative tracking approaches, and also with the work presented in this thesis. Grauman *et al.* [GSD03] demonstrate quantitatively superior performance when using multiocular observations. They combine silhouette contour information and 3D pose coordinates into a single feature vector and use a mixture of probabilistic principal components analysis (PPCA) to describe a prior density over training data. They are able to treat 3D pose reconstruction as a missing data problem (see also Appendix B for more detail on PPCA).

Recent work by Lawrence [Law05] has proposed a dual probabilistic interpretation of PCA (see also Appendix B) that may be non-linearised using Gaussian processes (GPs) [RW06] to give a form of probabilistic non-linear dimensionality reduction. This technique, termed the Gaussian process latent variable model (GP-LVM), has been shown to give good reconstruction results on human motion data [QDLM08] and has proven popular and effective in generative work (see also Section 2.3). More recently it has also been applied to the discriminative task [ETL07, MP06] using techniques similar to those described above. For example, Ek *et al.* [ETL07] adopt a shape plus structure approach and learn a joint latent space using training data from both the pose space and the image space (similar with [BMS98, BMS00]), while Moon and Pavlović use a particle filter to support multiple hypotheses in a non-linear latent space (similar with [OG99]). Ek *et al.* use dynamical models to disambiguate sequences of (one-to-many) pose

[1] Here the problem is the 2D projection of a 3D hand, rather than human body.

to latent space mappings. The approach is impressive but reconstruction errors are found to occur due to a lack of training data, or more specifically a lack of viewpoint data given the training activity (see Fig. 2 in [ETL07]). This work emphasises the fundamental difficulty of the discriminative task: large amounts of training data must be available, regardless of whether the approach is example- or learning-based.

Although learning-based approaches can remove the need to *retain* a large database for "online" searching, the data acquisition task required for "offline" learning remains formidable. In more recent work Ong *et al.* [OMBH06] attempt to achieve viewpoint invariant monocular tracking by moving to richer feature vectors, or "exemplars", that incorporate image information from twelve different viewpoints in addition to structural pose information. Clustering is performed in the exemplar space (without the use of dimensionality reduction) and a particle filter is again used to maintain multiple pose hypotheses. The approach is an attempt to negate the need to learn multiple models for multiple viewpoints e.g. [EL04].

In order to take a discriminative approach to pose estimation and/or tracking, training data must contain the necessary set of test poses and viewpoints, and it must be possible to extract the relevant image features from both training and test sequences. Thus, example-based approaches require huge databases to generalise to freeform motions. Those that attempt to generalise to new poses by interpolating a number of close matches, e.g. [Pop07a], may be more expressive but cannot guarantee the resulting pose solution is viable. For learning-based discriminative approaches, where an intermediate manifold representation is used, a viable pose is guaranteed by "clamping" image feature projections to the manifold before mapping to the pose space, e.g. [EL04], but precludes the recovery of a novel pose. A more expressive system able to correctly estimate some novel inputs is described by Agarwal and Triggs [AT04a] who learn a mapping directly from image space to pose space. However, where the input is ambiguous there is still no guarantee that the resulting pose estimate is kinematically viable, see

for example the "compromise solutions" in Fig. 7 of [AT04a]. Prior knowledge, including kinematic constraints, is difficult to incorporate into such approaches.

These difficulties have played a part in the wide adoption of generative approaches (see also Section 2.3), where a body model is moved to coincide with image features, rather than image features having to be "recognisable". This is the route taken in this thesis. However, many of the same difficulties apply to generative approaches also, e.g. reliable feature extraction, generalisation to novel poses, supporting multiple hypotheses. Learning-based discriminative approaches and generative approaches have therefore seen application of many of the same techniques. Their combination as two separate but complementary tracking processes has proven effective and is currently the focus of much attention, further discussion is given in Section 2.4.1.

2.3 Generative (Model-Based) Approaches

Generative approaches to pose estimation and tracking involve the projection of a geometric body model into the image observation for the maximisation of an observation likelihood or *objective* function. Such approaches are, overwhelmingly, based upon the 3D kinematic tree of Marr and Nishihara [MN78] with some choice of volumetric primitive to model individual limbs e.g. cylinders [DR05] or superquadrics [ST03a]. Although there are a range of choices available for the parameterisation of such a model a high number of degrees of freedom is inescapable, leading to a high-dimensional ambient state space (> 30D). A brute force search of such a space for the optimal pose given the objective function is not feasible, all but precluding the use of generative models for pose estimation. The problem is analogous to the task of searching a "complete" pose database in example-based discriminative tracking. Generative approaches are therefore limited to tracking tasks, where a good initialisation is available with the first observation and the problem can be reduced to recovering a series of small inter-frame changes in pose.

Even when limited to tracking applications, the traditional generative approach described above has a number of drawbacks. Without sufficiently rich observations, ambiguity in the 3D to 2D projection, or in the observation model can lead to a persistently multimodal objective surface over the state space [ST02a]. Although probabilistic multiple hypothesis methods exist for supporting (at least temporarily) this ambiguity (e.g. multiple hypothesis tracking [CR99], particle filtering [IB98a, AMGC02]) and for attempting to resolve it (e.g. annealed particle filtering [DR05] covariance scaled sampling [ST03a] and kinematic jump sampling [ST03b]) they are computationally expensive and the true mode does not always win out.

These probabilistic approaches use Bayes' rule to approximate a posterior distribution by combination of the observation model with a predictive prior on pose. If the gap in pose space between the true solution and the set of current hypotheses grows large with respect to the predictive prior, there is little hope of recovery. In practice, this has meant that all but a handful of the most powerful (and complex) generative approaches that employ sophisticated hypothesis propagation techniques, e.g. [ST03a, ST03b], are unable to recover *freeform* motion from *monocular* observations. It should also be noted that even these methods require "good" monocular observations – e.g. high quality silhouettes [ST03b] – that may be difficult to achieve outside the laboratory. In general, generative algorithms require multiocular sensor observations featuring a minimum of four wide-baseline cameras for robust tracking (see quantitative studies in [BSB05, BEB08, SBB10]).

The limitations of generative approaches have motivated a number of developments in the field. First, the combination of discriminative approaches with generative ones, the former providing initialisation at the first frame and reinitialisation from errors (see also Section 2.4.1). Second, the use of activity-specific predictive priors imposed either within the high-dimensional state space, or by the learning of a low-dimensional embedding of the pose space, and used for the propagation of pose estimates (see also Section 2.3.2). Finally, the emergence of a new subset of *bottom-up* generative approaches that are distinct from the class of

"synthesise and test" approaches discussed so far (which are subsequently referred to as *top-down*). Bottom-up approaches model the body as a set of independent limbs in a global coordinate system, with only weak constraints enforced between neighbours. Inference involves detecting and then assembling these limbs into a plausible pose. This class of approaches has been particularly successful in performing efficient *2D* pose estimation in monocular images and this in turn has led to a new strand of approaches that attempt to infer 3D pose from 2D pose.

Because bottom-up approaches employ a less constrained model of the human body, e.g. connections between limbs can be "loose" [SBR+04] as opposed to the exact constraints of top-down models, they are sometimes discussed separately from generative approaches e.g. [MHK06], or sometimes not at all e.g. [Smi08, Urt06]. Here a discussion of bottom-up approaches is included as there appears to be potential for their integration with top-down generative approaches such as the ones presented in this thesis (see also Section 2.3.2 and Chapter 6). Further, bottom-up approaches are classified as generative (as in [Pop07b, Dau09]) because they still involve the projection of an – albeit weakly constrained – body model into the observation for comparison with image features. The body model typically features fixed size cylindrical limbs, and an orthographic camera projection is used for their comparison with image features, e.g. [FH05].

2.3.1 Bottom-Up

Bottom-up generative approaches to pose estimation and tracking have been heavily influenced by the work of Fischler and Elschlager on pictorial structure models (PSMs) [FE73]. PSMs use a collection of parts arranged in some deformable configuration to model the appearance of objects in images. Given an observation, matching is performed by the minimisation of an objective function that incorporates each part's fit with image evidence and its deformation cost given its immediate neighbours. Based on this approach a number of applications to human pose estimation have been described [IF01, FH05, SBR+04, SB06b, LH05, RFZ07]. Ioffe and Forsyth [IF01] present methods for 2D pose

estimation based on finding a large number of possible locations for individual body parts and then "pruning" the results to leave only *groups* of parts that satisfy the kinematic constraints of the full body. Felzenszwalb and Huttenlocher [FH05] show that by discretising limb state spaces and modelling neighbouring limb interactions with a particular form of spring-like connection, a globally optimal 2D pose may be found using dynamic programming (DP). Ramanan *et al.* [RFZ07] present methods for the automatic construction of person-specific appearance models for individual limbs in PSMs, facilitating identification and robust 2D tracking of multiple subjects.

The class of bottom-up generative approaches are capable of 2D monocular pose estimation where top-down generative approaches are not, but they are also restrictive in a number of ways. First, modelling only neighbouring limb constraints precludes the consideration of long range limb interactions e.g. to deal with occlusions (although efforts are made to account for this in [LH05]). Second, the model offers no capacity to include temporal constraints. Finally, efficient inference relies on relatively heavy discretisation of each model part's state space and is not suitable for extension into 3D. Recent work by Sigal *et al.* [SB06b, SB06c, SBR+04] has gone some way towards alleviating these problems.

Sigal *et al.* [SBR+04] define a loose-limbed body model that supports a broader range of interactions between any (not only neighbouring) pair-wise combination of limbs. They show how non-parametric belief propagation (NPBP) may be used to perform 3D pose inference. Sigal and Black [SB06b] apply the approach to 2D pose estimation by giving consideration to self occlusions when evaluating image likelihoods. In the standard bottom-up PSM formulation [FH05] there is nothing to stop multiple body parts occupying the same image feature (to minimise the energy function). This problem is known as "over-counting" and is also addressed by the work of Jiang [Jia09]. The authors later use this approach to provide an intermediate 2D pose estimate from monocular sequences, before "lifting" to recover a 3D pose estimate [SB06c]. They achieve this by generalising the discriminative approach of Agarwal and Triggs [AT04a] to learn a mapping from 2D poses – rather than 2D silhouettes – to 3D poses. Similarly, Micilotta

et al. [MOB06] assemble 2D upper body pose estimates using a combination of AdaBoost and RANSAC [MOB05] before "lifting" to 3D via an example-based discriminative database search.

In summary, bottom-up approaches can be broadly divided between the following alternative methodologies: (i) the use of a simple and coarsely adjustable body model for fast 2D pose inference; (ii) the adoption of a more expressive 3D body model at the cost of more expensive and (necessarily) approximate inference [SBR+04]; (iii) the recovery of 2D pose [SB06b] in an intermediate stage before "lifting" to 3D pose estimates [SB06c, MOB06]. In Section 2.4.2 it is argued that the final methodology motivates a previously unexplored combination of bottom-up and top-down generative approaches within a single tracking framework. This is where the top-down 3D tracking approach takes as its input the 2D pose estimates of a simultaneous bottom-up scheme.

2.3.2 Top-Down

Top-down generative approaches optimise the configuration of a body model (usually 3D) to coincide with features in the image observation. Although a number of options are available for the parameterisation of a 3D kinematic tree such as that of Marr and Nishihara [MN78] e.g. Euler angles [BSB05], quaternions [SBR+04] and exponential maps [BM98], the dimensionality of the complete pose vector is typically upwards of 30D regardless of the choice. Even when using sophisticated objective functions [ST03a, SB01], direct optimisation methods will encounter many local minima, as shown experimentally by Sminchisescu and Triggs [ST02a]. Even given a good initialisation, such schemes are likely to be distracted from the true configuration by local optima, never to recover. For this reason probabilistic inference has been favoured over deterministic optimisiation.

By taking the ingredients of the top-down generative tracking problem (observations of human movement $\underline{z}_0, \underline{z}_1, ..., \underline{z}_t$, an initialising pose $\underline{s}_0$ and a model of the observation likelihood $p(\underline{z}_t|\underline{s}_t)$), making a first order Markov assumption about

the underlying pose evolution $p(\underline{s}_t|\underline{s}_0, \underline{s}_1, ..., \underline{s}_{t-1}) = p(\underline{s}_t|\underline{s}_{t-1})$ and an independent sensor assumption $p(\underline{z}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_{t-1}, \underline{s}_0, \underline{s}_1, ..., \underline{s}_t) = p(\underline{z}_t|\underline{s}_t)$, one can use Bayes' rule to derive the following expression for the posterior state density (see also Appendix A)

$$\underbrace{p(\underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t)}_{\text{Posterior at time } t} = \frac{1}{C} \underbrace{p(\underline{z}_t|\underline{s}_t)}_{\text{Likelihood}} \int_{\underline{s}_{t-1}} \underbrace{p(\underline{s}_t|\underline{s}_{t-1})}_{\text{Dynamical model}} \underbrace{p(\underline{s}_{t-1}|\underline{z}_0, \underline{z}_1, ..., \underline{z}_{t-1})}_{\text{Posterior at time } t-1} d\underline{s}_{t-1}. \tag{2.1}$$

This expression, commonly called the *filtering equation*, may be interpreted as Bayes' rule for inferring a posterior state density from data for the time-varying case [IB98a]. It has formed the basis for inference in a great number of visual tracking applications via both Kalman filtering [GdBUP95, KM96] and particle filtering [DNBB99, BSB05].

Kalman filtering offers a provably optimal solution where the observation likelihood is Gaussian and the dynamics linear with Gaussian noise [May79]. However, where the system state is complex, e.g. an articulated body, and observations ambiguous, e.g. monocular [ST02a] or cluttered [BI98], the observation likelihood is inevitably multimodal, and therefore non-Gaussian. Furthermore, a presumption of linear dynamics is a poor one for human motions where the dynamical model is required to reflect non-linearities such as joint angle accelerations, hard limits or "end stops", and limb collisions [SBB10]. The extended Kalman filter (EKF) takes steps to support non-linear observations likelihoods and dynamics, requiring only that they are differentiable in order that a first order Taylor expansion approximation can be used. However, this locally linear assumption is still restrictive and the EKF cannot incorporate discontinuous dynamics such as joint endstops, or support truly multimodal posterior distributions due, for example, to an absence of visual information, e.g. at the "elbow singularity"[2] [DNBB99].

In their evaluation of the Kalman filter, Deutscher *et al.* [DNBB99] demonstrate multimodality experimentally using Monte-Carlo estimation for sample-based

[2] A straight elbow leads to high uncertainty in shoulder parameters as it is no longer possible to observe rotations of the upper arm.

non-parametric estimates of the true posterior. Rather than simply a simulation tool, efficient time-recursive Monte-Carlo methods such as particle filtering[3] [AMGC02] have become the dominant framework for probabilistic inference in visual tracking. Here an arbitrary posterior distribution is represented via a set of weighted "particles" moving within the state space according to some prior dynamical model. Each particle is weighted based on its associated observation likelihood, and "resampling" of the particle set – selecting particles with probability proportional to their weight – produces distributions that approximate the true posterior at each instant. Particle-based methods form the basis for inference in this thesis and full details follow in Chapter 3.

Using particle filtering it is possible to support a dynamical model that enforces joint angle limits, and to disallow interpenetration of limbs e.g. [DNBB99, BSB05, SBB10]. It is also possible to support a multimodal observation likelihood. This is demonstrated experimentally by Balan *et al.* [BSB05] who apply a particle filter (with silhouette based observation likelihood) to human motion tracking from multiocular observations. They find that particle filtering is able to support, and eventually to resolve, temporary multimodality in the posterior. This is in contrast to a similar annealing-based approach that concentrates particles around a *single* pose interpretation, occasionally leading to tracking failure.

In theory, given enough particles, long periods of ambiguous observation data might be supported by simultaneously maintaining a large range of alternative pose interpretations (wide, dense, multimodal particle distribution) until such time as observations allow the true pose parameters to be resolved. However, the number of particles required to sample a given state space with constant lattice spacing is exponential in the dimensionality of that space. For the tracking of full body human motion the state space dimensionality is typically above 30 and each particle requires propagation and observation likelihood evaluation (with associated computational costs). In practice this precludes the sampling of all but a small sub-region of the state space and the capacity to maintain broad

[3] introduced to the computer vision community in the form of the CONDENSATION algorithm [IB98a].

multimodal distributions is therefore reduced. Furthermore, particle filtering has been shown to collapse into a unimodal posterior even where weightings are kept (artificially) flat, in a process known as "sample impoverishment" [KF00]. Experimental investigations have found standard particle filtering to fail after only seconds when restricted to observations from fewer than three cameras [BSB05]. Consequently, considerable work has been done on how best to spread samples within the state space, or "smart sampling". This work has broadly taken two routes: (i) *high-dimensional approaches* – that attempt to confine sampling to pertinent portions of the ambient state space; (ii) *low-dimensional approaches* – that learn a low-dimensional latent pose space from training data and perform sampling in the latent space. These approaches have parallels with example-based and learning-based discriminative techniques, respectively.

2.3.2.1 High-Dimensional Approaches

In high-dimensional approaches, inference is undertaken in the body model's full or ambient pose space. This is challenging because of the high dimensionality of this space. The most straight forward way in which to constrain the ambient state space is to impose maximum and minimum limits beyond which joints cannot rotate. Such constraints are simple to enforce in particle-based tracking approaches, with particles propagated into illegal regions of the state space simply disregarded. Similar checks can be performed to ensure that limb interpenetration is not permitted [BB06], and environment interactions such as surface contact respected [VSJ08]. If joint limits are learned from training data, e.g. walking data in [BSB05], they can serve as a restrictive prior that helps to constrain the tracking problem, but precludes the tracking of freeform motions not present in the training data. This is similar to the approach of Caillette *et al.* [CGH05] where high-dimensional pose data is clustered and particles restricted to the vicinity of the learned clusters during tracking. More recent work by Sigal *et al.* [SBB10] has employed a less restrictive set of anatomical joint limits. The authors find that tracking fails rapidly when such constraints are lifted. More complex approaches to joint rotation modelling have been developed in the robotics and biomechanical

communities, e.g. hierarchical implicit surface joint limits where the rotational capacity of limbs in the kinematic tree is a function of their parent's rotation [HUF04].

MacCormick and Isard [MI00] attempt to overcome the high dimensionality of the state space by "carving" it into independent low-dimensional spaces that may be separately estimated by particle filtering. In the context of an articulated body, their partitioned sampling (PS) approach is similar to the search space decomposition (SSD) of Gavrila and Davis [GD96], where the torso is localised and the result used to constrain the search for the immediate children in the hierarchical tree, i.e. the upper arms and upper legs. Navaratnam *et al.* [NTTC05] apply a similar approach in a bottom-up generative context. The most problematic aspect of such searches is that failure to locate the hierarchical root (torso) – which is often most severely affected by self occlusions – can lead to complete failure. A quantitative evaluation of PS and comparison to other techniques discussed in this section is given by Bandouch *et al.* [BEB08].

Deutscher and Reid [DR05] address the high dimensionality of the state space by attempting only to recover the single pose that maximises the observation likelihood function at each frame, rather than propagating a full posterior approximation. The resulting annealed particle filtering (APF) algorithm employs a number of separate resampling stages at each time instant to concentrate particles into a globally optimal pose solution. A quantitative study by Balan *et al.* [BSB05] showed that where sufficiently rich observation data is available more accurate tracking is achieved versus a standard particle filter [BSB05]. However, this improvement is at the expense of the Bayesian framework and the authors also find APF to suffer during periods of ambiguous observation data, sometimes recovering an incorrect pose interpretation from which it is unable to escape. The APF algorithm formed the "baseline" in the most recent and extensive quantitative human motion tracking study in the literature [SBB10]. More details on the APF algorithm are given in Chapter 3.

In contrast to APF's single interpretation tracking philosophy other high dimensional approaches have focused on maintaining a broad representation of the posterior. Hyperdynamic sampling (HS) [ST02b] makes a modification to the observation likelihood in order to create "bumps" at the cores of local minima forcing particles to transition (via nearby saddles in the objective surface) between competing pose interpretations. Covariance scaled sampling (CSS) [ST03a] scatters particles widely before using deterministic optimisation to recover a number of local maxima in the posterior. The results are then used to estimate the shape and weighting of a number of Gaussian-like distributions which jointly approximate the full distribution. A similar approach is taken by Poon and Fleet [PF02] where the posterior gradient is followed to recover a number of good hypotheses. Cham and Rehg use a similar, purely functional, Gaussian mixture approximation to the posterior in their multiple hypothesis tracking (MHT) approach [CR99]. For the challenging case of monocular tracking specifically, kinematic jump sampling [ST03b] may be incorporated into sampling schemes to spread samples across a number of plausible 3D pose solutions that project to the same 2D observation. As noted in Section 2.2.2, the 3D to 2D projection means that these competing solutions may be well separated in terms of the ambient pose space. These techniques are among some of the most sophisticated in the literature. They are also complex – and therefore computationally expensive – and have not had the benefit of quantitative evaluation on the datasets with associated ground truth that have appeared since their introduction, e.g. [SB06a].

2.3.2.2 Low-Dimensional Approaches

In low-dimensional approaches, the idea is to recover a latent pose space from training data and conduct inference within this new low-dimensional space. The reduced dimensionality of this restricted state space means that the number of particles needed for probabilistic inference is reduced and alternative techniques such as deterministic optimisation become viable. Just as for learning-based discriminative approaches (see also Section 2.2.2) a central issue is how to recover embeddings of non-linear training data, usually in the form of motion capture

(MoCap) data. Accordingly, low-dimensional top-down generative approaches to tracking have seen a combination of non-linear dimensionality reduction techniques with the state space search techniques discussed in Section 2.3.2.1. This line of attack has facilitated many examples of 3D human motion tracking from monocular observations, e.g. [SJ04, SBF00, UFHF05, HGC+07].

Linear dimensionality reduction using PCA has proven remarkably successful for the representation of some human motions [AT04b, SBF00, UFF06b]. Demirdjian [Dem03] has also enforced more general articulated body model constraints by projecting unconstrained body model transformations onto a linear articulated motion space. Within the PCA space Sidenbladh *et al.* [SBF00] use a particle filter for tracking and Urtasun *et al.* deterministic optimisation [UFF06b]. However, investigation of the unconstrained PCA space has ultimately proven problematic. The underlying probabilistic assumption [TB99] that latent data is Gaussian distributed is too simplistic. The mean pose is often nonsensical and all regions of the resulting space that are far from latent data potentially contain "illegal" poses [Bow00]. Similarly with the evolution of learning-based discriminative approaches, learning locally linear models of pose data has become more typical. Li *et al.*, for example, use locally linear coordination (LLC) to model walking pose data and the multiple hypothesis tracking (MHT) algorithm for inference [LYST06].

Any non-linear dimensionality reduction technique employed for the purpose of pose space reduction must offer a mapping from latent space back to the original ambient space in order that hypotheses can be evaluated using the observation likelihood. Sminchisescu and Jepson [SJ04] find a non-linear embedding using Laplacian eigenmaps [BN03] and recover a mapping (sometimes referred to as an "inverse mapping") from the latent to original pose space separately, using radial basis function (RBF) regression. They are then able to perform probabilistic inference using CSS [ST03a]. In later work Lu *et al.* [LPS07] extend the Laplacian eigenmaps embedding to produce a probabilistic latent variable model with inverse mapping, the Laplacian eigenmaps latent variable model (LELVM). They use particle filtering to reconstruct 3D poses from monocular sequences.

Tian *et al.* [TLS05] use the GP-LVM [Law05] to reduce the dimensionality of 2D pose training data. The GP-LVM naturally provides a probabilistic Gaussian process (GP) mapping from latent to ambient space (see also Section 3.3.4). They use a standard particle filter for inference during tracking. Urtasun *et al.* later employed a scaled Gaussian process latent variable model (S-GPLVM) [GMHP04] in order to account for different variances within the dimensions of the training set with different length scales for the corresponding GPs [UFHF05] (see also Section 3.3.4.1). The authors are able to use straightforward gradient descent optimisation during tracking. The GP-LVM preserves *dissimilarity* in the ambient space and so while nearby points in the latent space map to nearby points in the ambient space, there is no guarantee that the reverse is true. This can lead to "wormholes" in the latent pose space of the kind seen when modelling 2D projections of the body/hand [BMS98, OG99] (see also Section 2.2.2). The choice of dynamical model therefore becomes a more important consideration, with simple noise-based dispersion of hypotheses unlikely to be adequate.

Some latent variable models have demonstrated a capacity to recover intra-activity variations in style [UFF06a], and other work has looked specifically at separating style and content for synthesis [BH00] and at modelling transitions between activities [UFGP08]. However, the central problem for low-dimensional top-down generative tracking approaches is that they are unable to generalise to new poses. Unless the motion to be tracked comprises poses featured in the training set, it cannot be recovered. As in learning-based discriminative approaches, moving away from the latent variables to other regions of the latent pose space may produce *novel* poses, but not necessarily *useful* ones. In this thesis (and in [DLC10, DLC+09]) it is argued that this limitation motivates the following two efforts: (i) the integration of low-dimensional and high-dimensional top-down generative approaches within a single framework able to recover known activity with fewer particles/cameras but also able to increase the scope of its search effort to recover novel motions (see also Section 2.4.4); (ii) the adoption of "richer" low dimensional latent variable models for tracking, such as hierarchies

of latent variables able to incorporate conditional independencies between body parts [LM07].

Hierarchical low-dimensional generative approaches have previously been proposed to provide the potential for independence between part-based latent models describing separate partitions of the state space, e.g. [KHM00, RRR09]. These are distinct from hierarchical PCA[4] and related dimensionality reduction techniques where the aim is to recover a piecewise representation of full-body or *global* training poses [BMS97, HH97]. They are also distinct from their hierarchical counterparts in the high-dimensional literature (see also Section 2.3.2.1) where the aim is to reduce the difficulty of the inference task, e.g. [MI00, GD96, BEB08]. Rather, the intention of the hierarchical decomposition is to produce a more expressive model of pose where part-based models can vary in tandem or in isolation.

Interacting with such a model manually (e.g. by clicking with a mouse) allows for the creation of novel poses not seen in the training set, and the potential for application in character animation has been noted, e.g. see [LM07] and accompanying source code. How to *automate* the search for such poses in, say, a generative tracking scenario, remains a challenging problem. Existing techniques have tended to descend the hierarchy with a set of particles [KHM00, RRR09], starting with complete coordination between part-based models and moving to complete independence. The difficulty is that pose diversity is constrained by those global poses that perform well at the top level and novelty becomes limited. Conversely, starting at the bottom of the hierarchy with a collection of uncoordinated part-based models sacrifices the benefit of longer-range correlations present in the training data. Despite their potential for originality, hierarchical models have typically been applied to known activity tracking [KHM00] and classification [HLWJ08] problems, or have relied on a final high-dimensional search step to recover unknown poses [RRR09].

[4] Although Karaulova *et al.* [KHM00] do use a *hierarchy of* hierarchical PCAs.

2.3.2.3 Dynamical Models

So far constraint of the system state ($\underline{s}_t$ in Eq. 2.1) by definition of illegal regions of the state space (see Section 2.3.2.1), or by learning low-dimensional embeddings from training data (see Section 2.3.2.2) has been covered. A discussion of the dynamical model in Eq. 2.1 has been avoided. A widespread approach, and one that is compatible with the filtering equation, is to approximate motion as a first order Markov process. Although generally inappropriate for human motions where non-linear variations must be treated as noise (see also the Kalman filter discussion in Section 2.3.2), a simple presumption of zero dynamics plus Gaussian noise (estimated from training data) has enabled multiocular probabilistic particle-based inference [DR05, BSB05]. The dynamical model is found by finite differencing training data to produce a diagonal covariance matrix composed of the maximum changes found in each dimension.

Many authors have chosen to violate the Markov assumption made in Bayesian filtering and build second order dynamical models, e.g. [PF02, SBF00, PRM00]. The use of second order autoregressive processes (ARPs) for modelling "repetitious" motion dynamics by Rittscher and Blake [RB99] and North *et al.* [NBIR00] also constitutes a second order linear-Gaussian Markov model. Balan *et al.* [BSB05] conduct quantitative human motion tracking experiments with a second order model, estimating joint angle velocities from particle movements over the last two time steps. Interestingly, they find worse performance with the second order model than with a simple (first order) noise model: the domain of allowable poses becoming quickly over constrained by the particle set's momentum.

A discussion of the use of latent pose spaces to constrain the space of allowable system states was given in Section 2.3.2.2. An alternative interpretation of this family of techniques is that restriction of inference to the latent space imposes a form of dynamical model on the *ambient* pose space, i.e. the use of a latent space in itself constitutes a dynamical model. Nevertheless, a dynamical model must be recovered for the latent space and high-dimensional techniques – although sometimes adopted [TLS05] – are not necessarily appropriate.

Non-linear dimensionality reduction techniques such as the GP-LVM [Law05] that preserve dissimilarities (rather than similarities) from the high-dimensional *pose* space can create jumps or wormholes in the latent pose space. Using linear dimensionality reduction technique such as PCA used to model non-linear activity data can lead to similar considerations. Here the result for a single activity is continuously distributed data (no wormholes) forming a non-linear manifold within the linear latent space. The mean pose may be nonsensical and indeed there is no guarantee that any poses away from the manifold are meaningful. In both instances, the presumption of a smoothly evolving latent coordinate (as in [TLS05]) that can be modelled by low level noise is not appropriate for the discontinuous latent pose space.

Estimating first order Markov transition probabilities between clusters of data has proven useful in particle-based *discriminative* approaches where low-dimensional linear models learned (at least in part) from the *image* space can also feature wormholes [OG99, HH98] (see also Section 2.2.2). Bowden [Bow00] has also shown that a first order Markov model of cluster transitions within a linear latent *pose* space can be used to produce realistic activity synthesis (each cluster is further decomposed onto its own principal components in an HPCA approach). The final model could (arguably[5]) be interpreted as a *hidden* Markov model (HMM) where states are not directly observable, but rather emit observables via some distribution over an observation space. This approach introduces the idea of "partitioned" dynamics where each observation distribution provides a local dynamical model, while a transition matrix controls the movement between component models. This is the explicit aim of the more expressive switching linear dynamical systems (SLDSs) of Pavlović *et al.* [PRCM99] where a Markov transition matrix controls movement between a number of linear dynamical systems (rather than fixed observation densities) to give a composite model of activity dynamics. This use of local *dynamical* models is reminiscent of the use of locally linear *spatial* models to represent non-linear data distributions (see also Section 2.2.2 and Section 2.3.2.2).

[5]This is not the interpretation presented in the original paper.

The recovery of a latent space can simplify the task of estimating activity dynamics, reducing the amount of training data needed, for example. This is analogous to the use of manifold learning as an intermediate stage in discriminative approaches to simplify the task of learning a mapping to the high-dimensional pose space (see also Section 2.2.2). Models have also been introduced to learn low-dimensional manifolds and dynamical processes simultaneously [MP06, WFH08, LTS07]. Wang *et al.* [WFH08] introduced the Gaussian process dynamical model (GPDM), an extension of the GP-LVM with an additional GP prior over the latent space giving $p(\underline{s}_t|\underline{s}_{t-1})$. The GPDM naturally recovers a smooth distribution of data in the latent space (see also Section 3.3.4.3). Urtasun *et al.* have used this model with deterministic optimisation techniques for human motion tracking [UFF06a] and Raskin *et al.* using an annealed particle filter [RRR08a]. Li *et al.* [LTS07] learn a piecewise linear representation of non-linear manifolds where each region has its own linear dynamical model. Their latent dynamical model is a generalisation of the SLDS [PRCM99] and they use it in combination with a multiple hypothesis tracker to recover human activity.

Smooth first order dynamics are suitable for use in the filtering equation (see also Eq. 2.1) and have been found to perform well for single activities (better than second order models) [BSB05]. Their application becomes more problematic when extended to multiple activities, however. For example, finite differencing ambient pose data for both *punch* and *kick* activities produces an "aggregated" covariance matrix that is unsuitable for tracking either activity in isolation. Noise is consistently high in the degrees of freedom relating to both the arm *and* the leg rather than one or the other. Such issues can also arise within more complex individual activities where dynamics evolve over time. This further motivates the use of a "piecewise" dynamical model such as an HMM or SLDS to capture a *range* of temporal properties.

Isard and Blake [IB98c] have shown how a particle filter may be used to incorporate multiple dynamical models, linked through a first order Markov transition matrix. The resulting "mixed state" particle filter is adopted by Deutcher *et al.* [DNBB99] to model non-linearities in human motion dynamics, such as joint

endstops. A fixed transition matrix (usually set by hand) means that one must constantly sacrifice some reasonable fraction particles to the wrong dynamical model. The danger being that transitions will otherwise be missed. Similarly, Pavlović *et al.* [PRCM99] learn SLDSs for two separate activities before combining them into a single SLDS using one transition matrix. Only a single dynamical model need be "active" at each instant, but the ability of the resulting matrix to cope with sequences featuring multiple activities relies upon those activities *sharing* one or more dynamical states.

With HMMs comes the additional benefit of well understood algorithms for the *classification* of observations both between an individual HMM's states and between multiple HMMs [Rab89]. In the context of particle-based inference the ability to classify observations (poses) between multiple HMMs means that particles need not be "wasted" on the wrong dynamical model. In the work of Wren and Pentland [WP98] groups of HMMs are learned over a common state space in order to provide classification and activity specific predictions during multiple activity tracking. Again, the ability of such a set of models to cope with sequences featuring multiple activities relies upon HMMs sharing states. This raises a subtle but important point about the segmentation of multiple activity training data.

Where multiple activity training data is continuous the system state will trace out a continuous trajectory through state space over time. Learning a piecewise representation of such a sequence – e.g. by clustering data – naturally leads to component activities sharing states. See for example the sequences of consecutive ballet moves[6] processed by Hou *et al.* [HGC+07]. This is not necessarily the case for segmented multiple activity data – that is, data that does not feature activity transitions. For example, all walking poses are spatially well-separated from jogging poses; there is no natural overlap between their segmented activity data. Creating a latent space from such data leads to two well-separated activity

[6]It would be interesting to know the effect of alternative choreographies using the same component moves on tracking accuracy.

manifolds (see also Chapter 5 for examples) and how best to capture transitions is not clear.

Where two or more activities *do* naturally share a component dynamical or spatial state, pose may evolve from that state in two or more different ways. Several approaches have appealed to higher order dynamical models to resolve this ambiguity i.e. by looking at a longer pose history it may be possible to determine the current activity and move away from a "junction" state in the appropriate way. For example, a second order SLDS has been applied to combinations of two activities [PRM00]. Agarwal and Triggs [AT04b] use activity-specific second order ARPs to propagate particles within a CSS scheme but take an interesting "soft partitioning" approach, learning a Gaussian mixture model over class centres to calculate a weighted mixture over nearby ARPs, given a particle's location. Hou *et al.* [HGC+07] use variable length Markov models [RST94] capable of automatically increasing their temporal "memory length" in ambiguous portions of the state space. The quantitative investigation presented in Appendix C suggests that disambiguation is not always possible by using longer state histories, suggesting that a multiple hypothesis estimation framework remains important.

2.4 Discussion and Conclusions

This chapter has given a brief overview of the state of the art in human pose estimation and tracking. In keeping with the findings of other reviews, e.g. [Pop07b], various opportunities for synergy between competing or seemingly unrelated approaches have arisen, e.g. generative and discriminative approaches, bottom-up and top-down approaches, classification and tracking. Furthermore, a novel area of contribution has been identified: the combination of high-dimensional and low-dimensional generative search strategies within a single framework. In the remainder of this thesis two methods for achieving this are explored: (i) the simultaneous consideration of quite separate low- and high-dimensional particle-based generative trackers; (ii) the gradual transition of inference from a global

low-dimensional latent pose space, through a number of increasingly short ranged part-based latent spaces, to the ambient pose space. This chapter concludes by briefly reviewing each of the combined approaches that have been highlighted and discussing their relevance to this thesis.

2.4.1 Generative and Discriminative

Top-down generative tracking approaches require hand initialisation and this has motivated their combination with discriminative approaches for "bootstrapping" at the first frame [SBB07] and for reinitialisation from errors [Dem04]. Sigal *et al.* [SBB07] use a mixture of regressors mapping to pose space to get an initial estimate for a generative tracking approach based on APF [DR05]. Demirdjian [Dem04] combines the results of an earlier generative tracking algorithm [Dem03] with discriminative view-based pose estimates to achieve robust tracking. The merging of these two approaches allows the tracker to recover from errors by reinitialising.

The approaches of Sigal and Black [SB06c] and Micilotta *et al.* [MOB06] discussed in Section 2.3.1 are bottom-up generative approaches that use a discriminative step to perform "lifting" from 2D to 3D. Sigal and Black achieve this by generalising the discriminative learning-based approach of Agarwal and Triggs [AT04a] to learn a mapping from 2D poses – rather than 2D silhouettes – to 3D poses. Micilotta *et al.* compare 2D upper pose estimates against a database of images with known 3D structure in an example-based discriminative step.

Importance sampling due to Isard and Blake [IB98b] represents another important contribution to uniting top-down tracking with image based discriminative techniques. Particles are drawn from a proposal distribution created based on the current observation, rather than simply from the dynamical prior, thus incorporating the *current* observation into future hypothesis creation. This approach enables automatic initialisation and tracking with fewer particles. The unification of discriminative and generative approaches, as in [SKM06b], is an important

research topic that is receiving much attention, but no contributions to this area are made in this thesis.

2.4.2 Top-Down and Bottom-Up

Bottom-up approaches (see also Section 2.3.1) are able to provide fast estimates of body pose from single images but are – with only a handful of exceptions (e.g. [SBR+04]) – limited to 2D estimates. Conversely, top-down approaches (see also Section 2.3.2) have no capacity to self initialise, but they are able to impose temporal consistency and support multiple 3D pose interpretations. One way in which top-down and bottom-up generative tracking approaches might be effectively unified is to have a bottom-up process producing "observations" for the evaluation of a top-down process's objective function. That is, the set of 2D joint positions produced by the bottom-up process might be used by a top-down generative tracker to evaluate the likelihood of 3D pose hypotheses.

Rather than extrapolating to 3D from intermediate bottom-up 2D pose estimates using *discriminative* techniques [SB06c, MOB06], a generative 3D tracker can be used to support multiple hypotheses, for example. The potential for the combination of bottom-up and top-down approaches in future suggests there is much value in work that seeks to infer 3D poses from 2D joint locations e.g. [UFHF05, UFF06a, HLF00, Tay00]. Chapter 6 (and [DLC+09]) describes a novel contribution to this class of approaches.

2.4.3 Classification and Tracking

Probabilistic particle-based inference is a dominant methodology in top-down generative 3D human motion tracking, where observation data is inevitably ambiguous on occasion. Equally, the state of the art in (the much less common) *3D* bottom-up generative tracking [SBR+04, Sig08, SBIH10] is also achieved using a variation of particle filtering: particle message passing (PaMPas) [Isa03]. The repeated application of Bayes' law (see Eq. 2.1) requires the specification

of a dynamical model that inevitably colours the resulting posterior distribution estimate [DNBB99]. In Section 2.3.2.3 the need for such a model to be activity specific and the challenges of switching between multiple models to support multiple activity tracking was discussed. It is desirable that there exists some mechanism to *classify* the current system state in order that the assumptions made during subsequent inference might be tailored appropriately:

> *It is highly desirable to develop systems where classification feeds back into the perception of motion since perception and classification are inextricably bound together.* Rittscher and Blake [RB99].

Where a state space is shared between multiple activities then HMMs provide an appropriate technique for classification and activity specific synthesis [WP98]. New human motion recognition techniques have been devised that are capable of outperforming HMMs, but these often violate the constraints of the tracking framework e.g. the consideration of past *and future* system states [SKM06a]. Where there is no "natural" overlap between activities it is not clear how transitions might be modelled. This is a subtle but important difference between training on continuous (e.g. [HGC+07]) and segmented (e.g. [SB06a]) multiple-activity data. In ambient pose space a mixed-state particle filter [IB98c] can be used to investigate "quantum leaps" between activity class dynamics via a first order Markov transition matrix. A variation on this idea is adopted in Chapter 5 (and in [DLC10]) with an extra "activity transition" class used to permit particles to flow gradually between activity manifolds in a latent pose space.

2.4.4 Low- and High-Dimensional

An inevitable consequence of multiple activity tracking for low-dimensional generative approaches is that latent spaces must be constructed for *every* activity to be recovered. Alternatively a joint latent space that contains all activities must be created. Additionally, if one accepts the conclusions of Sections 2.3.2.3 and 2.4.3 – that activity specific dynamical models are necessary – then separate

dynamical models must also be estimated. This is an unrealistic basis on which to conduct tracking and this thesis will argue it motivates the combination of high- and low-dimensional search strategies.

An "unknown" dynamical model may be defined, operating within the ambient state space and capable (with sufficient computational resources and observations) of recovering freeform motion e.g. [DR05, SBB10]. This activity model may be complemented with activity-specific latent spaces, also with their own dynamical models. Model switching can be conducted via a first order Markov transition matrix using a mixed-state particle filter [IB98c]. Importantly, these two spaces have quite different computational requirements in terms of the estimation task. Low-dimensional latent spaces require only a few particles for successful exploration while the ambient state space of the body model requires a large number. In Chapter 5 (and in [DLC10]) an approach is proposed that recovers known and unknown human motions by dynamically adjusting particle numbers to conduct inference in state spaces of differing dimensionality at appropriate computational cost.

Chapter 3

Theory and Techniques

This chapter describes each of the component techniques that are drawn together to define generative tracking approaches in later chapters. These consist of methods for solving the estimation *task and methods for learning priors on pose and dynamics, or "activity models". A number of different observation formats are introduced, motivating the introduction of novel objective functions for the* modelling *task in Chapter 4.*

3.1 Introduction

In the remainder of this thesis a number of techniques are introduced to address the problem statement defined in Chapter 1. These are all based around a *generative* approach to tracking that employs particle filtering techniques to recover pose estimates from a pre-defined state space. The state space may be the ambient high-dimensional pose space of the geometric body model, or a low-dimensional latent pose space recovered using some form of dimensionality reduction technique. A dynamical model must also be specified for the exploration of the state space. This generic combination of pose space and dynamical model is referred to as an *activity model*. In this chapter the methods used to construct activity models and to conduct inference in later chapters are reviewed, namely: particle-based Bayesian tracking (Section 3.2.1), the use of annealing (Section 3.2.2),

body model specification (Section 3.3.1), linear/non-linear dimensionality reduction (Section 3.3.2), the estimation of temporal dynamics (Section 3.4) and the form of system observations (Section 3.5).

3.2 Estimation

Generative approaches to tracking human motion must be able to cope with both non-linear motions, and non-Gaussian observation functions caused, for example, by background clutter. Particle filtering supports both these requirements, maintaining a finite number of weighted samples to approximate a conditional probability density for the pose configuration given observed data and a dynamical model. Particle filtering is reviewed below, before describing the annealing extension proposed by Deutscher and Reid [DBR00].

3.2.1 Particle Filtering

Human motion tracking problems can be formulated as the evolution of a system state $\underline{s}_t$ over time, $t = 0, 1, 2, ..., T$, described by a Markov process and observed by some sensor providing independent observations given $\underline{s}_t$. The state density $p_t(\underline{s}_t)$, or *posterior distribution*, given by $p(\underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t)$, where $(\underline{z}_0, ..., \underline{z}_t)$ is the set of all observations up until time t, may be propagated over time with the following rule [AMGC02] (a full derivation is given in Appendix A):

$$p(\underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t) \propto p(\underline{z}_t|\underline{s}_t) \int_{\underline{s}_{t-1}} p(\underline{s}_t|\underline{s}_{t-1}) p(\underline{s}_{t-1}|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t)\, d\underline{s}_{t-1}. \tag{3.1}$$

The sequential importance resampling (SIR) [AMGC02], or conditional density propagation (ConDensAtion) algorithm [IB98a], allows for the representation of a multimodal posterior distribution via a finite set of N weighted particles,

$$S_t^\pi = \left\{ (\underline{s}_t^{(1)}, \pi_t^{(1)}), ..., (\underline{s}_t^{(N)}, \pi_t^{(N)}) \right\}. \tag{3.2}$$

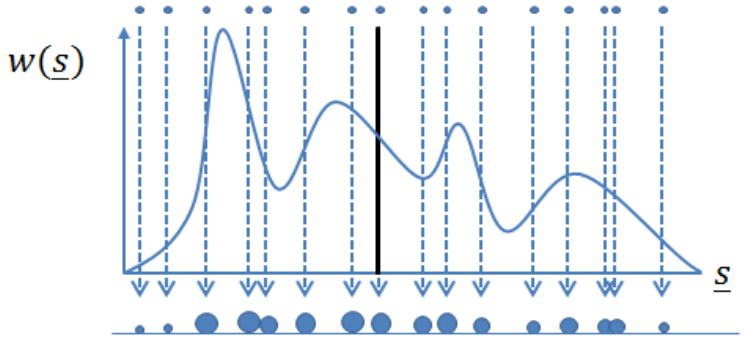


FIGURE 3.1: Visualisation of weighted particles. The location of the global maximum of $w(\underline{s})$ is not clear and calculating the expected pose (solid black vertical line) doesn't lead to a good solution.

See for example, Fig. 3.1, where particles have been spread across a 1D state space by a dynamical model and weighted in proportion to the likelihood of the observation given the corresponding system state.

After initialisation of the particle set at the point $\underline{s}_0$ (usually with ground truth), N particles are randomly sampled and dispersed by a dynamical model, $p(\underline{s}_t|\underline{s}_{t-1})$. Each new point in the state space $\underline{s}_t^{(n)}$ is evaluated using an objective function $w(\underline{z}_t, \underline{s}_t^{(n)})$ and assigned a proportional weighting $\pi_t^{(n)}$, approximating the observation likelihood $p(\underline{z}_t|\underline{s}_t^{(n)})$. Resampling then takes place, with N particles randomly sampled for dispersion from the existing distribution, with likelihood proportional to their weighting, and with replacement. In this way, the particle set may be propagated over time to maintain a representation of $p(\underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t)$. The expected pose at each instant in time t can be found by

$$\mathcal{E}(\underline{s}_t) = \sum_{n=1}^{N} \pi_t^{(n)} \underline{s}_t^{(n)}. \tag{3.3}$$

As discussed in Section 2.3.2.1 this pose estimate may be inadequate for a number of reasons, see for example Fig. 3.1 where the expected pose lies some distance from the globally optimal pose.

3.2.2 Annealed Particle Filtering

Given an observation, annealed particle filtering (APF) [DBR00] attempts only to recover the single pose that maximises the objective function. This is done by "cooling" the weighting distribution calculated at each time step and then gradually "warming" it over a number of successive resampling stages, or *layers*. The result is a slow transition from a broad and inclusive distribution over the pose space to a narrow and discriminative one. This causes resampled particles to concentrate gradually into the globally optimal mode of the objective function. See for example Fig. 3.2 which depicts the recovery of a globally optimal pose using four resampling and dispersion stages at a single time step. The posterior distribution is not fully represented – a departure from the formal Bayesian framework – but APF has been found to give good results on human motion tracking problems, outperforming SIR [BSB05, SBB10].

Resampling takes place at $r = R, R-1, ..., 0$ separate resampling layers at each time step t, where

$$w_r(\underline{z}_t, \underline{s}_t) = w(\underline{z}_t, \underline{s}_t)^{\beta_r}, \tag{3.4}$$

with $\beta_0 > \beta_1 > ... > \beta_R$. Setting the exponents too high risks particles becoming distracted by other local optima. Setting them too low means a large number of layers are required to recover an optimal pose. Deutscher and Reid [DBR00] proposed a method for the automatic selection of these parameters based on achieving a desired particle *survival rate* at each layer. The survival rate [MI00] is an approximation of the fraction of particles that will be resampled from a distribution for inclusion in the next layer,

$$\alpha_r = D_r/N, \tag{3.5}$$

where D_r is an estimate of the number of particles resampled,

$$D_r = \left(\sum_{n=1}^{N} (\pi_{t,r}^{(n)})^2 \right)^{-1}. \tag{3.6}$$

A high survival rate results in an evenly spread weighting distribution, while a low survival rate concentrates weights into just a few particles. Quantitative investigations into human motion tracking using APF [BSB05, SBB10] have found good expected tracking poses can be reliably recovered using a constant survival rate of 0.5,

$$\alpha_R = ... = \alpha_0 = 0.5. \tag{3.7}$$

APF is used for the estimation step in the generative approaches presented in this thesis. To aid exposition in later chapters the steps described by Deutscher and Reid [DBR00] for a single annealing run are summarised in Fig. 3.3.

The original APF implementation proposes a quite general first-order dynamical model $\text{func}_0(\underline{s}_{t-1})$, using the addition of Gaussian noise to approximate $p(\underline{s}_t|\underline{s}_{t-1})$. Finite differencing of training data is used to find the maximum change in each body model parameter between consecutive time steps. These values form the diagonal covariance matrix $\boldsymbol{P}_0$ of a multivariate Gaussian random variable with zero mean $\underline{n}_0 \sim N(\underline{0}, \boldsymbol{P}_0)$, that is used for the dispersion of particles. The magnitude of the dynamical model is rescaled at each annealing layer (denoted by $\text{func}_r()$ in Fig. 3.3) by multiplication of the covariance matrix by the particle survival rate α_r, to give

$$\boldsymbol{P}_r = \alpha_R \times ... \times \alpha_r \times \boldsymbol{P}_0 \tag{3.8}$$

and

$$\underline{n}_r \sim N(\underline{0}, \boldsymbol{P}_r). \tag{3.9}$$

This is in order that particle diffusion decreases at the same rate the particle set density increases, see the gradual reduction in the magnitude of particle dispersion in Fig. 3.2. The use of this activity model – ambient pose space plus Gaussian noise – during tracking is referred to as *standard APF*. Standard APF results are included as a baseline in many of the experiments presented in this thesis and also form the recently published baseline [SBB10] for the *HumanEva-II* dataset.

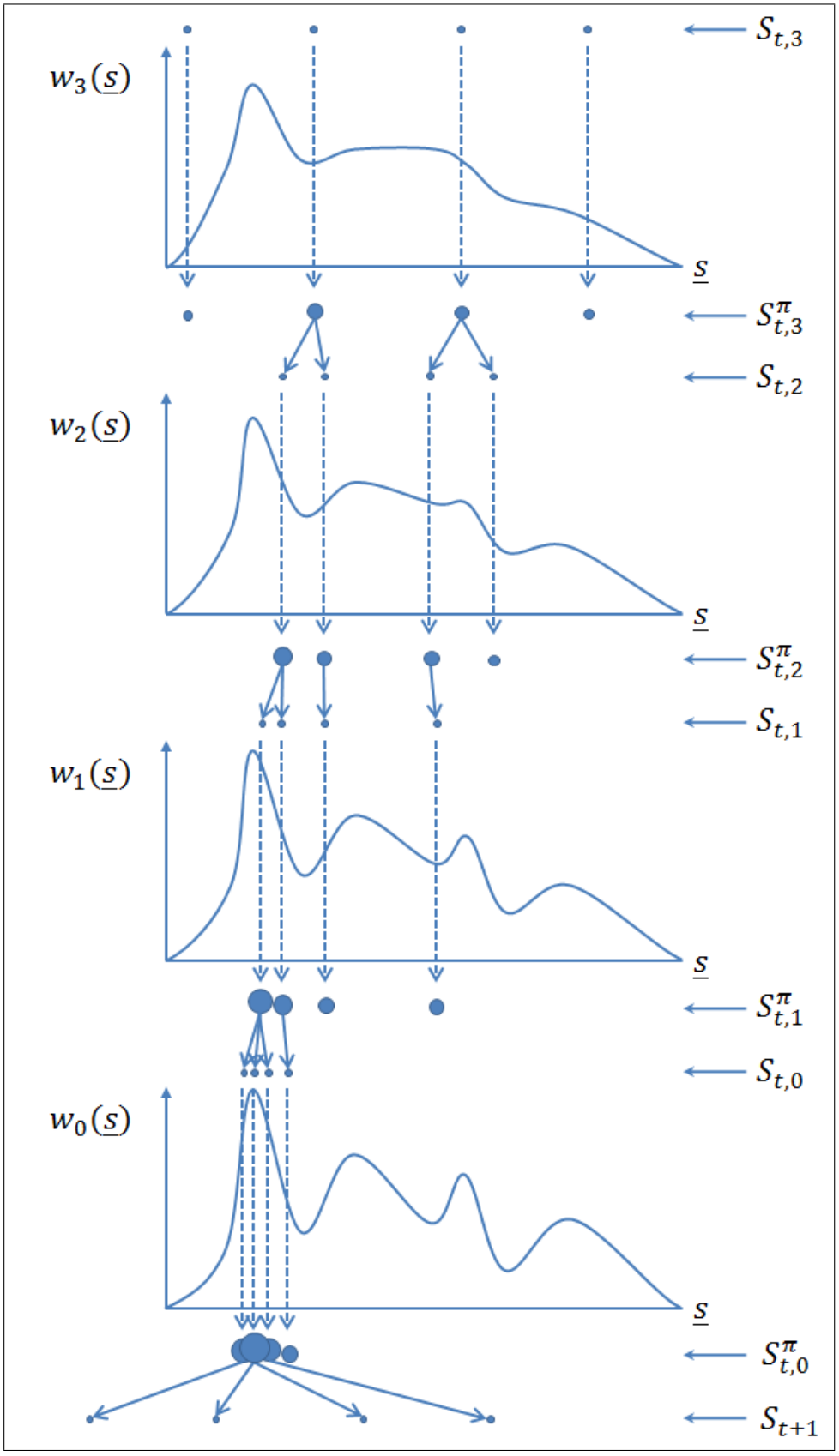


FIGURE 3.2: Visualisation of APF particle dispersion [DBR00].

1. At each time step t annealing begins at layer $r = R$.

2. The annealing run is initialised by a set of unweighted particles, $S_{t,r} = \{(\underline{s}_{t,r}^{(1)}), ..., (\underline{s}_{t,r}^{(N)})\}$. These may be the result of a previous annealing run, or the manually initialised particle set $S_{1,r}$.

3. Each particle is then assigned a weight based on the evaluation of an objective function,
$$\pi_{t,r}^{(n)} \propto w_r(\underline{z}_t, \underline{s}_{t,r}^{(n)}) \tag{3.10}$$
and the results normalised, so that $\sum_{n=1}^{N} \pi_{t,r}^{(n)} = 1$. This forms the weighted particle set,
$$S_{t,r}^{\pi} = \left\{ (\underline{s}_{t,r}^{(1)}, \pi_{t,r}^{(1)}), ..., (\underline{s}_{t,r}^{(N)}, \pi_{t,r}^{(N)}) \right\}. \tag{3.11}$$

4. The weighted particle set $S_{t,r}^{\pi}$ is then resampled to give N particles randomly drawn with a probability equal to their weighting $\pi_{t,r}^{(n)}$ and with replacement. As the nth particle is drawn, it is dispersed to produce a new unweighted particle using
$$\underline{s}_{t,r-1}^{(n)} = \text{func}_r(\underline{s}_{t,r}^{(n)}) \tag{3.12}$$
where func_r represents an arbitrary dynamical model.

5. A new set $S_{t,r-1}$ has now been recovered and is used to initialise the layer $r - 1$. Steps 3 and 4 are repeated until the set $S_{t,0}^{\pi}$ is produced.

6. The set $S_{t,0}^{\pi}$ can be used to calculate the expected tracking pose by
$$\mathcal{E}(\underline{s}_t) = \sum_{n=1}^{N} \pi_{t,0}^{(n)} \underline{s}_{t,0}^{(n)}. \tag{3.13}$$

7. A new unweighted set $S_{t+1,R}$, used to initialise the first layer $r = R$ of the next annealing run at $t + 1$ is then found by
$$\underline{s}_{t+1,R}^{(n)} = \text{func}_0(\underline{s}_{t,0}^{(n)}). \tag{3.14}$$

FIGURE 3.3: Standard APF particle dispersion, as proposed by Deutscher and Reid [DBR00]. See also Fig. 3.2.

3.3 State Space

Assuming the geometric body model is fully specified at time t by a single state vector $\underline{b}_t$ composed of D_b parameters then it is usually assumed that $\underline{s} \in \Re^{D_b}$; that is, particles reside in the same space as the system state. However, where D_b is large an attractive alternative to is to recover a low-dimensional embedding of (a portion of) the original state space (see also Section 2.3.2.2). In this scenario

particles reside in a latent space with fewer dimensions. A mapping from the latent space to the original state space exists and permits the parameterisation of the body model for objective function evaluations. In this section notation for the body model parameters $\underline{b}_t$ is introduced and techniques for the recovery of an associated latent pose space from training data are reviewed.

3.3.1 High-Dimensional "Ambient" Pose Space

The state vector $\underline{s}_t$ must completely describe the configuration of some geometric model of the human body which can be projected into the image plane for comparison with observations. As mentioned in Chapter 2, a number of different options exist for the parameterisation of such a model e.g. Euler angles, quaternions and exponential maps. The themes discussed in this section are, however, quite general and refer to an arbitrary state vector $\underline{b}_t$ composed of D_b parameters, or degrees of freedom (DOFs) that can be used to completely specify a particular choice of body model at time t. In anticipation of subsequent partitioning of the state space, it is useful to write $\underline{b}_t$ in terms of a set of *position parameters* and a set of *pose parameters*. A small number of D_ω "position" parameters describe the overall location of the model in a global coordinate system,

$$\underline{\omega}_t = (\omega_t^1, ..., \omega_t^{D_\omega})^\top \tag{3.15}$$

and a larger number of D_y "pose" parameters describe the configuration of its component parts or limbs relative to one another,

$$\underline{y}_t = (y_t^1, ..., y_t^{D_y})^\top. \tag{3.16}$$

The body model's state vector is then given by the concatenation of the position and pose vectors,

$$\underline{b}_t = [\underline{\omega}_t, \underline{y}_t] = (\omega_t^1, ..., \omega_t^{D_\omega}, y_t^1, ..., y_t^{D_y})^\top. \tag{3.17}$$

where $D_\omega < D_y$ and $D_\omega + D_y = D_b$. When referring to activity training data, the notation $\Omega = \{\underline{\omega}_1, ..., \underline{\omega}_M\}$ is used to denote a set of M *position vectors*, and $Y = \{\underline{y}_1, ..., \underline{y}_M\}$ to denote a set of M *pose vectors*.

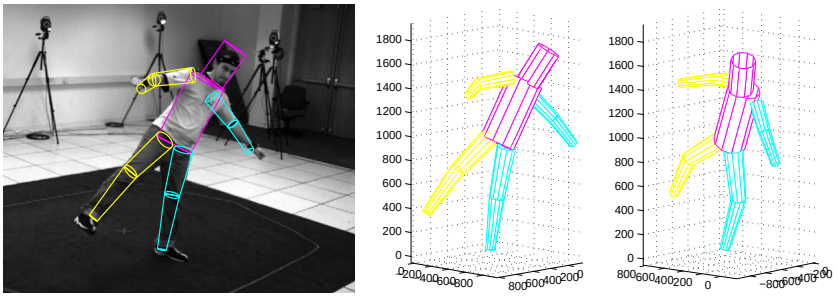
If the search for a pose solution is undertaken in the body model's ambient pose space, then particles are dispersed in a D_b-dimensional space in an attempt to recover the system state,

$$\underline{s}_t \equiv \underline{b}_t. \tag{3.18}$$

In this thesis the number of position parameters ranges between $D_\omega = 1$ for simple horizontal displacement in monocular video (e.g. Chapter 6) and $D_\omega = 6$ for full rotational and translational control (e.g. Chapter 5). Similarly the number of pose parameters ranges between $D_y = 36$ for the *HumanEva* body model [SB06a] and $D_y = 50$ for the more detailed CMU body model [CMU]. Regardless of the particular choice of parameterisation $D_\omega + D_y = D_b \geq 30$ and the use of enough particles to sample a high-dimensional space with sufficient density is required. The high-dimensional approach to estimation places no restrictions on pose but is both challenging and computationally expensive, e.g. [DBR00, BSB05, SBB10].

3.3.1.1 *HumanEva* Data

Before proceeding it is useful to introduce a specific example of body model parameterisation. Working with *HumanEva* data is an important part of the work presented in this thesis, and to track the HumanEva subjects the body model of Bălan *et al.* [BSB05] is adopted. The model itself is simple, comprising a kinematic tree of ten truncated cones but, importantly, the precise cone diameters and lengths are available for each of the *HumanEva* subjects [SB06a]. Fig. 3.4 shows a subject's body model superimposed onto a pose observation, the configuration has been calculated from MoCap markers attached to the subject's body. In this work the body model's configuration is defined using a set of $D_\omega = 6$ position parameters giving the global translation and rotation of the pelvis, and a set of $D_y = 36$ pose parameters giving the relative 3DOF Euler joint rotations



(a) (b) (c)

FIGURE 3.4: 3D body model: (a) projected into an observation for comparison; (b-c) from two rotated views.

between limbs,

$$\underline{b}_t = [\underline{\omega}_t, \underline{y}_t] = (\omega_t^1, ..., \omega_t^6, y_t^1, ..., y_t^{36})^\top. \tag{3.19}$$

These parameters can be calculated from the MoCap data in the *HumanEva-I Training* partition to give sets of *position vectors*, $\Omega = \{\underline{\omega}_1, ..., \underline{\omega}_M\}$ and sets of *pose vectors*, $Y = \{\underline{y}_1, ..., \underline{y}_M\}$. Fig. 3.5 shows series of pose vectors extracted from *walk* and *jog* activity sequences.

If this body model is used in tracking then searching for pose solutions in the body model's ambient pose space to recover the system state at each timestep requires the use of enough particles to sample a $D_\omega + D_y = 42$-dimensional space with sufficiently high density. It is precisely this kind of result that motivates use of dimensionality reduction techniques. A body model modest in its complexity and level of realism leads quickly to a high-dimensional search problem. The *HumanEva* format facilitates a number of discussions and comparisons in the remainder of this chapter. In later chapters other choices of parameterisation are made, but each is similar and takes the form of Eq. 3.17.

3.3.2 Low-Dimensional "Latent" Pose Space

To reduce the difficulty of the high-dimensional estimation task, a low dimensional latent pose space can be recovered from training data. This approach is motivated by the observation that individual degrees of freedom in high-dimensional

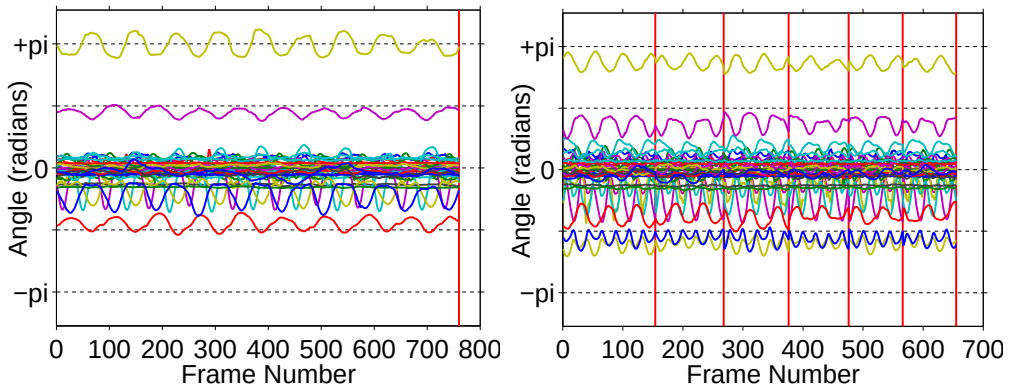


FIGURE 3.5: 36D pose vectors for known activities: (left) *walk*; (right) *jog*. Vertical red lines denote the omission of bad MoCap data.

human motion data tend not to vary in complete isolation, but are correlated and can be well described through a mapping from an underlying low-dimensional process. Recovery of low-dimensional embeddings of human activity tends to focus on pose parameters (excluding position parameters) to learn a model of activity that is independent of a training subject's position and orientation. Such a model is suitable for reuse in different tracking scenarios.

By recovering a set of latent variables $X = \{\underline{x}_1, ..., \underline{x}_M\}$ from the set of pose vectors $Y = \{\underline{y}_1, ..., \underline{y}_M\}$, each with dimensionality $D_x < D_y$, the particle filtering task can be reduced to finding a set of position parameters and *latent* pose parameters

$$\underline{s}_t = [\underline{\omega}_t, \underline{x}_t] = (\omega_t^1, ..., \omega_t^{D_\omega}, x_t^1, ..., x_t^{D_x})^\top, \tag{3.20}$$

where depending on the choice of technique as few as $D_x = 2$ dimensions often suffice for good tracking results, e.g. [SBF00, UFF06b, UFHF05]. The low-dimensional approach to estimation is less challenging and can be achieved at reduced computational expense. However, this benefit comes at the expense of flexibility; only known activities – that is activities featured in the training set – may be recovered during tracking.

Two related dimensionality reduction techniques have been widely adopted for generative tracking: (linear) principal components analysis (PCA), e.g. [UFF06b, SBF00], and the (non-linear) Gaussian process latent variable model (GP-LVM),

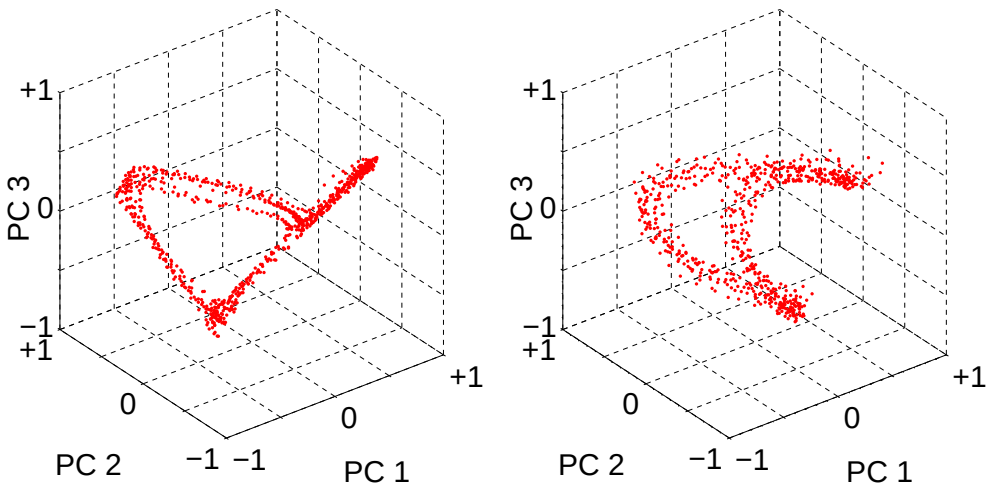


FIGURE 3.6: 3D latent pose spaces and latent variables found by PCA: (left) *walk*; (right) *jog*. Joint angle data is that shown in Fig. 3.5.

e.g. [TLS05, HGC+07]. The competing benefits of each approach are explored in the remainder of this section.

3.3.3 Principal Components Analysis

Principal components analysis (PCA) can be used to decompose the variation in a set of M pose vectors, $Y = \{\underline{y}_1, ..., \underline{y}_M\}$. The mean $\bar{\underline{y}}$ and covariance matrix $\boldsymbol{S}$ are calculated for the data and singular value decomposition used to find the eigenvectors, $\underline{\phi}_i$ and eigenvalues, χ_i of $\boldsymbol{S}$. This allows for an estimate of any data point in the training set, $\underline{y}_m$, using

$$\underline{y}_m \approx \bar{\underline{y}} + \boldsymbol{\Phi}\underline{x}_m, \tag{3.21}$$

where $\boldsymbol{\Phi} = [\underline{\phi}_1, \underline{\phi}_2, ..., \underline{\phi}_{D_x}]$ contains the first D_x eigenvectors corresponding to the largest eigenvalues, and the *latent variable* is given by

$$\underline{x}_m = (x_m^1, ..., x_m^{D_x})^\top = \boldsymbol{\Phi}^\top(\underline{y}_m - \bar{\underline{y}}). \tag{3.22}$$

In this way the training data Y are approximated by a set of latent variables $X = \{\underline{x}_1, ..., \underline{x}_M\}$, such as those shown in Fig. 3.6. Using the approximation in

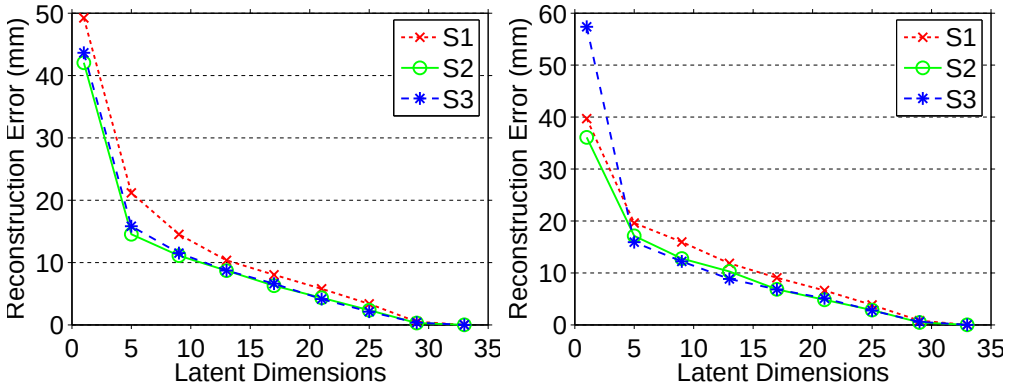


FIGURE 3.7: Average 3D absolute error between 500 original and reconstructed pose vectors using a range of PCs: (left) *walk*; (right) *jog*. Body dimensions of S4 were used for fair comparison.

Eq. 3.21, the body model can be fully specified using a single weighting vector $\underline{x}$ taken from the resulting latent pose space, and a position vector $\underline{\omega}$.

In Fig. 3.7 errors are plotted for the *walk* and *jog* pose vectors of HumanEva subjects S1-S3 (see Section 3.3.1.1 for details of the body model) reconstructed from PCA pose spaces with a range of different dimensionalities, D_x. The error measure is the same used to evaluate tracking performance and is calculated from the average distance between 15 joint centres in the original and reconstructed poses, as defined by Sigal *et al.* [SBB10][1]. Although at a given dimensionality PCA produces higher reconstruction errors than the non-linear alternatives explored in remainder of this section, these errors are still below the state of the art in generative tracking from 4 cameras [SB10] by $D_x = 4$ (around 20mm). Furthermore, in contrast to the non-linear alternatives, this 4D linear pose space has negligible computation time and a simple bi-directional mapping to the high-dimensional ambient pose space. Where there is a significant quantity of training data or where calculating mappings to and from ambient space is necessary, PCA is favoured, e.g. Chapter 4 and Chapter 5. Where the amount of training data is small and the mapping only operates in one direction, non-linear alternatives are preferred, e.g. Chapter 6. The remainder of this section reviews the GP-LVM [Law05], a non-linear latent variable model that can be derived by considering a novel probabilistic interpretation of PCA.

[1] A full definition is given in Section 3.5.4.1.

3.3.4 Gaussian Process Latent Variable Models

A probabilistic interpretation of PCA (PPCA) has been derived only relatively recently, by Tipping and Bishop [TB99]. The derivation starts from a simple probabilistic model where a series of low-dimensional latent variables are related to a series of high-dimensional training data through a matrix of linear mapping parameters $\boldsymbol{W}$ and the addition of noise. More recently, Lawrence [Law05] has developed a dual probabilistic interpretation of PCA (DPPCA) that allows for the non-linearisation of this mapping. The final result of DPPCA is simply stated here, but a more detailed derivation highlighting the duality between both probabilistic interpretations of PCA is given in Appendix B.

DPPCA gives the conditional probability of a matrix of centred (mean subtracted) high-dimensional data vectors $\boldsymbol{Y} = [\underline{y}_1, ..., \underline{y}_M]^\top$ given a centred matrix of low-dimensional latent variables $\boldsymbol{X} = [\underline{x}_1, ..., \underline{x}_M]^\top$ as

$$p(\boldsymbol{Y}|\boldsymbol{X}, \beta) = \prod_{i=1}^{D_y} N(\underline{y}_{:,i}|\underline{0}, \boldsymbol{K}), \tag{3.23}$$

where $\underline{y}_{:,i}$ is the ith *column* of $\boldsymbol{Y}$ and the matrix $\boldsymbol{K}$ is developed from the covariance between individual latent variables plus a noise term, $\boldsymbol{K} = \boldsymbol{X}\boldsymbol{X}^\top + \beta^{-1}\boldsymbol{I}$. This result can be recognised as a product of D_y independent Gaussian processes [O'H78], each being associated with a different dimension of the original high-dimensional ambient data space, and each sharing the same linear covariance function plus noise. Lawrence has shown [Law05] how DPPCA can be extended to non-linearise the mapping between latent and ambient parameters with different choices of matrix $\boldsymbol{K}$. The resulting probabilistic non-linear model is known as the Gaussian process latent variable model (GP-LVM).

3.3.4.1 Gaussian Processes

Formally, a Gaussian process (GP) is defined as a collection of random variables with the particular property that any finite subset of the collection has a joint

distribution that is a Gaussian [RW06]. More informally, a GP can be thought of as a probability distribution over functions [Law05]. Perhaps the most helpful context in which to introduce GPs is that of non-linear regression. Starting with the case of one-dimensional input and output spaces, if training data $\{y_1, ..., y_n\}$ is available for a range of input values, $\{x_1, ..., x_n\}$ how might a suitable function be fitted to the training data? Further, how might the solution be used to find a new prediction given a new input value x_*?

Adopting the analogy of Rasmussen and Williams [RW06] a suitable function can be thought of as a very long (but finite-dimensional) vector where each element contains the value $f(x)$ for a particular value of x. Regression using GPs assumes a Gaussian distribution over all "functions" that explain the training data; that is, the set of observations relate to the elements of a single vector sampled from an n-dimensional Gaussian. The GP extends (multivariate) Gaussian distributions to infinite dimensions and can be used to describe any number of new instantiations of the function.

The GP is defined by a *mean function* (presumed to be zero here) and a *covariance function*; both are functions of the input space. The form of the covariance function may be tailored to provide results that satisfy prior beliefs about the function, e.g. that it is smooth. The top row of images in Fig. 3.8 shows visualisations of a number of different covariance functions as greyscale images; elements that co-vary strongly appear lighter. Prediction at a new point x_* involves calculation of the posterior distribution $p(x_*|x_1, ..., x_n)$, which is given by a Gaussian distribution. Predictions conditioned on training data are consistent, regardless of the number of input values that are queried. The results of such queries are also consistent with all other such finite queries. The covariance function usually contains a number of parameters, or "hyperparameters", upon which predictions are also conditioned. The values of hyperparameters can be inferred from training data and so GP regression is often referred to as being non-parametric.

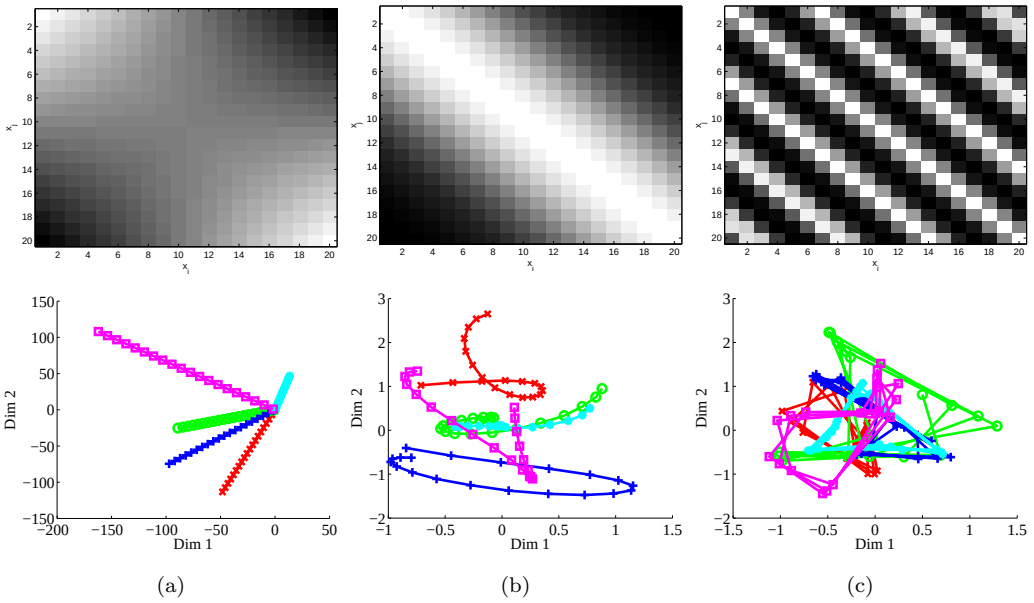


FIGURE 3.8: Priors over latent space: (a) linear; (b) RBF; (c) periodic. Covariance matrices are shown as greyscale images (top row) and five example samples (bottom row).

The covariance function for a GP prior over linear functions corrupted by noise is given by

$$k(\underline{x}_i, \underline{x}_j) = \underline{x}_i^\top \underline{x}_j + \beta^{-1}\delta_{ij}, \tag{3.24}$$

where $\underline{x}_i$ and $\underline{x}_j$ are vectors from the input space. If these inputs are taken from the matrix, $\boldsymbol{X}$, and Eq. 3.24 used to calculate the covariance between each of the M points, then the following covariance matrix is recovered

$$\boldsymbol{K} = \boldsymbol{X}\boldsymbol{X}^\top + \beta^{-1}\boldsymbol{I}. \tag{3.25}$$

This can be recognised as the linear kernel with noise from Eq. 3.23: PCA is a product of GPs, each with a linear covariance function. Fig. 3.9(a) shows example functions produced using the linear kernel. Each "function" in fact consists of 200 individual points given by a single sample drawn from a 200D multivariate Gaussian distribution.

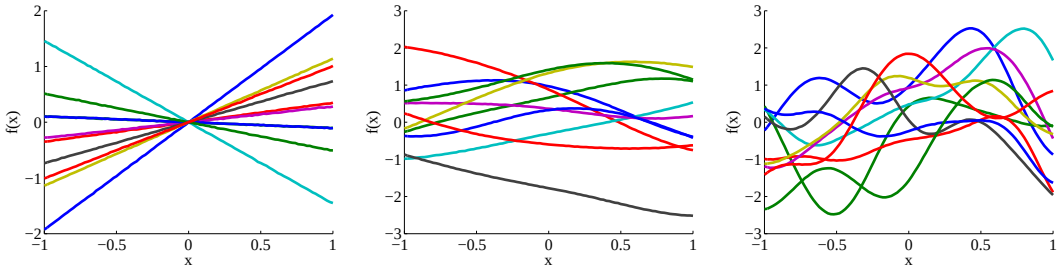


FIGURE 3.9: Functions sampled from different GPs: (a) linear kernel plus noise; (b) RBF kernel with $\alpha = 1$ and $\gamma = 1$; (c) RBF kernel with $\alpha = 1$ and $\gamma = 10$, reducing the horizontal length scale.

The key contribution of the GP-LVM is to replace the linear kernel with a non-linear alternative to produce a non-linear model,

$$\boldsymbol{K} = ? \tag{3.26}$$

A popular choice is the radial basis function (RBF) kernel (plus noise) which ensures that nearby points are well correlated,

$$k(\underline{x}_i, \underline{x}_j) = \alpha \exp\left(-\frac{\gamma}{2}(\underline{x}_i - \underline{x}_j)^\top(\underline{x}_i - \underline{x}_j)\right) + \beta^{-1}\delta_{ij}. \tag{3.27}$$

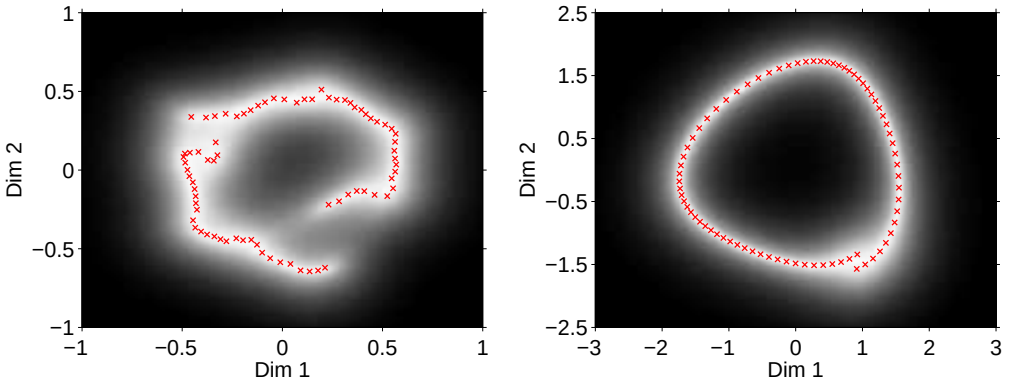
The hyperparameters α and γ control the vertical and (inverse) horizontal length scales respectively (see Figs. 3.9(b) and 3.9(c)), and their values can be inferred from the data.

3.3.4.2 Optimisation

Maximising the likelihood in Eq. 3.23 is equivalent to minimising its negative logarithm [Law05],

$$L = \frac{D_y N}{2}\ln 2\pi + \frac{D_y}{2}\ln|\boldsymbol{K}| + \frac{1}{2}\operatorname{tr}(\boldsymbol{K}^{-1}\boldsymbol{Y}\boldsymbol{Y}^\top). \tag{3.28}$$

In the case where $\boldsymbol{K} = \boldsymbol{X}\boldsymbol{X}^\top + \beta^{-1}\boldsymbol{I}$, the linear kernel with noise, it is possible to obtain a closed form solution [Law05]. The eigenvalue problem that is developed



(a) GP-LVM. (b) GPDM.

FIGURE 3.10: GP-LVMs learned from single *walk* cycle, with and without dynamics: (a) GP-LVM; (b) GPDM. Note that in Eq. 3.23 the covariance matrix is shared by all dimensions of the ambient data space. This leads to an identical level of uncertainty being associated with each dimension of reconstructed data. The figures above have been shaded to indicate the uncertainty of the mapping at each point in the latent space.

can be shown to be equivalent to that solved in PCA. However, if the aim is to account for non-linear processes by experimenting with non-linear kernels such as the RBF kernel, there will be no closed form solution and likely multiple local optima.

Here the *gradient* of Eq. 3.28 with respect to the latent points must be found then be used in combination with Eq. 3.28 in a non-linear optimiser to obtain a latent variable representation of the data. Gradients with respect to the hyperparameters of the kernel matrix (e.g. noise level, horizontal length scale, vertical length scale) may also be computed and used to jointly optimise $\boldsymbol{X}$ and the kernel's parameters. In practice latent variables are initialised using PCA and hyperparameters manually and optimisation is performed using the scaled conjugate gradient (SCG) algorithm [Møl93]. An example of the results for a single cycle of *HumanEva walk* data is given in Fig. 3.10(a).

3.3.4.3 Dynamics

An interesting extension to the GP-LVM is provided by enforcing a prior, $p(\boldsymbol{X})$ on the latent space to provide a dynamical model. To create a GP-LVM the

mappings $\boldsymbol{W}$ are marginalised (see Appendix B) and once this is done integrating out the latent space and an associated dynamical prior is not tractable. However, the dynamical model can be combined with the GP-LVM likelihood in Eq. 3.23 and an MAP solution recovered.

Wang *et al.* [WFH08] adopt this approach to enforce an *autoregressive* model of dynamics, an extra Gaussian process being used to model $p(\underline{x}_t|\underline{x}_{t-1})$ in the latent space. The resulting dynamical model – termed the Gaussian process dynamical model (GPDM) – has proven useful in tracking human motions through occlusions [UFF06a] and has the attractive property that the smooth distribution of latent variables produced by PCA initialisation tends to be preserved during optimisation. See for example Fig. 3.10(b) where the use of dynamics results in a smooth distribution of latent variables (no wormholes) that is similar to Fig. 3.6. This is useful for particle-based inference using simple Gaussian random variables for particle dispersion, e.g. [RRR08a]. The GPDM is adopted in a baseline experiment in Chapter 6.

An alternative *regressive* model of dynamics introduced by Lawrence and Moore [LM07] is also used in this thesis. Here a Gaussian process prior is placed over the latent pose space, taking as its inputs the vector of times at which the ambient sequence was observed, $\underline{m} \in \Re^{M\times 1}$,

$$p(\boldsymbol{X}|\underline{m}) = \prod_{i=1}^{D_x} N(\underline{x}_{:,i}|\underline{0}, \boldsymbol{K}_m) \tag{3.29}$$

where $\underline{x}_{:,i}$ is the ith column of $\boldsymbol{X}$ and $\boldsymbol{K}_m$ is a covariance matrix given by a covariance function such as the RBF kernel, see Eq. 3.27. Examples for a 2D latent space are shown in Fig. 3.8(b); note that nearby values are strongly correlated as evidenced by a bright diagonal in the covariance matrix and a smooth set of sample functions.

The prior in Eq. 3.29 could be combined with the GP-LVM likelihood in Eq. 3.23 to give a new model,

$$p(\boldsymbol{Y}|\underline{m}) = \int p(\boldsymbol{Y}|\boldsymbol{X})p(\boldsymbol{X}|\underline{m})\, d\boldsymbol{X} \tag{3.30}$$

but the required marginalisation is analytically intractable. However, multiplication of Eq. 3.23 by a prior on the latent variables gives a joint distribution $p(\boldsymbol{Y}, \boldsymbol{X}|\underline{m})$. This joint distribution is proportional to the posterior distribution $p(\boldsymbol{X}|\boldsymbol{Y}, \underline{m})$ and so maximising the negative log-likelihood in Eq. 3.28 plus an expression for $\log p(\boldsymbol{X}|\underline{m})$ is equivalent to seeking a *maximum a posteriori* (MAP) solution,

$$\log p(\boldsymbol{X}|\boldsymbol{Y}, \underline{m}) = \log p(\boldsymbol{Y}|\boldsymbol{X}) + \log p(\boldsymbol{X}|\underline{m}) + \text{const.} \tag{3.31}$$

The gradient of the first and second terms can be included with those of the hyperparameters Θ for joint optimisation with the SCG algorithm.

Where the GPDM's autoregressive dynamics give a unimodal prediction of $\underline{x}_t$ as a function of $\underline{x}_{t-1}$, the regressive alternative removes this relationship, permitting the trajectory of $\underline{X}$ in the latent space to cross or overlap and subsequently to bifuricate[2]. Further, the requirement for the samples $Y = \{\underline{y}_1, ..., \underline{y}_M\}$ to be equally spaced is removed. The utility of regressive dynamics is well demonstrated by the work of Andriluka *et al.* [ARS08]. In this thesis they are useful in introducing the idea of hierarchy into the construction of the latent variable model: Gaussian process priors over the ambient space are conditioned on latent variables that are in turn constrained by a Gaussian process prior conditioned on sampling intervals. In Section 3.3.5 (and Chapter 6) the simple dynamical model reviewed here is extended to a deeper hierarchy of latent variables.

3.3.5 Hierarchical GP-LVM

The H-GPLVM [LM07] allows for the incorporation of conditional independencies into latent variable models of human motion. Pose models are learned for

[2]Consider the example of a subject who walks for several cycles before breaking into a run.

individual body parts and additionally for the correlations *between* parts. The resulting model can be used to specify natural body part parameterisations either jointly or independently. The hierarchy used in this thesis is depicted in Fig. 3.11.

Following the derivation of the dynamical model in Section 3.3.4.3, a Gaussian process prior can be placed over the root node (see $\boldsymbol{X}_9$ in Fig. 3.11) to provide regressive dynamics, leading to the following marginalisation

$$\begin{aligned} p(\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_6|\underline{m}) = & \int p(\boldsymbol{Y}_1|\boldsymbol{X}_1) \times \ldots \times \int p(\boldsymbol{Y}_6|\boldsymbol{X}_6) \ldots \\ & \times \int p(\boldsymbol{X}_2, \boldsymbol{X}_3|\boldsymbol{X}_7) \times \int p(\boldsymbol{X}_4, \boldsymbol{X}_5, \boldsymbol{X}_6|\boldsymbol{X}_8) \ldots \\ & \times \int p(\boldsymbol{X}_1, \boldsymbol{X}_7, \boldsymbol{X}_8|\boldsymbol{X}_9) \ldots \\ & \times p(\boldsymbol{X}_9|\underline{m})\, d\boldsymbol{X}_9\, d\boldsymbol{X}_8\, d\boldsymbol{X}_7\, d\boldsymbol{X}_6\, d\boldsymbol{X}_5\, d\boldsymbol{X}_4\, d\boldsymbol{X}_3\, d\boldsymbol{X}_2\, d\boldsymbol{X}_1 \end{aligned} \tag{3.32}$$

where each conditional distribution is given by a Gaussian process. Just as in the previous section the necessary marginalisations are not tractable, but an MAP solution can again be found by maximising

$$\begin{aligned} \log p(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_9|\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_6, \underline{m}) = & \log p(\boldsymbol{Y}_1|\boldsymbol{X}_1) + \ldots + \log p(\boldsymbol{Y}_6|\boldsymbol{X}_6) \ldots \\ & + \log p(\boldsymbol{X}_2, \boldsymbol{X}_3|\boldsymbol{X}_7) \ldots \\ & + \log p(\boldsymbol{X}_4, \boldsymbol{X}_5, \boldsymbol{X}_6|\boldsymbol{X}_8) \ldots \\ & + \log p(\boldsymbol{X}_1, \boldsymbol{X}_7, \boldsymbol{X}_8|\boldsymbol{X}_9) + \log p(\boldsymbol{X}_9|\underline{m}). \end{aligned} \tag{3.33}$$

Following [LM07], initial estimates of the latent variables in the leaf nodes are made using PCA. Initial estimates for the parents of leaf nodes are found by applying PCA to the concatenated latent variables of their dependents. Bottom-up construction continues in this manner until the root nodes are reached. There is one root node for each activity modelled, and the latent model in each root node is a function of the latent variables of its dependents that belong to its specific activity only. In this work all latent spaces have two dimensions. The covariance function used for the dynamical model is specified by the periodic

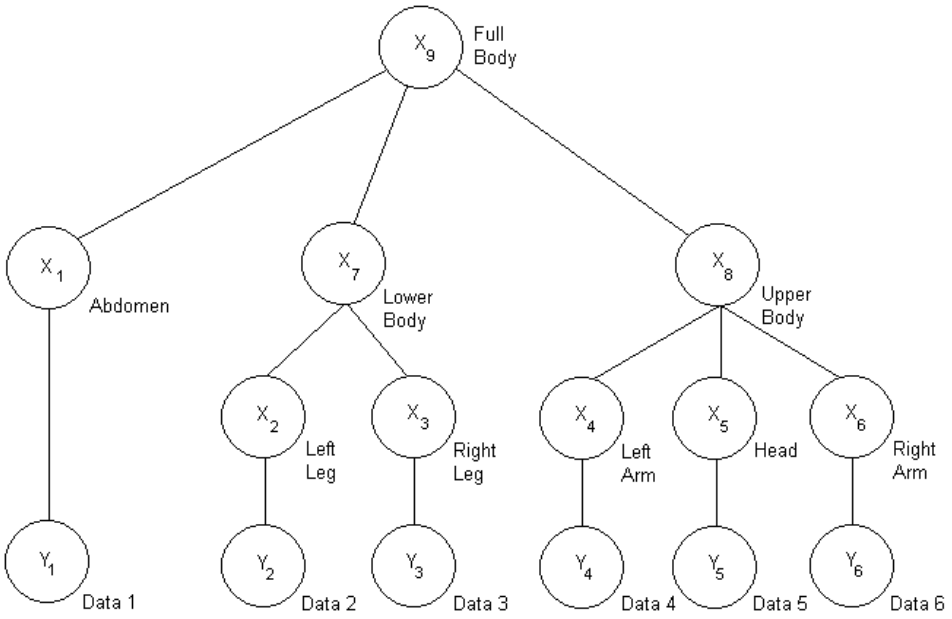


FIGURE 3.11: A hierarchy for capturing conditional independencies in the human body. $\boldsymbol{Y}_2$ is the subset of training data relating to the left leg, $\boldsymbol{Y}_3$ is that of the right leg, etc. This data is modelled by the latent variables $\boldsymbol{X}_2$ and $\boldsymbol{X}_3$ which are in turn modelled by $\boldsymbol{X}_7$.

function given by Rasmussen and Williams [RW06]

$$k_m(m_i, m_j) = \alpha \exp\left(-2\gamma \sin^2\left(\frac{m_i - m_j}{2}\right)\right). \tag{3.34}$$

Examples for a 2D latent space are shown in Fig. 3.8(c). The parameters of the dynamical model are not optimised, in order that they constrain the root node's latent space. Furthermore, for this constraint to be reflected at each layer of the hierarchy, the noise parameter β^{-1} of each Gaussian process not in a leaf node is fixed at 1×10^{-6}. Without this step optimisation can act in such a way as to remove the effect of the dynamics as the hierarchy is descended.

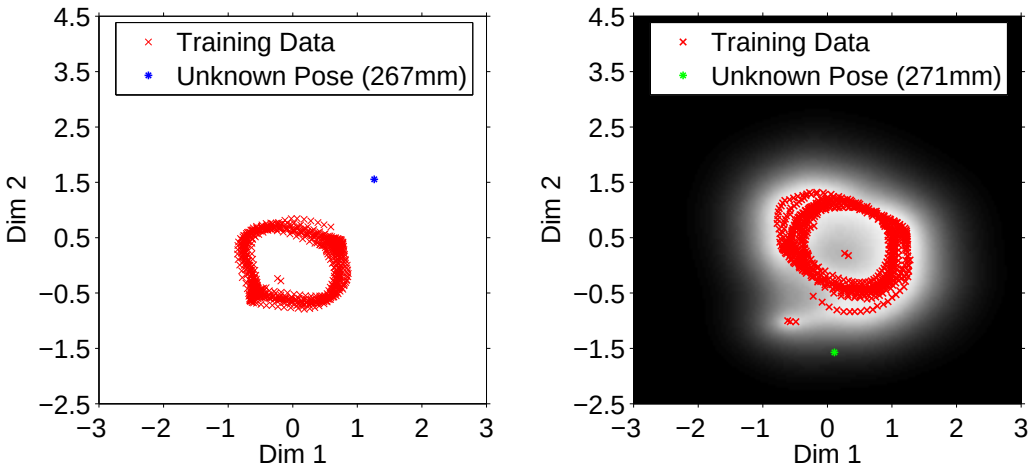
3.3.6 Generalisation

PCA and the GP-LVM have competing advantages in terms of data reconstruction accuracy and computational cost. The principal axes can be recovered by singular value decomposition of the pose vectors' covariance matrix, requiring negligible computation time. In contrast, GP-LVMs have training requirements with

complexity cubic in the number of training points. The GP-LVMs in Fig. 3.10 for example, took 580 and 500 seconds to learn from a single walking cycle (with and without dynamics, respectively) and it is generally necessary to employ a sparse representation of activity training data. However, the resulting latent variable model can be used to give lower activity reconstruction errors than PCA; see Quirion *et al.* [QDLM08] for a comprehensive comparison. Perhaps more importantly however, PCA and the GP-LVM share a common limitation: regardless of the particular choice of dimensionality reduction technique, the resulting latent pose space has only a very limited capacity to generalise beyond the training data.

To illustrate the inability of these models to generalise, a set of *HumanEva-I walk* pose vectors (of the form described in Section 3.3.1.1) were processed to recover both a 2D back constrained GP-LVM (BC-GPLVM) [LQC06] and a 2D PCA subspace. A single pose from an unknown *box* activity was then mapped into each latent space (see the "unknown pose" coordinates in Fig. 3.12) and then reconstructed by mapping *back* to the ambient pose space. Both spaces fail to preserve the *box* pose, giving high reconstruction errors; 271mm for BC-GPLVM and 267mm for PCA. Similarly, an exhaustive sampling-based approach will not result in a *box* pose being found. This result is representative, and neither the linear nor the non-linear latent space is able to generalise to substantially novel unknown poses.

This result is the key motivation for the work presented in this thesis. Latent pose spaces recovered by (possibly sophisticated, non-linear) dimensionality reduction have only a very limited capacity to generalise. The large number of low-dimensional generative tracking techniques reviewed in Chapter 2 will fail upon encountering unknown activity, continuing to recover known poses from the latent space. The benefits of these approaches are highly desirable – robust and inexpensive tracking of known activity – but further steps are necessary to avoid failure during unknown activity. This is the aim of this thesis. In Chapter 5 a method for efficiently and fairly combining separate low-dimensional and high-dimensional inference tasks to track known and unknown activities is proposed.



(a) PCA. (b) BC-GPLVM.

FIGURE 3.12: Reconstructions of an unknown *box* pose from *walk* latent pose spaces: (a) PCA latent pose space; (b) BC-GPLVM latent pose space using a multi-layer perceptron with 15 hidden nodes for back constraints. Note that neither space is able to generalise to the unknown pose. The BC-GPLVM and PCA spaces give reconstruction errors of 271mm and 267mm, respectively.

In Chapter 6 a richer pose space embedding composed of a hierarchy of non-linear latent variable models found by learning an H-GPLVM is used to recover novel poses.

3.4 Temporal Dynamics

The dimensionality reduction techniques discussed in Section 3.3 each provide a method for constraining the system state $\underline{s}$. However, PCA is not a dynamical model and steps must be taken to define $p(\underline{s}_t|\underline{s}_{t-1})$. This section investigates potential options for both pose and position parameters.

3.4.1 Finite Differencing

Given a particular choice of state space, and regardless of whether it is high- or low-dimensional, a dynamical model $\text{func}_0(\underline{s}_{t-1})$ must be specified in order to conduct particle-based estimation, e.g. by APF. One option is to use finite differencing of training data, just as in the original APF paper [DBR00]. By

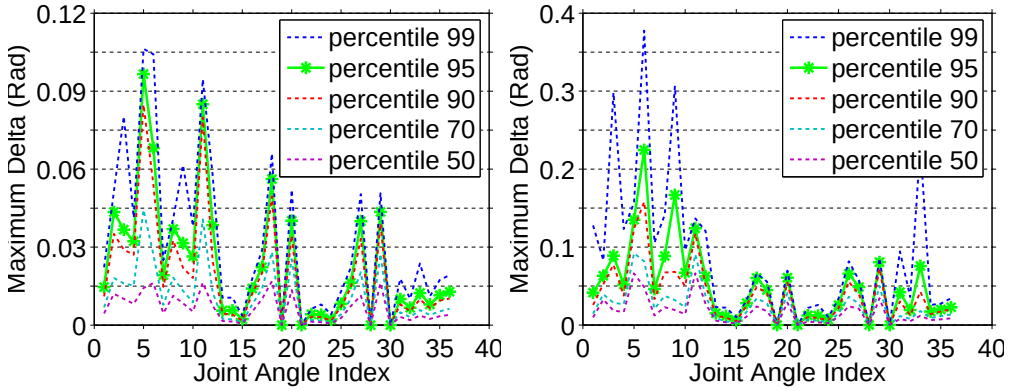


FIGURE 3.13: Maximum delta values for individual joint angles: (left) *walk*; (right) *jog*. Joint angle data is that shown in Fig. 3.5.

finite differencing sets of position vectors $\Omega = \{\underline{\omega}_1, ..., \underline{\omega}_M\}$, and pose vectors $Y = \{\underline{y}_1, ..., \underline{y}_M\}$, associated Gaussian random variables of the same form as Eq. 3.9 can be recovered,

$$\underline{n}_r^{\omega} \sim N(\underline{0}, \boldsymbol{P}_r^{\omega}) \tag{3.35}$$

$$\underline{n}_r^{y} \sim N(\underline{0}, \boldsymbol{P}_r^{y}). \tag{3.36}$$

Similarly, given a set of latent variables $X = \{\underline{x}_1, ..., \underline{x}_M\}$, (e.g., see Fig. 3.6) one can compute

$$\underline{n}_r^{x} \sim N(\underline{0}, \boldsymbol{P}_r^{x}). \tag{3.37}$$

A single anomalous jump of large magnitude in any one pose space dimension, for example due to inaccuracies in MoCap training data, can result in an overly noisy dynamical model that makes estimation difficult. For this reason the 95th percentile of delta values was used for estimation of covariance matrices. Results for a range of different percentiles in ambient and latent pose spaces recovered from *HumanEva-I* data (in the format described in Section 3.3.1.1) are shown in Figs. 3.13 and 3.14 respectively. Notice that the 99th percentile is not always a suitable representative of the data as a whole. Choosing to work with the highest single difference can lead to erratic tracking results.

Gaussian random variables are used for particle dispersion throughout the work presented in this thesis. For example, $\underline{n}_r^{\omega}$ is used to disperse position parameters in *all* experiments, and $\underline{n}_r^{y}$ to disperse pose parameters when tracking unknown

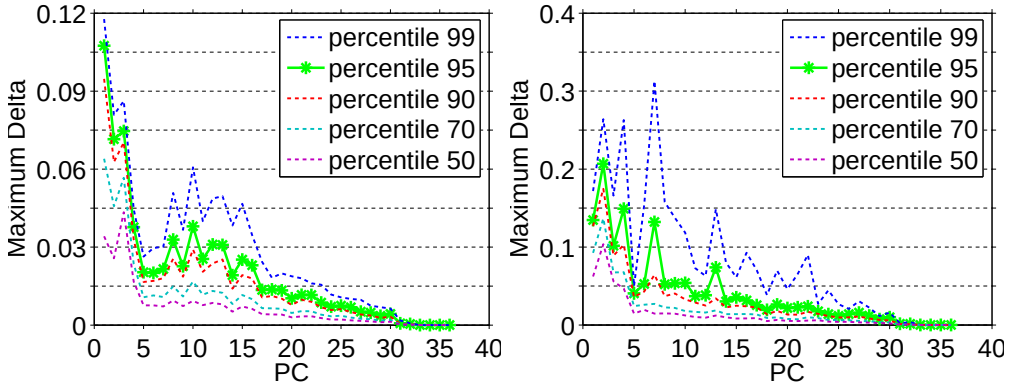


FIGURE 3.14: Maximum delta values for individual latent variable parameters: (left) *walk*; (right) *jog*. Latent variables for first three PCs are shown in Fig. 3.6.

activities in the ambient pose space (details are postponed for later chapters). Additionally, the use of $\underline{n}_r^x$ to disperse *latent* pose parameters is adopted as a baseline – termed *latent APF* – in later experiments.

Ultimately, experimental results in later chapters show that latent APF is unable to provide robust tracking of known activity, and in Section 3.4.2 a method to recover better constrained dynamical models using hidden Markov models is described.

3.4.2 Hidden Markov Models

For the GP-LVM points nearby in the latent space map to points nearby in the ambient pose space, but the reverse is not necessarily true and "quantum leaps" often occur in latent space, see for example Fig. 3.10 where consecutive walking poses have been moved far apart in latent space at two points in the activity cycle. In the case of PCA, however, latent variables $X = \{\underline{x}_1, ..., \underline{x}_M\}$ recovered from pose vector training data form smooth trajectories: points that are nearby in the ambient pose space are also nearby in the latent pose space. The resulting distributions (see for example Fig. 3.6) have two important properties: dynamics vary depending on the current location within the training manifold, and the latent space away from the latent variables may contain unrelated, even

impossible, pose configurations. Given these properties it is desirable to recover a better constrained dynamical model than the simple addition of Gaussian noise.

The hidden Markov model's (HMM) construction is ideal for modelling dimensionally reduced activity data. A human's intentions to produce movement are imprecisely realised (by their muscles) and the resulting pose configurations are then imprecisely measured (by sensors) [CBA+96]. That is, the performance of human activity is an inherently stochastic process, and the latent coordinates $X = \{\underline{x}_1, ..., \underline{x}_M\}$ constitute noisy observations of that process. HMMs allow for the description of such a doubly stochastic system [Rab89]. An HMM λ is specified by the parameters $\{S, \boldsymbol{A}, \underline{a}, p_i(\underline{x})\}$ where,

1. $S = \{s_1, ..., s_N\}$ is the set of hidden states;
2. The matrix, $\boldsymbol{A}$ is the transition matrix, where the entry A_{ij} gives the probability of a transition from state s_i to state s_j;
3. The vector $\underline{a}$ is a prior with a_i giving the probability of a sequence starting in state s_i;
4. $p_i(\underline{x})$ is the probability density associated with state s_i. In this thesis this emission probability is modelled by a single multivariate Gaussian over the latent space; $p_i(\underline{x}) = N(\underline{x}|\underline{\mu}_i, \boldsymbol{\Sigma}_i)$ with mean $\underline{\mu}_i$ and covariance matrix $\boldsymbol{\Sigma}_i$.

Fig. 3.15 shows an example of a simple three-state HMM learned over a 1D state space: the transition matrix controls movement between states and individual state probability densities control the emission of observables across a shared space.

The following steps are taken for all HMM training in later chapters. States are initialised by K-means clustering of X. The transition matrix $\boldsymbol{A}$ is set randomly (with rows normalised) and the prior $\underline{a}$ "flat" with every value equal to $1/N$. Each HMM parameter is then reestimated using no more than 50 iterations of the Baum-Welch algorithm. In order that synthesis may begin from any point

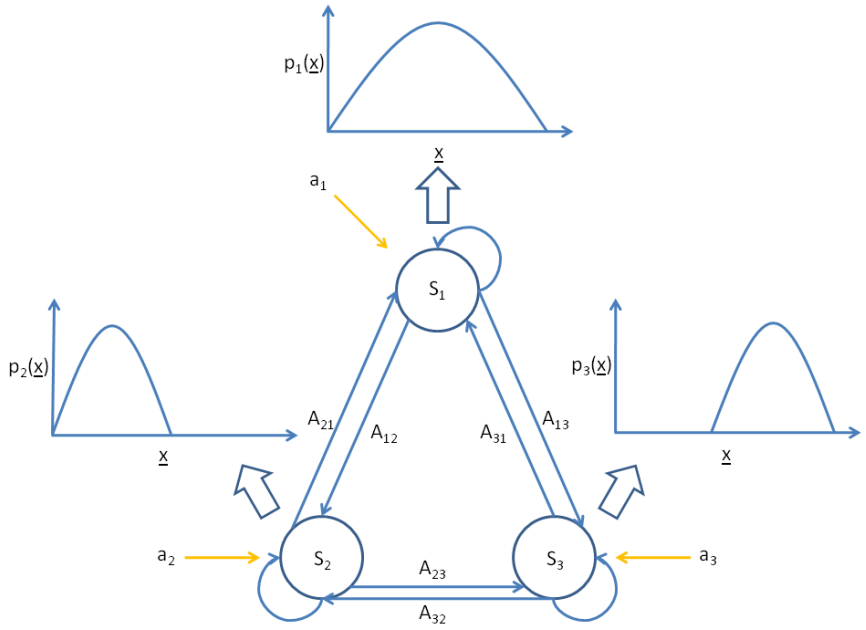


FIGURE 3.15: An example of a three state HMM with each state emitting an observable $\underline{x}$.

in the activity cycle without penalty, $\underline{a}$ is not reestimated. Appendix C gives full details of the use of HMMs with single multivariate Gaussian observation functions for training and classification.

Fig. 3.16 shows HMMs learned from the *HumanEva-I* activity data of subject S2. Using the addition of $\underline{n}_r^x$ for particle dispersion will result in a random walk through the latent pose space, while traversing an HMM will provide a spatially sensitive dynamical model and restrict pose estimates to lie close to training data. Furthermore, a *set* of HMMs can also be used to *classify* pose data. Where separate HMMs are trained to represent a set of different activities, the probability that subsequent test data were produced by each model can be evaluated and the activity classified as belonging to the most likely HMM. In this way, a set of N distinct activities can be classified using a set of N HMMs. A quantitative investigation into the accuracy of human motion data classification using HMMs is presented in Appendix D.

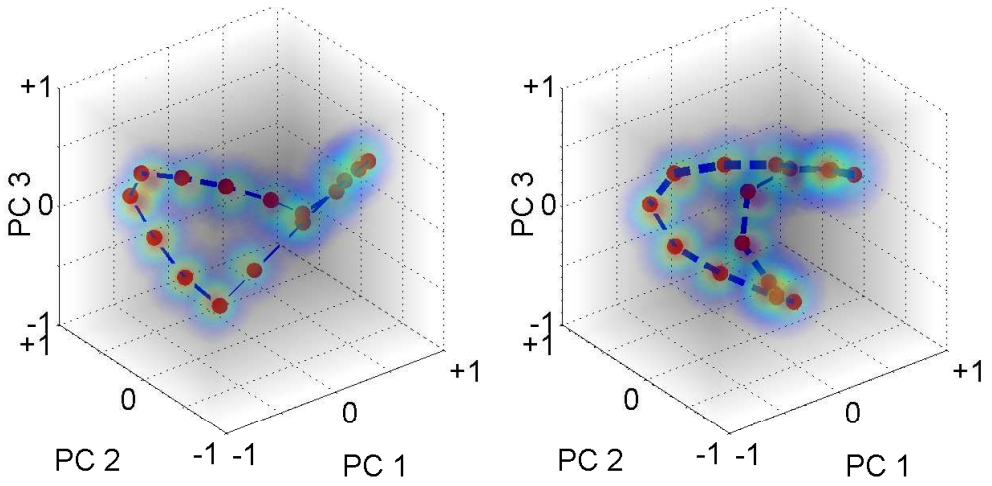


FIGURE 3.16: Visualisation of HMMs trained from latent variable distributions: (left) *walk*; (right) *jog*. State means are plotted in red, significant transitions in blue.

3.4.3 Inflating Dynamics

Bălan *et al.* [BSB05] have observed that if standard APF becomes stuck in the incorrect mode of the objective function, even if only for a few time steps, tracking may never be recovered. This is because the magnitude of the jump through state space required to recapture track quickly becomes larger than that which is permitted by the dynamical model. For this reason it can be beneficial to exaggerate the levels of diffusion produced by the dynamical model. This technique has proven beneficial in both discriminative [OG99] and generative [ST03a, Smi08] particle-based approaches, as noted in Chapter 2.

For high-dimensional spaces this ambition is particularly problematic. As the dimensionality of a state space grows, the number of samples required to sample a unit interval with constant density grows exponentially; the so-called "curse of dimensionality". Low-dimensional embeddings of the pose spaces are therefore "cheaper" to investigate and by using HMMs to provide dynamics particle propagation can be further constrained and inference made more efficient still. This more specific approach to particle dispersion where particular changes in pose are anticipated is sometimes called "smart sampling".

In the remainder of this thesis a number of methods for the inflation of activity dynamics are presented. Each dynamical model that is defined can be used to produce $p(\underline{s}_{t-1+T_0}|\underline{s}_{t-1})$ where $T_0 \geq 1$, when creating a new particle set for the next frame with Eq. 3.14. In line with the APF dispersion scaling in Eq. 3.8, steps are taken to rescale the number of synthesised time steps after each annealing layer using the survival rate α_r.

3.4.3.1 Time Reversal

Where a Gaussian random variable is estimated for use in particle dispersion the time ordering of latent variable training data is unimportant. That is, finite differencing $\{\underline{x}_1, ..., \underline{x}_M\}$ will result in precisely the same dynamical model as finite differencing $\{\underline{x}_M, ..., \underline{x}_1\}$. HMMs on the other hand *are* sensitive to time ordering of training data. Where an HMM is used for particle dispersion, it will synthesise "future" activity poses as implied by the ordering of the training data.

Smart sampling using an HMM allows particles to flow forward in time along an "activity axis" comprising a number of hidden states that give a piecewise approximation of the training data manifold: for example, compare the HMMs in Fig. 3.16 with the latent variable distributions in Fig. 3.6. If the estimation step recovers an incorrect future pose, dispersion by the HMM at succeeding time steps provides no mechanism to explore "past" activity poses and recover track. This may be especially problematic where one wishes to inflate dynamics artificially. It would be desirable to have particles able to move both forwards *and backwards* along this activity axis.

To address this problem the incorporation of a second *time reversed* transition matrix $\hat{\boldsymbol{A}}$ for use during particle dispersion is proposed. An equilibrium distribution $\underline{\psi}$ where each element ψ_i gives the probability of being in state s_i at any time t can be estimated from the transition matrix $\boldsymbol{A}$ by generating a number of transitions and recording the current state index. Given $\underline{\psi}$, the elements of a

time reversed transition matrix $\hat{\boldsymbol{A}}$ may be calculated by [Nor98],

$$\hat{A}_{ji} = \frac{\psi_i}{\psi_j} A_{ij}. \tag{3.38}$$

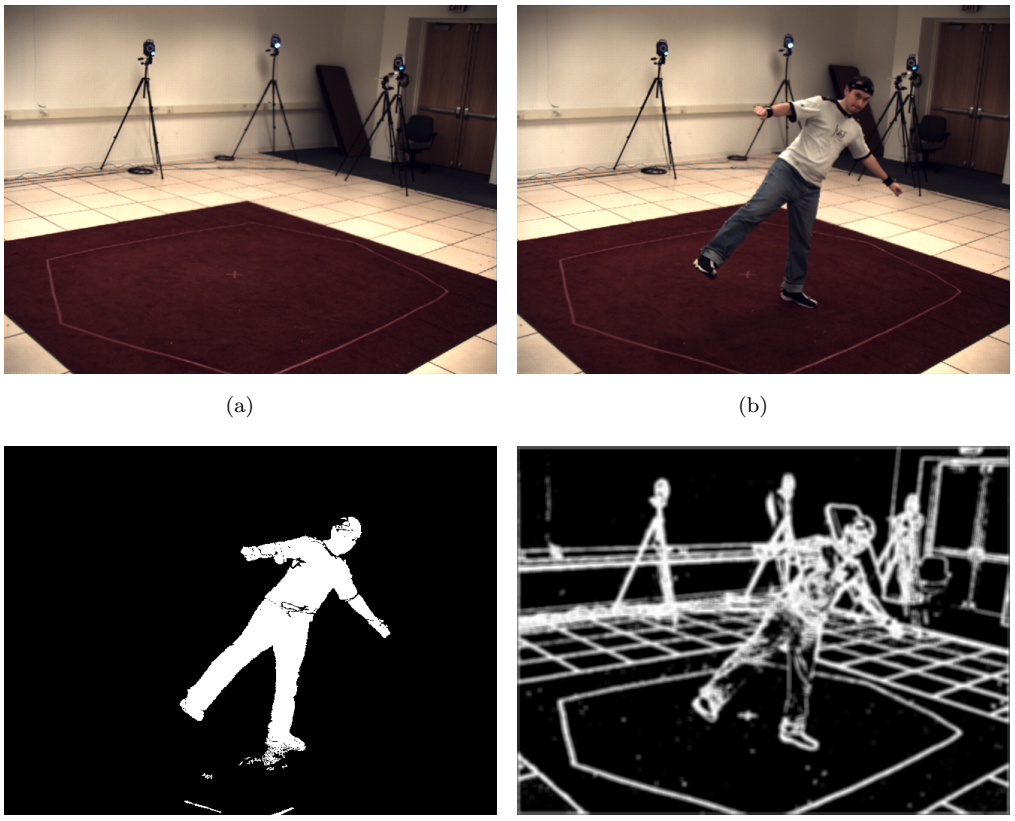
The matrix $\hat{\boldsymbol{A}}$ is used to provide a second dynamical model for the time reversed activity. Using this model for synthesis causes activity to run backwards.

3.5 Visual Cues for Activity Tracking

In addition to a dynamical model (discussed in Section 3.4), particle-based tracking also requires the specification of an objective function, $w(\underline{z}_t, \underline{s}_t)$, for assigning particle weightings. The purpose of APF is to recover the single pose that maximises the objective function given the current observation, or the *optimal pose*. This section reviews a number of valuable observation formats from which useful (discriminating) objective functions can be derived (see also Section 4.3).

The form of the objective function depends upon the image cues that can be reliably extracted. The laboratory settings used to capture *HumanEva-I* and *HumanEva-II* sequences allow for the computation of silhouette features by assuming a static background. Extra images containing the background *only* are captured (see also Fig. 3.17(a)) and used to train a Gaussian mixture model for each pixel [BSB05, SB06a, SBB10]. In more natural settings, however, such an assumption may be difficult or impossible to guarantee.

Section 3.5.1 reviews the multiocular case where multiple synchronised, well-separated sensors are available in a controlled setting. Section 3.5.2 and Section 3.5.3 consider alternatives that might be used where multiple sensors are unavailable or impractical and/or background and lighting conditions cannot be controlled. Finally the use of marker-based motion capture techniques for the provision of training data (for learning) and ground truth (for evaluation) is discussed in Section 3.5.4.



(a) (b)

(c) (d)

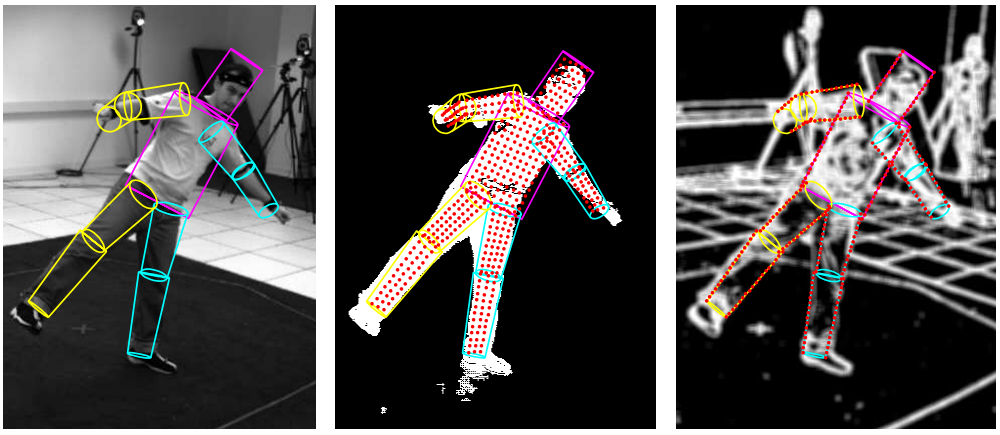
FIGURE 3.17: Silhouette and edge cues: (a) background image; (b) tracking observation; (c) background subtracted observation; (d) chamfer image.

3.5.1 Multiocular Observations

The scenario addressed by the original APF algorithm [DBR00] is that of multiocular tracking where synchronised observations are available from a number of well-separated cameras. There, and in the vast majority of other generative tracking schemes, silhouette features are used in the calculation of the objective function.

The objective function is based on a sum-squared difference (SSD) Σ^s between a binary *observation foreground mask* V^s found by background subtraction of the observation image, and a set of points $\{\xi\}$ drawn from the surfaces of the cones in the body model hypothesis projected into the image,

$$\Sigma^s = \frac{1}{|\{\xi\}|} \sum_{\xi} (1 - V^s(\xi))^2. \tag{3.39}$$



(a) (b) (c)

FIGURE 3.18: Points extracted from body model for objective function evaluation: (a) body model as projected into observation; (b) extraction of sample points from cone surfaces; (c) extraction of sample points from cone edges.

Fig. 3.17(a) shows the background image for a static camera and Fig. 3.17(b) a subsequent observation featuring a subject. Subtraction of the former from the latter can be used to find a foreground mask such as the one shown in Fig. 3.17(c). The set of points that are taken from the surfaces of the body model for use in sampling are shown in Fig. 3.18(b).

A similar measure is also used for a comparison of edge features by calculation of Σ^e. Here V^s is replaced by a *chamfer image* V^e calculated by convolution of the observation with a gradient-based edge detection mask. Results are thresholded and smoothed with a Gaussian mask before being rescaled to the interval $[0, 1]$, giving each pixel a measure of proximity to an edge. The set of points $\{\xi\}$ are drawn from the edges of the cones in the body model hypothesis projected into the image,

$$\Sigma^e = \frac{1}{|\{\xi\}|} \sum_{\xi} (1 - V^e(\xi))^2. \tag{3.40}$$

Fig. 3.17(d) shows a chamfer image calculated from the observation in Fig. 3.17(b). The set of points that are taken from the edges of the body model for use in sampling are shown in Fig. 3.18(c).

Two SSDs can then be combined and exponentiated to give a single score for a particular pose,

$$w(\underline{z}_t, \underline{s}_t) \propto \exp\left[-(\Sigma^s + \Sigma^e)\right]. \tag{3.41}$$

Or where SSD scores are available from a number of different cameras C, these measurements can be combined to give

$$w(\underline{z}_t, \underline{s}_t) \propto \exp\left[-\sum_{c=1}^{C}(\Sigma_c^s + \Sigma_c^e)\right], \tag{3.42}$$

where Σ_c^* is the SSD for camera c.

Quantitative investigations have shown that this objective function is sufficient to track slow motions such as *walk* when observations are available from *at least three* cameras [BSB05, SBB10]. Where fewer cameras are available, or where faster activities such as *jog* are observed, tracking fails. For this reason the challenging monocular and stereo tracking scenarios are a focus of the work presented in this thesis (see also Chapter 4).

3.5.2 Narrow-Baseline Stereo Observations

A narrow-baseline stereo camera provides synchronised image pairs from 2 close-mounted parallel cameras. Processed as part of a multi-camera wide-baseline tracking scheme such as APF [DBR00] the observations are so similar that their combination offers negligible benefit over monocular tracking performance. However, by calculation of the disparity between matching features in the paired images, range information for surfaces in the scene can be estimated at video frame rates, e.g. [Kon97]. The resulting depth data are sometimes referred to as 2.5D.

3.5.2.1 Ideal Stereo Model

Fig. 3.19 shows the geometry of an idealised stereo pair [KB04]. Images lie in a common plane, orthogonal to the cameras' principal rays and their horizontal

axes are shared. Any 3D point, $\underline{S}$ projects (through each camera's focal point) to a point in each image with a common vertical coordinate, $v = v'$. The difference between the horizontal coordinates $d = u - u'$ is the *disparity* of the 3D point. The disparity can be related to the distance r of the object $\underline{S}$ normal to the image plane by

$$r = \frac{f \times T_x}{d} \tag{3.43}$$

where T_x is the baseline distance. Using this relationship 2D points in the image can be reprojected to 3D points in a real world coordinate system centred about the focal point of the left camera.

In practice, cameras in stereo rigs are not well modelled by perfect pinhole imagers (as assumed above) and camera calibration is an important step in achieving good results. Calibration involves the estimation of *intrinsic* and *extrinsic* camera parameters that can be used to warp or *rectify* acquired images into idealised image pairs. Once done, horizontal lines correspond in each image and reasonable disparity estimates can be made. Calibration usually requires the imaging of a simple planar calibration target such as a checkerboard, see Fig. 3.20, and the application of a non-linear optimisation procedure.

Once a reasonable calibration has been recovered, the main challenge is to identify corresponding image elements in rectified images. The search for correlations is usually conducted between small patches of the images, and across a number of different disparity values. By setting the upper and lower values of this disparity range, one can control the nearest (highest disparity) and farthest (lowest disparity) planes at which matching can take place. This results in a 3D volume of interest or *horopter* that can be adjusted based on the particular application.

In general, stereo range data provide a relatively noisy image cue. Range *accuracy* is affected by errors in camera alignment and calibration while range *resolution* – the minimum discernable change in distance given a change in disparity, Δd – increases (deteriorates) as the square of the range [KB04],

$$\Delta r = \frac{r^2}{f \times T_x} \Delta d. \tag{3.44}$$

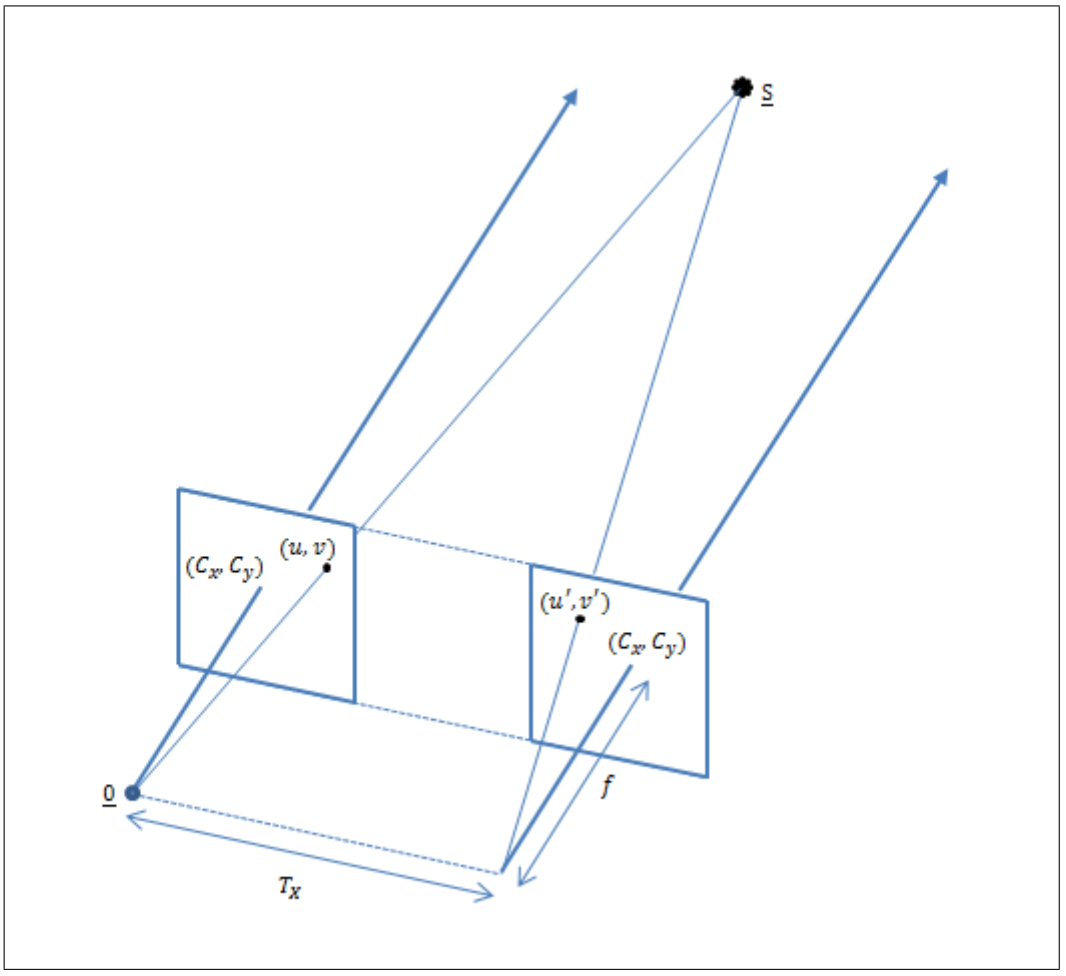
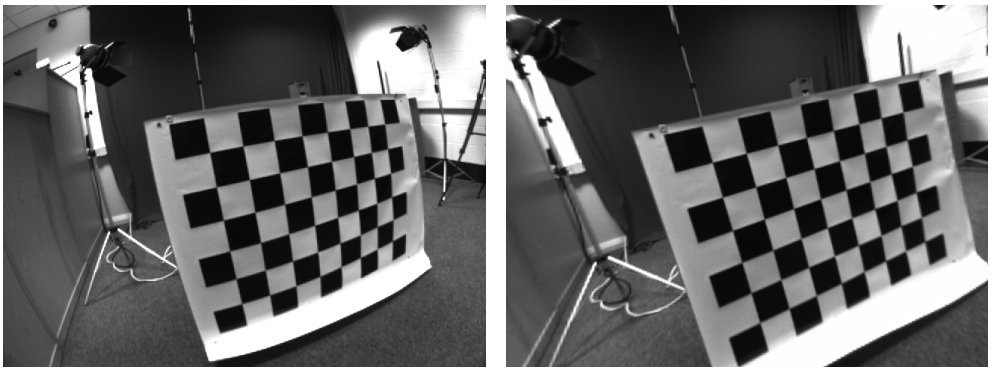


FIGURE 3.19: Ideal stereo camera geometry [KB04]. Cameras are identical, lie in a common plane, are vertically aligned, and have the same focal lengths f. Principal rays intersect the image planes at the same coordinate, C_x, C_y and a 3D point in the scene, $\underline{S}$ projects to identical *vertical* coordinates $v = v'$. By finding the *disparity* between a 3D point's horizontal projected coordinates $d = u - u'$, its distance normal to the image plane can be calculated.

Nevertheless, a stereo sensor provides disparity information *in addition* to what is essentially a monocular observation, and has a compact physical footprint similar to that of a monocular sensor. Although noisy, this range data provides precisely the kind of cue that is helpful in addressing the difficult kinematic ambiguities (e.g. "forwards/backwards flipping" [ST03b]) that arise from the 3D to 2D projection, without the need to move to a wide-baseline sensor. A depth-based objective function is proposed in Section 4.3.1.



(a) (b)

FIGURE 3.20: Camera calibration: (a) original image; (b) rectified image.

3.5.2.2 Related Work

Surprisingly few approaches have incorporated stereo range data into the human motion tracking problem. Amongst those that do, the themes of dimensionality reduction and particle-based probabilistic inference identified in Chapter 2 remain common. The iterative closest point algorithm (ICP) was used by Demirdjian to find the transformation between a set of 3D points on a body model and a set of range data coordinates [Dem03]. Articulated body model constraints were modelled by the projection of the unconstrained body model transformation onto a linear articulated motion space. Azad *et al.* [AUAD07] considered other image cues in addition to range data, segmenting the hands and head of the subject by colour and locating their corresponding 3D positions in range evidence. The result was used to constrain the state space explored by a particle filter which incorporated edge and region information into its weighting calculation. Both approaches used relatively simple body models composed of rigid primitives for limbs and were shown to track sequences of upper body movement featuring some self occlusion.

Jojic *et al.* [JTH99] used a body model described by an articulated set of 3D Gaussian "blobs". Tracking was performed using expectation-maximisation and articulated constraints enforced by an extended Kalman filter. Real time tracking of head and arm movements was demonstrated on a sequence featuring some

self-occlusion. The authors note that depth data makes possible background subtraction by range thresholding.

Plänkers and Fua employed a sophisticated deformable body model to track using range and silhouette data estimated from a narrow-baseline trinocular range sequence [PF03]. A set of Gaussian density distributions, or "metaballs" were used to form an articulated soft object model (ASOM). The form of the ASOM allows for the definition of a distance function to range data that is differentiable and so an objective function may be maximised using deterministic optimisation methods. The parameters of the ASOM were estimated in a frame-to-frame tracking stage by minimisation of the objective function given range data, before being refined by a global optimisation over all frames (not strictly fitting with the tracking scenario explored here). Remarkable upper body reconstruction results were demonstrated on sequences of a bare-skinned subject performing abrupt arm waving and featuring self occlusion. ASOMs were later used for comparison with stereo range data featuring walking and running by Urtasun and Fua [UF04]. Full body tracking was achieved by minimising the objective function with respect to the first 5 coefficients of a pre-computed pose space recovered from MoCap training data using PCA.

3.5.3 Monocular Observations

Correctly estimating pose from monocular observations is very challenging. Even where a prior model of activity is available and silhouettes can be reliably calculated, many diverse poses from the same activity can agree well with a single observation. Although this problem may be alleviated if position parameters are *also* constrained based on training data, this approach is very restrictive and is not pursued here. In Chapter 4 a silhouette-based approach to monocular tracking is investigated. The original APF weighting function (see Eq. 3.39) is extended to consider the difference in the total area of observation foreground and the total area of hypothesis foreground. By requiring strict agreement between these values the tracking of known activity from monocular observations becomes

possible (see also Section 4.4.2). However, the approach relies very heavily on good quality silhouette features.

Extracting silhouette features relies on two strong assumptions. First, the ability to generate a good background model in an offline learning step before tracking starts. Second, the ability to control the tracking environment during observation capture so that the background model remains relevant. The second assumption means ensuring a static background scene, using a stationary camera, and ensuring consistent lighting conditions (usually by filming indoors). As the number of cameras used in tracking is reduced, so the dependence on accurate background modelling for good segmentation increases.

Where only a single camera observation is available, a silhouette-based objective function (see also Section 4.3.2) *can* facilitate monocular tracking of known activity. However, such approaches will inevitably be sensitive to inaccuracies in silhouette extraction. Even subtle lighting changes – e.g. shadows cast by the subject themselves – become problematic. The ability to satisfy the strong assumptions about the tracking environment becomes unrealistic in all but the best controlled laboratory conditions and so removing this requirement is desirable. The $\mathcal{WSL}$ tracker [JFEM03] – which models the appearance of the object of interest itself rather than its surroundings – is one possible mechanism for achieving this. The $\mathcal{WSL}$ tracker provides robust 2D feature tracks that have been processed in a number of other generative tracking studies [UFHF05, UFF06a]. More generally, the tracker's output *format* – a collection of 2D joint locations – is of particular interest as it matches with that of the family of generative bottom-up tracking approaches discussed in Section 2.3.1.

3.5.3.1 The Wandering-Stable-Lost ($\mathcal{WSL}$) Tracker

The $\mathcal{WSL}$ tracker [JFEM03] maintains an adaptive model of appearance based on three components: a *stable* component learned over long time scales and based on image features that are relatively static; a *wandering* component learned over

short time scales and able to adapt to rapid changes in appearance, and to provide initialisation; and a *lost* component designed to account for outliers. In the context of human motion tracking, the example of a walking subject neatly motivates the $\mathcal{WSL}$ tracker's construction. The stable component is able to account for slowly changing appearance due to changes in 3D viewpoint e.g. as the subject changes direction. The lost component accounts for outliers due to fleeting occlusions by other objects or momentary self occlusions by other limbs e.g. as one foot swings in front of the other. The wandering component is able to cope with more rapid changes in appearance e.g. due to the reappearance of a long-term occluded arm from behind the subject's torso.

Following Jepson *et al.* [JFEM03] the individual components are introduced below in terms of a probability density for a single real-valued 1D observation z_t:

- Wandering ($\mathcal{W}$): a Gaussian density $p_w(z_t|z_{t-1}, \sigma_w^2)$ where the mean is given by the last observation and the variance is fixed.
- Stable ($\mathcal{S}$): a Gaussian density $p_s(z_t|\mu_{s,t}, \sigma_{s,t}^2)$ where $\mu_{s,t}$ and $\sigma_{s,t}^2$ are slowly varying functions of time.
- Lost ($\mathcal{L}$): a uniform distribution over the observation domain $p_l(z_t)$.

The separate strands $\mathcal{W}$, $\mathcal{S}$ and $\mathcal{L}$ are then combined through a mixture model,

$$p(z_t|\underline{q}_t, \underline{m}_t, z_{t-1}) = m_w p_w(z_t|z_{t-1}) + m_s p_s(z_t|\underline{q}_t) + m_l p_l(z_t) \tag{3.45}$$

where the mixing probabilities are given by $\underline{m} = (m_w, m_s, m_l)$ and the stable component's parameters are contained in $\underline{q}_t = (\mu_{s,t}, \sigma_{s,t}^2)$. These parameters are updated online during tracking using expectation maximisation, and the mixture model is used to provide prediction densities for new observations, z_t.

The use of a range of image features is possible with the $\mathcal{WSL}$ tracker e.g. image brightness, colour and gradient statistics. For the work presented in this thesis, the parameters introduced in the original paper [JFEM03] are adopted

FIGURE 3.21: WSL tracker results for nine independent ellipses.

and image appearance is represented through a collection of filter responses from a steerable pyramid [SFAH92, SF95]. In practice, tracking is initialised manually by a user specifying an elliptical region of interest $\mathcal{N}_t$ within the first frame of a sequence of images. The region of interest is then decomposed into a set of phase observations at a number of orientations, scales and spatial locations within $\mathcal{N}_t$ denoted by $\{z_{i,t}\}_{i\in\mathcal{N}}$. A *set* of 1D $\mathcal{WSL}$ appearance models are then applied, one for each phase signal, resulting in the collective appearance model $\mathcal{A}_t = \{\underline{m}_{i,t}, \underline{q}_{i,t}\}_{i\in\mathcal{N}}$. During tracking it is straightforward to calculate the expected probability of ownership of every phase observation by each component ($\mathcal{W}$, $\mathcal{S}$ and $\mathcal{L}$). This allows stable image elements to be emphasised when evaluating the similarity of parts of frame $t+1$ with $\mathcal{N}_t$. Only during initialisation or where rapid appearance changes occur is the $\mathcal{W}$ component expected to assume control; the system gracefully degrading to a two-frame tracker until stable features can be reestablished. Further details and derivations are given in [JFEM03].

$\mathcal{WSL}$ tracking for a number of regions of interest on a walking person are shown in Fig. 3.21. The idea is to extract single coordinates from these moving elliptical regions to give 2D estimates of a subject's $\mathcal{M}$ joint locations at each frame. This technique has previously been used by Urtasun *et al.* in a number of publications that use gradient based methods to track 3D pose changes [UFHF05, UFF06a, Urt06]. In Chapter 6 a simple distance-based objective function is used to "lift" 2D joint locations to 3D poses in a particle-based pose estimation scheme.

3.5.4 Motion Capture

So far only non-invasive or "markerless" approaches to observing the system state have been considered. Invasive marker-based motion capture (MoCap) approaches have significant advantages over markerless alternatives. The use of retro-reflective markers illuminated by multiple infrared cameras allows for accurate and reliable 3D feature extraction with no special background requirements. By using carefully defined marker placements (e.g. the Helen Hayes full body marker set), having subjects wear tight-fitting clothing and carefully recording their anatomical measurements, commercially available software (e.g. the Vicon Plug-In Gait package) can be used to extract joint centres and full 3D limb (Euler) rotations. Although not perfect[3] these results are consistently more accurate than the current state of the art in markerless tracking using large numbers of cameras (greater than 10), at around 15mm [CMG+10, SB10].

MoCap systems also have a number of drawbacks, however. They are expensive and operate only over a small capture area within a laboratory setting, see Fig. 3.22. Expert users are required to perform calibration, to manually identify markers before capture commences, and to post-process the resulting data. More fundamentally, the MoCap approach – unlike its non-invasive alternatives – requires the cooperation of the subject. Marker-based systems play two important roles in facilitating the markerless tracking techniques presented in later chapters. First, joint centre locations estimated using MoCap are used in many of the experiments as a record of ground truth. These are compared against the body model configurations recovered by the markerless algorithm to produce quantitative tracking results. Second, series of limb rotations estimated using MoCap are used as training data from which prior models of activity are learned. These activity models are then used to constrain the search for new poses during tracking.

[3] For example the hip joint can only be localised to an accuracy of around 2-10mm using the Vicon system [CCVC07]

FIGURE 3.22: Motion capture lab. Retro-reflective markers on the subject's body are returning light from the camera's flash bulb.

In adopting MoCap as a record of ground truth it is important to acknowledge that neither the estimation of joint centres (for evaluation) nor the estimation of limb rotations (for training) is exact. Further, for synchronised datasets such as *HumanEva-I* and *HumanEva-II* [SB06a, SBB10] and their predecessor presented in [SBR+04, BSB05] slight inaccuracies are also introduced when estimating the coordinate system alignment between the (non-infrared) video cameras and the (infrared) Vicon system cameras, and during the subsequent (software based) synchronisation of the data that results from each system. These datasets also feature subjects wearing "normal" loose fitting clothing (rather than skin tight body suits) to create a realistic markerless tracking scenario. From the MoCap perspective this makes initial marker attachment less precise and leaves it liable to change as the subject moves during capture. This is also true for the *HumanEva-I Training* partition from which activity models are estimated. Despite these considerations the *HumanEva* datasets are a valuable resource (especially for the cross comparison of techniques) and the creators' estimate of 20mm for optimal performance remains lower than state of the art in markerless tracking from four cameras [SB10].

3.5.4.1 Evaluation

Having discussed the various caveats with which MoCap data is adopted as ground truth, the remaining question is how best to compute the error between a body model hypothesis and the MoCap data. In this thesis the measure introduced by Sigal *et al.* [SBR+04] is adopted. This comprises the average of the 3D Euclidean distances between $\mathcal{M} = 15$ corresponding joint centre pairs in the hypothesised body model configuration $\underline{b}$, and in the ground truth MoCap pose $\underline{\tau}$,

$$\delta(\underline{b}, \underline{\tau}) = \frac{1}{\mathcal{M}} \sum_{i=1}^{\mathcal{M}} ||l_i(\underline{b}) - l_i(\underline{\tau})|| \tag{3.46}$$

where $l_i()$ returns the 3D location of the ith joint centre. Note that the dimensionality of $\underline{\tau}$ need not match that of $\underline{b}$. For example, in *HumanEva* data each limb is permitted 6DOFs in the MoCap ground truth. This means that there is no requirement for limbs to touch and so it is typically impossible to configure a kinematic tree specified by $\underline{b}$ in such a way that $\delta(\underline{b}, \underline{\tau}) = 0$.

In a particle-based approach there are in fact N pose hypotheses to evaluate at each time step[4] $S_t^\pi = \{(\underline{b}_t^{(n)}, \pi_t^{(n)})\}_{n=1}^N$. The approach taken in the remainder of this thesis is to calculate the expected pose $\mathcal{E}(\underline{s}_t)$ from the particle set (see also Eq. 3.13) and then to compare this single pose with MoCap data to give the *expected error*,

$$\Delta_E(S_t^\pi, \underline{\tau}_t) = \delta\left(\mathcal{E}(\underline{b}_t), \underline{\tau}_t\right). \tag{3.47}$$

Other measures do exist (e.g. weighted error, MAP error and optimistic error [BSB05]) but each give very similar results in the experiments presented here (see also [DLC08c, DLC08b, DLC+09] for examples). Given that the recent *HumanEva* baseline has been presented in terms of the expected error only [SBB10], the same approach is taken here.

[4]For the purposes of this discussion the particle location $\underline{s}$ has been exchanged for $\underline{b}$ to emphasise that evaluation steps – e.g. calculation of an expected pose – are conducted within the *ambient* pose space, even where particles reside in a latent pose space.

3.6 Discussion and Conclusions

The techniques described in this chapter provide the basis for contributions in later chapters. The most straightforward combination of the steps described is as follows:

1. Recover a latent pose space from activity training data by applying some form of dimensionality reduction technique.
2. Learn a dynamical model for the resulting latent variables by training an HMM.
3. Use the resulting HMM to propagate particles through the latent pose space during tracking.
4. Calculate particle weights by comparison of body model hypotheses with observations.
5. Attempt to recover a globally optimal tracking pose at each frame using annealing.

These steps are indeed implemented in Chapter 4 and used to recover known activity from stereo and monocular observations. However, more interesting contributions follow in Chapter 5 and Chapter 6 from looking at the limitations of such a scheme. What steps should be taken to model more than one known activity? How can transitions between known activities be recovered? How should sequences containing known and unknown activities be tackled? A list of specific contributions with forward references can be found in Section 1.2, and also at the beginning of each chapter.

Chapter 4

Known Activity

In this chapter a simple but effective low-dimensional generative tracking approach is introduced. A linear latent pose space is recovered from activity training data by the application of PCA. A dynamical model is then recovered by learning an HMM from the resulting distribution of latent variables. During estimation particles are dispersed through the latent pose space by the HMM in a modified form of annealed particle filtering. Robust tracking performance is demonstrated with novel objective functions designed for processing monocular, narrow-baseline stereo and wide-baseline stereo observations.

4.1 Introduction and Related Work

Research into the markerless tracking of human motion has recently benefitted from the introduction of common data sets that include ground truth motion capture (MoCap) data [BSB05, SB06a]. These have allowed for the quantitative evaluation and cross-comparison of tracking approaches. Annealed particle filtering (APF) [DBR00] and sampling importance resampling (SIR) [AMGC02] have been shown to recover pose from multiple cameras using silhouette and edge cues [BSB05, SBB10]. However, both approaches have been found to fail when limited to observations from fewer than three cameras (this result is tested in Section 4.4.2). Many distinct pose hypotheses may agree well with the available image evidence and, despite large particle numbers, ambiguous evidence causes tracking to fail.

It is therefore useful to appeal to the idea that human motion is well described by a low-dimensional subspace of the original state space (see also Chapter 2). In this chapter a new low-dimensional generative tracking approach is introduced. In light of the problem statement of this thesis, the eventual aim is to integrate this approach with a complementary high-dimensional tracking approach. Although this is not attempted until Chapter 5, the ambition informs a number of the design choices taken in this chapter. For example, principal components analysis (PCA) is chosen to reduce the dimensionality of joint angle vectors recovered from MoCap training data. More sophisticated non-linear alternatives such as the GP-LVM [Law05] are overlooked due to the expense of calculating the mappings that allow particles to flow between latent and ambient pose space (see also Section 3.3.6).

The application of PCA leads to a latent pose space that is both linear and continuous, containing many illegal configurations and making it unsuitable for direct sampling. To address this problem the dimensionally reduced data are treated as a set of noisy observations of a stochastic process, and a dynamical model and set of continuous observation density functions are learned by training a hidden Markov model (HMM) from the distribution of latent variables in the PCA space. Sampling guided by the HMM produces poses close to the training data, with the recovered observation densities precluding the sampling of illegal regions. More sophisticated higher order dynamical models have been adopted in the human motion tracking literature, for example the use of variable length Markov models [RST94] by Hou *et al.* [HGC+07], but HMMs provide a well understood classification framework [Rab89] that is ideal for use in multiple known activity *joint* latent pose spaces for both activity labelling and particle propagation.

The main contributions of this chapter are as follows:

- Use of an HMM to model a non-linear "activity axis" within a linear latent pose space recovered using PCA (Section 4.2).
- Integration of the HMM into an annealed particle filtering framework for use in particle propagation (Section 4.2.1).

- Inflation of dynamics to recover known activity from ambiguous monocular and stereo observations (Section 4.2.1.1 and Section 4.2.1.2).
- Use of a time-reversed transition matrix to synthesise both past and future pose hypotheses for robust tracking (Section 4.4.3).
- Construction of *chamfer volumes* for use in tracking with range data observations. Where chamfer image pixels hold a value proportional to their proximity to an edge, chamfer volume voxels hold a value proportional to their proximity to a surface (Section 4.3.1).
- Proposal of a novel objective function that performs "XOR-like" comparisons between hypothesis and observation foreground (Section 4.3.3).

4.2 Activity Model Definition

If the class of activity is known a priori then, as discussed in Section 3.3.2, inference during tracking can be confined to a $(D_\omega + D_x)$-dimensional space where D_ω is the dimensionality of the body model's global position vector and D_x is the dimensionality of a latent pose space recovered from training data. Where PCA is used to recover this latent pose space then low error reconstructions may be achieved with values as low as $D_x = 4$ (see also Fig. 3.7). A simple linear mapping (see also Eq. 3.21) exists from latent to ambient pose space which allows for fast parameterisation of the body model $\underline{b}_t$ with complexity independent of the number of training data.

Position parameters are typically subject to some simple set of max/min constraints; e.g. the body model must reside within the confines of the capture environment, but are otherwise free to occupy any value within this range. The estimation of the Gaussian random variable $\underline{n}_r^\omega$ from training data remains an appropriate mechanism for position parameter dispersion during tracking. This is not true for the dispersion of pose parameters, however. Latent training data typically forms a twisted, non-linear manifold away from which poses are not

guaranteed to be relevant to the original activity, or even anatomically possible. Simply estimating the parameters of a Gaussian random variable $\underline{n}_r^x$ does not constrain the movement of particles sufficiently and is unlikely to facilitate robust tracking (this claim is investigated experimentally in Section 4.4.3). As an alternative to simple noisy dispersion of particle pose parameters, the use of HMMs as discussed in Section 3.4.2, is advocated.

There is now considerable quantitative experimental evidence to show that APF is more successful than particle filtering for human motion tracking [BSB05, SBB10]. However, APF is adopted at the expense of the Bayesian framework and the annealing process can recover the wrong pose interpretation when faced with multiple maxima of approximately equal magnitude [BSB05]. For this reason, a recurring theme in this thesis is the artificial inflation of dynamics to produce $p(\underline{s}_{t-1+T_0}|\underline{s}_{t-1})$ where $T_0 \geq 1$ to facilitate recovery from tracking errors (see also Section 3.4.3).

For pose parameter dispersion via an HMM this simply means making not one, but T_0 transitions via the transition matrix before sampling from the final state's observation density. In general, hypotheses that extend too far along the activity manifold die out during annealing, discounted by comparison with image evidence in the evaluation of the objective function. However, maintenance of a wider distribution of activity pose samples permits escape from an incorrect interpretation where image evidence is ambiguous. The only question is how to set the value T_0. Two approaches are investigated here: (i) experimental determination of a constant T_0 value based on tracking accuracy (Section 4.2.1.1), (ii) a heuristic approach for dynamically adjusting T_0 during tracking that has proven particularly effective where HMMs are used in isolation (Section 4.2.1.2).

4.2.1 Known Activity (HMM-APF)

In the maximal dispersion step at $r = 0$ (see also Fig. 3.3) position parameters are dispersed by addition of the Gaussian random variable $\underline{n}_0^\omega$, estimated by finite

differencing of training data (see also Section 3.4.1). Latent pose parameters are dispersed by classification to an HMM state, followed by synthesis of T_0 transitions via the matrix $\boldsymbol{A}$. Two methods for the inflation of $T_0 > 1$ are detailed in Section 4.2.1.1 and Section 4.2.1.2.

For the recovery of an optimal pose, the magnitude of dispersion in subsequent layers $r > 0$ should decrease at the same rate as the resolution of the particle set increases [DBR00]. To achieve this the particle survival rate α_r is used to rescale the number of timesteps synthesised T_r, the covariance of the Gaussian random variable $\underline{n}_r^{\omega}$ and the covariance matrices of the HMM state observation densities, $\boldsymbol{\Sigma}_{i,r}$. Optionally, the time-reversed matrix $\hat{\boldsymbol{A}}$ may also be chosen for synthesis, allowing particles to flow both forwards and backwards along the manifold.

As the state means $\underline{\mu}_i$ are constant, the effect of rescaling the observation densities during annealing is to force samples closer to the training data. For parameter re-estimation in later annealing layers, self transitions by states are more common as T_r becomes small. Where $s_j = s_i$ dispersion is uncoupled from $\underline{\mu}_i$ and samples are drawn from a Gaussian density using the parent state's scaled covariance matrix, $\boldsymbol{\Sigma}_{j,r}$ but with $\underline{\mu}_j$ replaced by the particle's current estimate of $\underline{x}_{t,r}^{(n)}$. This results in a piecewise approximation to manifold dynamics that stops training data from dominating the choice of new pose hypotheses, allowing the objective function scores to guide refinement.

The HMM-APF particle dispersion process described above is detailed in Fig. 4.1 and a visualisation of its application to a *walk* observation is given in Fig. 4.2(a). The visualisation shows the annealing process for a single observation at a single time step. Particles can be seen gradually concentrating around a pose solution over a number of separate annealing layers. The single most important aspect of this dispersion is that particles must follow the path of the training data. Particles are *not* free to move through the latent space but must flow along a twisting (non-linear) "activity axis" that is defined by the locations of the HMM states. Initial particle locations are shown in black, intermediate locations in green and final particle locations in red. The Gaussian covariances from which

samples are drawn are shown as blue ellipses. Cyan lines depict significant HMM state transition probabilities and shifted state observation densities are depicted with dashed ellipses (see also item 4(ii) in Fig. 4.1). Recall from Section 3.2.2 that maximal dispersion is applied as a final stage just *before* moving on to process the next observation (see also Eq. 3.14) and so it is actually the *final* dispersion that is greatest in magnitude.

4.2.1.1 Constant T_0

One approach to setting T_0 is to determine an optimal constant value experimentally, based on tracking performance. Such an investigation is undertaken in Section 4.4.3 where the range of values $T_0 = 1, 2, ..., 5$ are all tested. In line with the APF dispersion rescaling (Eq. 3.8), the number of synthesised time steps is rescaled after each annealing layer using the survival rate α_r, to give

$$T_r = \lceil \alpha_R \times ... \times \alpha_r \times T_0 \rceil \tag{4.4}$$

where $\lceil \rceil$ denotes the `ceiling` operation. Note that setting $T_0 = 1$ causes $T_r = 1$ for all r, in which case no inflation is in effect and the dynamical model is that which is implied by the training data.

4.2.1.2 Dynamic T_0

HMMs consist of a set of static hidden states each with an associated Gaussian observation density, and each accounting for a particular subgroup of the latent training data. This property is an attractive one as it forces particles take up meaningful poses along the activity manifold as annealing commences at each timestep. Partitioning the pose space in this way also leads to a more expressive dynamical model; the particular temporal properties of each state (and the pose data it represents) are captured in its corresponding row of the transition matrix, $\boldsymbol{A}$. Rather than simply setting T_0 to a constant value, it may therefore be desirable to tailor the inflation of dynamics as a function of the current state.

1. The position of the (n)th particle in the rth layer is given by the position and latent parameters, $\underline{s}_{t,r}^{(n)} = [\underline{\omega}_{t,r}, \underline{x}_{t,r}]$.

2. The particle's position parameters $\underline{\omega}_{t,r}$ are updated by the addition of the Gaussian random variable $\underline{n}_r^{\omega}$,

$$\underline{\omega}'_{t,r} = \underline{\omega}_{t,r} + \sum_1^{T_r} \underline{n}_r^{\omega}. \tag{4.1}$$

3. For $r = 0$: the particle's latent parameters $\underline{x}_{t,r}$ are updated using an HMM λ trained from the distribution of latent variables in a pose space recovered from training data (Section 3.4.2). The current latent vector estimate is assigned to the state s_i most likely to have emitted it as an observable via $p_i(\underline{x})$. The HMM is then used to make T_r state transitions before emitting a new estimate $\underline{x}'_{t,r}$ via the final state's observation density $p_{j,r}(\underline{x})$.

4. For $r > 0$: dispersion takes place just as in 3 (above), but with the following additional steps taken to aid refinement:

   (i) The dispersion rescaling procedure is extended to the observation densities at each HMM state to give $p_{i,r}(\underline{x}) = N(\underline{x}|\underline{\mu}_i, \boldsymbol{\Sigma}_{i,r})$, where

$$\boldsymbol{\Sigma}_{i,r} = \alpha_R \times ... \times \alpha_r \times \boldsymbol{\Sigma}_i. \tag{4.2}$$

   (ii) Where $s_j = s_i$ dispersion is uncoupled from $\underline{\mu}_j$ and $\underline{x}'_{t,r}$ is produced using the scaled version of the parent state's covariance matrix $\boldsymbol{\Sigma}_{j,r}$, but with $\underline{\mu}_j$ replaced by the particle's current latent parameter estimate, $\underline{x}_{t,r}$. This prevents the training data from dominating the choice of new pose hypotheses, allowing the objective function scores to guide final refinement.

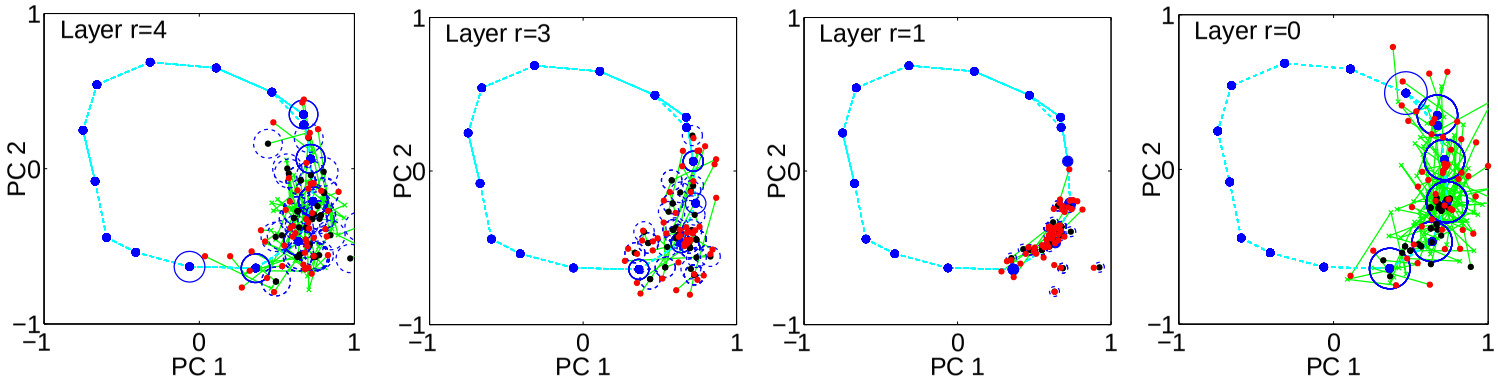
   (iii) Optionally, the transition matrix may be randomly selected as either $\boldsymbol{A}$ or the time reversed compliment $\hat{\boldsymbol{A}}$ in order to allow for the synthesis of past and future poses.

5. The new estimates are then used to create a particle in a new set

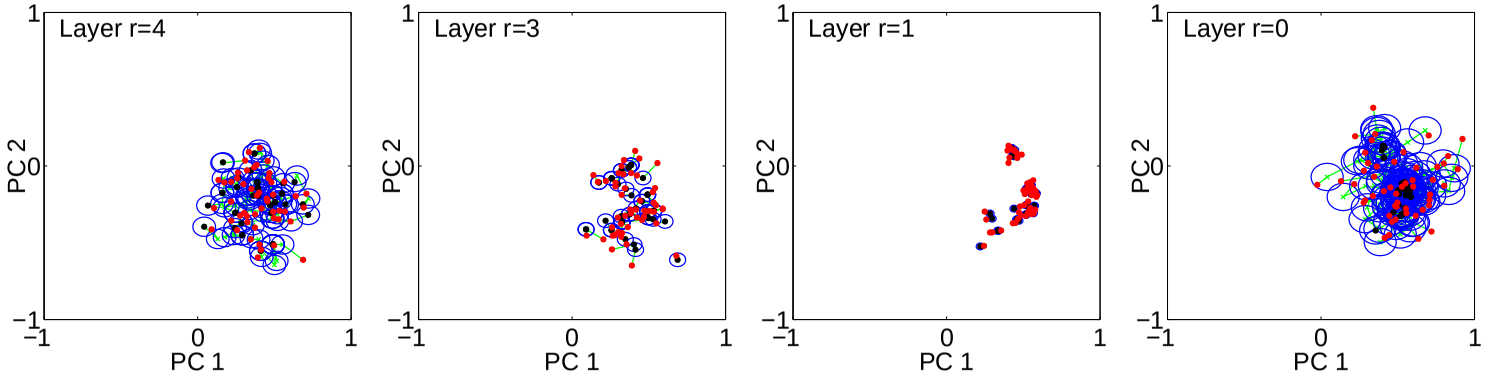
$$[\underline{\omega}'_{t,r}, \underline{x}'_{t,r}] = \begin{cases} \underline{s}_{t,r-1}^{(n)} & \text{if } r > 0; \\ \underline{s}_{t+1,R}^{(n)} & \text{if } r = 0. \end{cases} \tag{4.3}$$

FIGURE 4.1: Dispersion of a single particle for known activity: HMM-APF.

Take the example of a highly self-referential state. Self-referential states are those states that are highly likely to self transition, i.e. the matrix element A_{ii} is close to one. They arise wherever a state is responsible for modelling one or more large consecutive sequences of training data, e.g. where the training activity contains a static pose that is held for some period of time. These states are legitimate



(a) HMM-APF particle dispersion.



(b) Latent APF particle dispersion.

FIGURE 4.2: 2D view of particle dispersion over 5 layers for $T_0 = 4$ in the latent pose space. Layer $r = 2$ is omitted to maximise figure size. (a) Particle dispersion by the HMM causes particles to move along an "activity axis" ensuring relevant pose hypotheses; (b) particle dispersion by Gaussian noise produces a random walk through the latent pose space that ignores the path of training data.

products of the temporal properties of the training data, but their effect is to leave all but a small fraction of the particle set concentrated about the current state mean. The aim of inflating dynamics is to maintain an artificially wide distribution of pose candidates and therefore facilitate recovery from transient errors (due for example to poor observation data). In practice the magnitude of A_{ii} ranges widely between states and is a product of both the underlying activity, but also of parameter choices such as the number of states and initialisation values. Setting T_0 high enough that a reasonable fraction of particles escape a self-referential state is likely to result in erratic propagation elsewhere on the manifold.

To address this issue a "transition temperature" parameter ρ_T is introduced into the synthesis process. The transition temperature is a lower bound on the probability of a non-self state transition occurring and therefore a function of the current state. In line with APF dispersion rescaling the transition temperature is rescaled by the particle survival rate α_r at each annealing layer to give,

$$\rho_r = \alpha_R \times ... \times \alpha_r \times \rho_T. \tag{4.5}$$

For a particular state s_i at layer r, a non-self state transition is then ensured by making T_r state transitions where,

$$T_r = \left\lceil \frac{\log(1-\rho_r)}{\log(A_{ii})} \right\rceil. \tag{4.6}$$

Where the dependence of T_r on the state index i has been dropped in order to leave the notation in Fig. 4.1 applicable to both constant and dynamic approaches to inflation. This approach uses the temporal properties of activity data to ensure spatial variation amongst pose candidates and can be helpfully viewed as a simple form of time warping.

4.3 Objective Functions

Tracking from a minimum of four synchronised cameras in a laboratory environment can arguably be described as a solved problem. Providing good silhouette data can be extracted, Bayesian tracking techniques such as the annealed particle filter can maintain accurate 3D estimates of pose during freeform human activity performance [BSB05, SBB10]. As the number of cameras is reduced below three, however, tracking accuracy deteriorates sharply. This section introduces a number of objective functions intended for use where sensor numbers are limited (< 3). Each one is suitable for use in assigning particle weights during particle-based inference and is evaluated for use in human activity tracking in Section 4.4.

4.3.1 Range-Based

In standard APF [DBR00], a measure of agreement between edge features is calculated, Σ^e (see also Section 3.5.1 for a full discussion). This involves the detection of edges in the current image observation $\underline{z}_t$, and the convolution (or smoothing) of the resulting edge map with a 2D Gaussian kernel. The value of each pixel in the resulting image is proportional to that pixel's proximity to an edge. Such an image is also called a *chamfer image*. Pose hypotheses are then projected into the chamfer image and sample points are extracted from the edges of the component cones for the calculation of the SSD.

This approach can be extended to range data by discretising an (x, y, z) point cloud estimated by a narrow-baseline stereo camera onto a 3D grid. The data describes 2.5D *surfaces* calculated from the disparity between image pairs (see also Section 3.5.2). The data is smoothed by convolution with a 3D Gaussian kernel and the values rescaled to the range $[0, 1]$. The result is a volume V^v, where each voxel's value is proportional to its proximity to a surface, or *chamfer volume*. Chamfer volumes for a number of synthetic surfaces are shown in Fig. 4.3. A

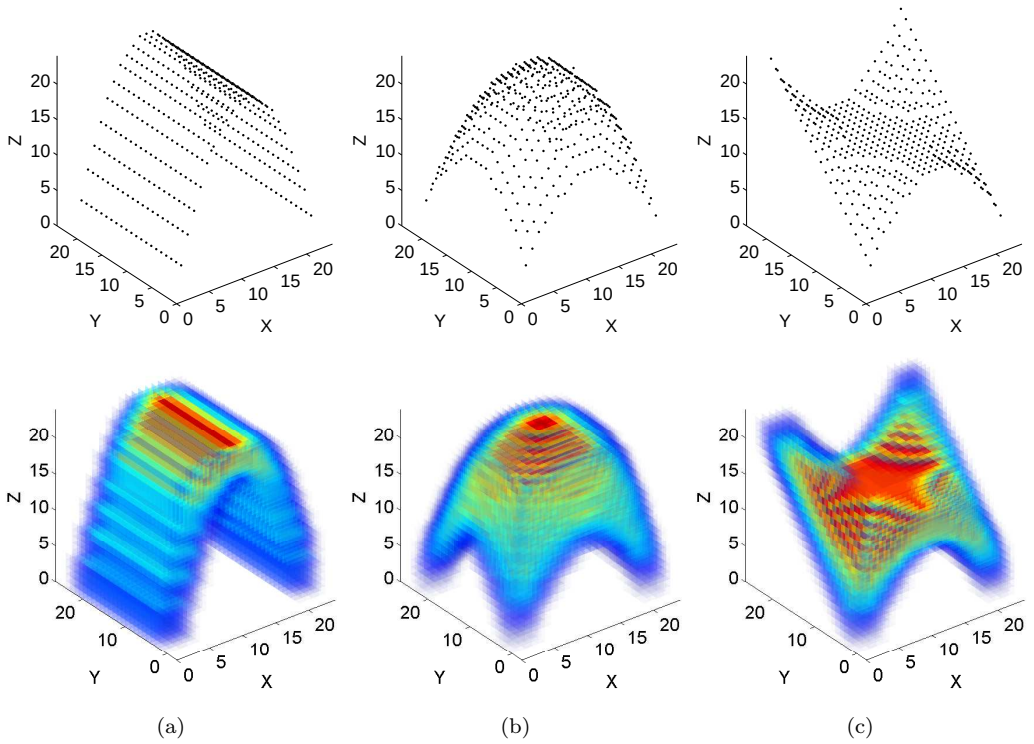


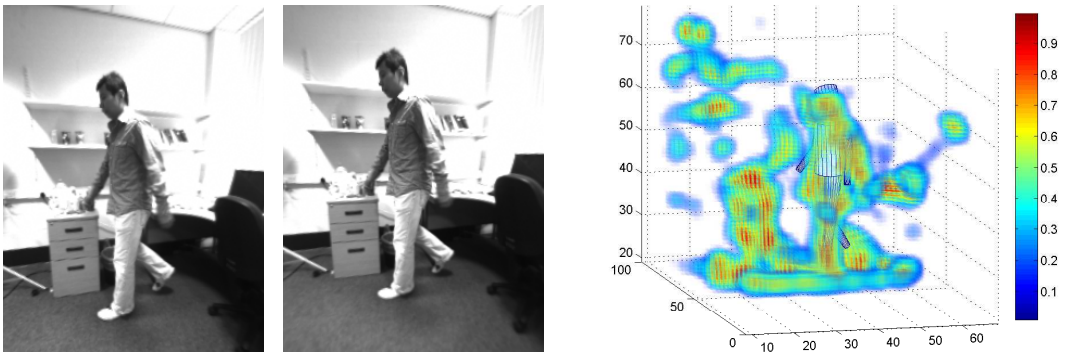
FIGURE 4.3: Example chamfer volumes: (a-c) surfaces have been discretised onto 20^3 grids and smoothed with a 7^3 spherical Gaussian kernel with $\sigma = 4$.

chamfer volume calculated from a real narrow-baseline stereo camera observation of a walking person is shown in Fig. 4.4(c).

To calculate particle weights a hypothesis $\underline{s}_t^{(n)}$ can be projected into the chamfer volume and a set of 3D sample points $\{\xi\}$ extracted from the visible surfaces of the body model's component cones. That is, portions of the cones with surface normals pointing away from the stereo camera are omitted from the calculation as are sample points occluded by other nearer cones. An SSD score can then be calculated by,

$$\Sigma^v = \frac{1}{|\{\xi\}|} \sum_{\xi} (1 - V^v(\xi))^2. \tag{4.7}$$

A visualisation of the sampling strategy is shown in Fig. 4.4(c) where the hypothesis can be seen projected into the chamfer volume. Fig. 4.5 shows the same pose hypothesis enlarged and rotated to show sample points (blue lines depict the camera's principal ray). Samples from regions of the model with surface normals pointing towards the camera are denoted by circles; those that are not



(a) (b) (c)

FIGURE 4.4: Narrow-baseline stereo images of a walking person: (a) top; (b) bottom; (c) chamfer volume and body model hypothesis.

self-occluded by crossed circles. The chamfer volume is used for tracking in Section 4.4.1.

4.3.2 Monocular

As a first step towards tracking from standard single-camera observations, this section presents work on a simple monocular silhouette-based objective function (further work using $\mathcal{WSL}$ tracks is presented in Chapter 6). The silhouette-based SSD function described in Section 3.5.1 is extended to enforce a match between the areas of hypothesis foreground (F_s) and observation foreground (F_z). This strategy is illustrated using simple shapes in Fig. 4.6: the aim is to minimise the difference in size between the foreground region produced by the hypothesised triangle and that produced by the observed triangle. A measure of the disparity between these values, W, is calculated as

$$W = \left(1 - \text{abs}\left(\frac{F_z - F_s}{\max(F_z, F_s)}\right)\right). \tag{4.8}$$

The silhouette-based SSD score Σ^s is then re-weighted by W as follows

$$\Sigma^s_W = \frac{1}{W^\gamma} \times \Sigma^s, \tag{4.9}$$

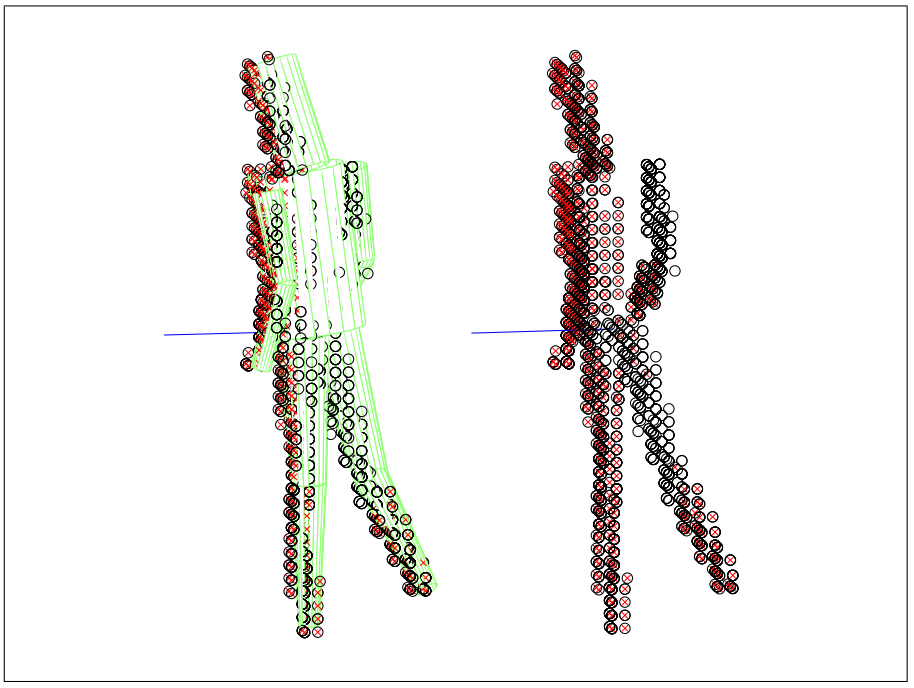


FIGURE 4.5: Body model from Fig. 4.4(c) rotated to show extracted sample coordinates (red) and effects of intra- and inter-cone occlusion (empty circles).

where the requirement for observation and hypothesis silhouette sizes to match may be enforced by varying the exponent γ. Strict agreement can be enforced by setting γ high, while setting $\gamma = 0$ is equivalent to using the original APF objective function Σ^s ($\Sigma^s_W = \Sigma^s$ in Eq. 4.9). This objective function is used for tracking in Section 4.4.2.

4.3.3 Wide-Baseline Stereo

In the multiocular calculation of particle weightings described in Section 3.5.1 and the monocular extension described above, there is no consideration given to foreground in image evidence which is left unaccounted for by a pose hypothesis. This becomes problematic where camera numbers are reduced and, in the absence of simultaneous observations from many different angles, the body model is free to take up compact but incorrect poses, or to move directly away from the camera simply to subsume itself in observation foreground. For example, see the observation foreground mask for a *box* pose in the left hand side of Fig. 4.8(b),

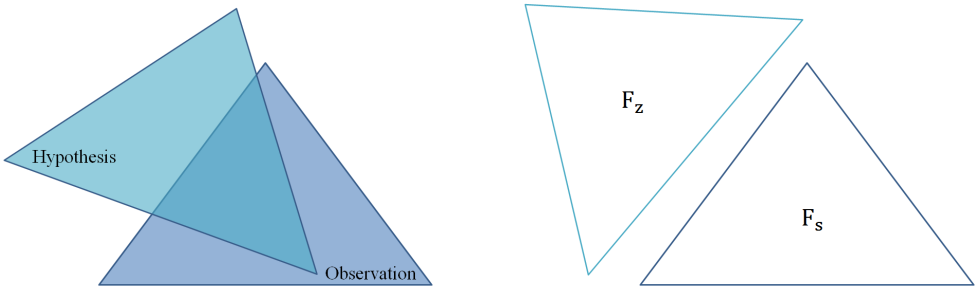


FIGURE 4.6: Diagram of observation and hypothesis foreground areas. Aim is to have F_s and F_z matching in size.

where an erroneous *low guard* pose hypothesis is largely subsumed by observation foreground and therefore scores well in terms of Σ^s.

Edge features can be useful in mitigating this problem as poses that explain the subject's outline score well in terms of Σ^e (see Eq. 3.40) [DBR00]. When tracking from only two cameras, however, edge cues have proven unable to prevent hypotheses moving directly away from sensors, see also standard APF results in Section 4.4.2 and [DLC08c]. This is likely to be due to the large number of internal edge responses that are recovered from casually dressed subjects such as those in the *HumanEva* database. These individuals wear loose-fitting clothing that creases and has detailing, see for example the high internal edge scores on the subject's torso in Fig. 3.17(d).

In this section a complementary silhouette-based measure is put forward as an alternative to the use of edge cues. Specifically, sampling of the observation foreground for comparison with the body model hypothesis is proposed. When combined with Σ^s, synthesised poses are required to satisfy two criteria: the body model should not lie over observation background *nor* leave observation foreground unaccounted for. This strategy is illustrated using simple shapes in Fig. 4.7: the aim is to minimise both the area of hypothesis that is left unexplained by observation *and* the area of observation that is unaccounted for by the hypothesis. The approach could be described as a sampling-based version of the symmetrical pixel-based objective function used by Sigal *et al.* [SBB10].

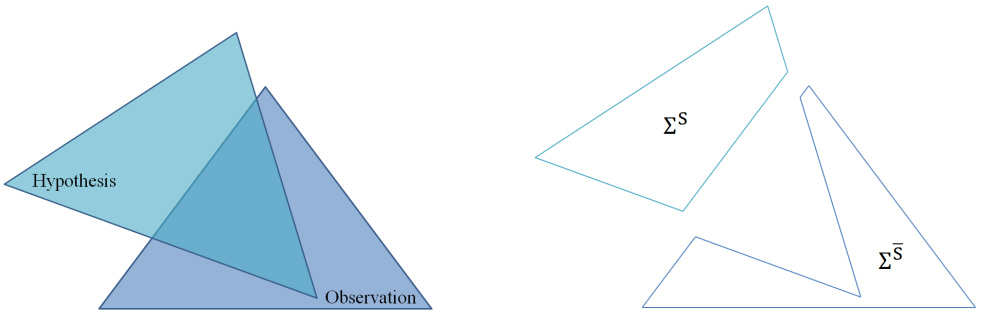


FIGURE 4.7: Diagram of symmetric sampling strategy. The aim is to minimise *both* the unexplained foreground regions, Σ^s and $\Sigma^{\bar{s}}$.

A measure of agreement $\Sigma^{\bar{s}}$ is computed between a binary *hypothesis foreground mask* $V^{\bar{s}}$, and a set of points $\{\nu\}$ drawn uniformly from the foreground region of the observation foreground mask V,

$$\Sigma^{\bar{s}} = \frac{1}{|\{\nu\}|} \sum_{\nu} (1 - V^{\bar{s}}(\nu))^2. \tag{4.10}$$

The resulting set of samples are shown in the right hand side of Fig. 4.8(b). The measure $\Sigma^{\bar{s}}$ is combined with the standard silhouette comparison Σ^s by substituting $\Sigma^{\bar{s}}$ for Σ^e in Eq. 3.41. A quantitative evaluation of this approach is presented below and it is used for tracking in Section 4.4.3.

The usefulness of this measure is demonstrated by considering the consequences of inducing known, artificial errors in the pose derived from a *HumanEva-I box* sequence. As shown in Fig. 4.8(b)-4.8(e), while the 500 frames of fragment run, the pose extracted stays motionless, and is compared on the one hand with the objective function scores extracted from the images, and on the other with the true pose obtained from motion capture (see also Section 3.3.1.1). This arrangement illustrates the relationship between the image-based SSD terms and true pose inaccuracies for a wide range of desired poses; during tracking, a wide range of possible poses will be tested against a single frame.

To account more accurately for the observation foreground masks cast by subjects, the binary hypothesis foreground mask is created from a set of truncated "clothes" cylinders with the subjects' limb widths scaled by a factor of 1.0-1.5.

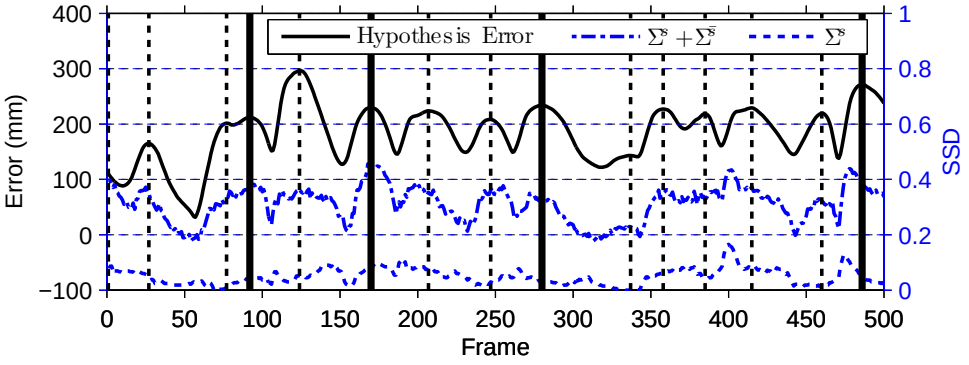
The hypotheses masks in Figs. 4.8(b)-4.8(e) use a scaling of 1.5 for each limb, but in practice these values are set manually based on a subject's clothing, e.g. 1.5 for a trousered lower leg, 1.0 for an unclothed head or forearm.

As can be seen from Fig. 4.8(a), both objective function scores relate quite closely to the true pose difference. However, the combined measure ($\Sigma^s + \Sigma^{\bar{s}}$) is in fact significantly more strongly associated with the true pose difference, as assessed by a Spearman rank correlation analysis. This shows that while the correlation between the original measure and the pose difference has $r = 0.267$, the new measure has $r = 0.677$. Due to the large number of frames, $d.f. = 498$, and so both of these values are significant far beyond $P = 0.05$. Similarly, the probability of the difference between these correlations being generated by chance is too small to be calculated.

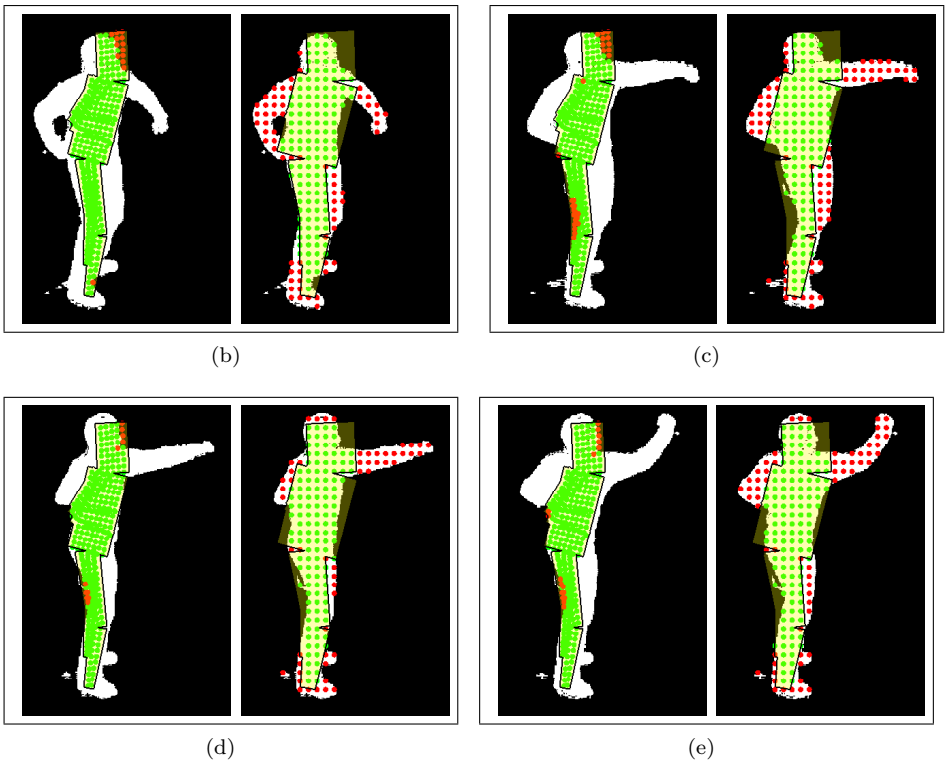
4.4 Experiments

This section contains known activity tracking results for monocular, narrow-baseline stereo, and wide-baseline stereo observations of *walk* and *jog* activities using HMM-APF combined with the objective functions defined in Section 4.3. Further details are given in the following subsections, but this introductory section covers a number of themes that are common to all experiments.

The body model of Bălan *et al.* [BSB05] is used in each experiment; a kinematic tree composed from a set of 10 truncated cones (see also Section 3.3.1.1). The adoption of this body model allows for the straightforward application of new tracking algorithms to the *HumanEva-I* and *HumanEva-II* datasets described in [SB06a, SBB10] and their predecessor described in [BSB05]. These datasets consist of a number of synchronised video sequences providing views of different human activities performed by a number of different subjects. Processing these sequences offers two important advantages over other methods of evaluation, (i) a synchronised record of MoCap ground truth permits the quantitative evaluation of markerless tracking techniques and (ii) an additional partition of MoCap data



(a) 3D absolute error between pose hypothesis and MoCap ground truth at each frame (mm).



(b) (c)

(d) (e)

FIGURE 4.8: Silhouette features for a fixed *low guard* pose hypothesis with hands held against the torso during a 500 frame *box* sequence. (a) 3D absolute error scores and corresponding SSD scores. Dashed vertical lines denote punches, with the 4 bold vertical lines corresponding to the image pairs (b)-(e). These show the sampling strategy for Σ^s (left) and $\Sigma^{\bar{s}}$ (right), with non matching samples plotted in red.

is provided that features the same subjects performing the same activities (at different times) and that is intended for training.

The training partition allows for cones in the body model to be accurately resized based on the measurements of individual subjects. It also permits the extraction of series of body model configurations that relate to a particular subject's pose

during the performance of training activities. For a particular activity, these configurations are given by series of global position vectors $\Omega = \{\underline{\omega}_1, ..., \underline{\omega}_M\}$ and pose vectors $Y = \{\underline{y}_1, ..., \underline{y}_M\}$ giving the relative joint rotations between limbs in terms of Euler angles. The position vectors comprise $D_\omega = 6$ parameters, three rotational and three translational and the pose vectors comprise $D_y = 36$ Euler angles. This pose representation features some redundancy; each joint is permitted three degrees of freedom, but many in fact require less. This is reflected by a negligible or zero variation across the training data (see also the lowest values in Fig. 3.13), meaning that particle positions do not vary in these dimensions[1].

Training data can be used for the learning of pose and dynamical models necessary for performing HMM-APF. Based on the investigation in Section 3.3.2 a latent pose space dimensionality $D_x = 4$ was chosen for all experiments, resulting in a corresponding set of latent variables $X = \{\underline{x}_1, ..., \underline{x}_M\}$ related to the original pose vectors Y through a linear mapping. HMMs of the form $\lambda = \{S, \boldsymbol{A}, \underline{a}, p_i(\underline{x})\}$ were estimated from latent variables using the steps described in Section 3.4.2. The Baum-Welch algorithm (see also Appendix C) guarantees only to find a local optimum and the HMM is reestimated from a new initialisation before each individual tracking experiment. Once training is complete a time-reversed transition matrix $\hat{\boldsymbol{A}}$ can optionally be calculated using Eq. 3.38, where the invariant distribution $\underline{\psi}$ is estimated by making 10^3 transitions via the original transition matrix, $\boldsymbol{A}$. Finite differencing of training data as described in Section 3.4.1 was also used to estimate the covariance matrices, $\boldsymbol{P}_r^\omega$, $\boldsymbol{P}_r^y$ and $\boldsymbol{P}_r^x$, used for dispersion at each layer. Note that these variables additionally facilitate tracking by standard APF or by SIR (see also Section 3.2.1 and Section 3.2.2) in either the latent or ambient pose spaces; these approaches are both adopted as baselines for comparison. The "default" APF parameters of 5 annealing layers and a constant survival rate of $\alpha_R = ... = \alpha_0 = 0.5$ were adopted from the literature, see Section 3.2.2 for a discussion of the implications of varying these values.

[1]For the dataset presented in [BSB05] a number of pose parameter variances are explicitly set to zero, effectively giving $D_y = 25$ but this does require some "ad hoc" manual adjustments to the remaining pose parameters.

4.4.1 Narrow-Baseline Stereo Tracking

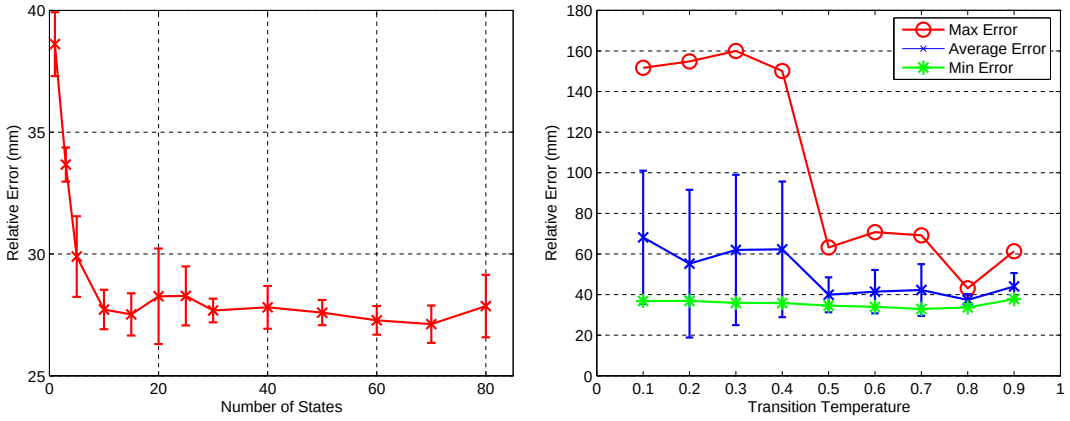
In this section the HMM-APF algorithm is applied to range data using the chamfer volume objective function defined in Section 4.3.1. First, a simulation is undertaken to investigate the effect on tracking of: (i) varying the number of HMM states and (ii) inflating dynamics via the transition temperature. Second, tracking is attempted using real stereo camera range data. Training data and test data are both taken from the dataset in [BSB05].

4.4.1.1 Simulation

To investigate the effectiveness of parameters chosen for the training and tracking processes, a series of simulation experiments were conducted using synthetic *walk* trials. The body model was "animated" using the 30fps ground truth test data from [BSB05]. The translation parameters in each position vector were set to zero to produce a pose recovery problem[2]. Synthetic range data relative to a fixed observation point was sampled from the visible surfaces of the cones and used to create a set of "idealised" chamfer volumes from which tracking was attempted using the SSD measure Σ^v. The scenario is one of a known subject performing known activity.

Expected error results (see also Section 3.5.4.1) for 40 particle HMM-APF using a range of state numbers to build the HMM are shown in Fig. 4.9(a). In Fig. 4.9(b) the number of states is held constant (at 15) and the effect of increasing the transition temperature on tracking accuracy is investigated. Each point plotted represents an average score from 10 separate tracks of the test sequence, with 4.9(b) also showing the best and worst tracking results. Fig. 4.9(a) used the first 75 frames of the sequence – which feature straight-line walking – to investigate the quality of pose recovery using different numbers of HMM states. Figs. 4.9(b) and 4.10 used a 150 frame sequence featuring a more challenging change of direction. Fig. 4.10 shows the performance of HMM-APF with 15 states and $\rho_T = 0.6$ versus

[2]Translation parameter variance was also set to zero in $\boldsymbol{P}_r^{\omega}$.



(a) (b)

FIGURE 4.9: Range data simulation results: (a) mean tracking error versus number of states; (b) tracking error versus transition temperature ρ_T.

latent SIR using $\boldsymbol{P}_r^x$ for propagation and an equivalent number of particles (200; that is, 40 particles multiplied by 5 annealing layers).

No significant improvement in performance was found using greater than 10 HMM states. Failures were observed when using only the HMM transition matrix $\boldsymbol{A}$ (low ρ_T, Fig. 4.9(b)) or the Gaussian random variable $\boldsymbol{P}_r^x$ (latent SIR, Fig. 4.10) as a dynamical model for tracking the longer sequence. However, inflating the learned dynamical model by increasing ρ_T to make more state transitions produced consistent reductions in the tracking error. This improvement was most pronounced above the value $\rho_T = 0.5$: ensuring that at least half the particle set is spread beyond the current state reducing the average error by around a third. To reflect this trend the transition temperature was fixed at $\rho_T = 0.6$ – just above the observed step-change in average error – for the comparison with SIR (Fig. 4.10) and other experiments in the remainder of this chapter.

Although choosing even higher transition temperatures may appear to bring futher benefits – e.g. see results for $\rho_T = 0.8$ – this corresponds to very high magnitude dispersion and particles have been observed moving right round the activity manifold in a single time step at these temperatures. The reduced error score at $\rho_T = 0.8$ is likely to be due to the reinforcement of these particles at or

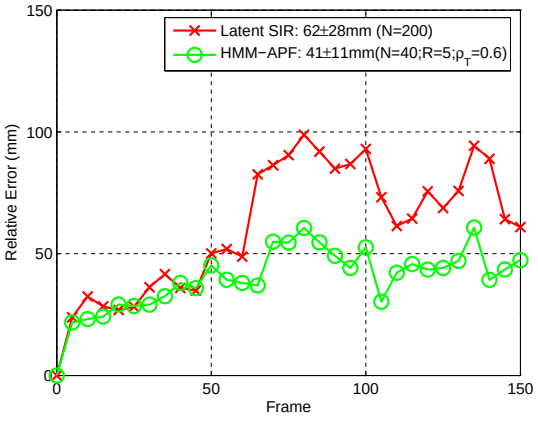


FIGURE 4.10: SIR versus HMM-APF for range data.

near the current pose estimate. This result is interesting but such high temperatures are inappropriate for all but periodic activities with cyclic manifolds[3]. The proposed HMM-APF method maintains tracking throughout each of the 10 tests with a more conservative transition temperature of $\rho_T = 0.6$. Fig. 4.10 shows frame-by-frame errors for the longer sequence, the increase in SIR tracking error at around frame 60 is due to tracking failures as the subject turns towards the camera. Attempting to propagate the full posterior does not lead to improved performance, reflecting the findings of [BSB05] on the same data.

4.4.1.2 Range Data

A 5 second sequence of an unknown walking subject was recorded at 30fps using a Videre MDCS2-VAR stereo camera [Vid]. The camera was held by hand and continually adjusted to ensure the subject remained fully in shot[4]. Range data was calculated using the commercially available Small Vision System software [Vid, Kon97] and discretised onto a 3D grid with a resolution of 4cm in each dimension. It was then smoothed with a $7 \times 7 \times 7$ Gaussian kernel with $\sigma = 4$ to produce a series of chamfer volumes. The body model of the subject from [BSB05] was hand-initialised at the first frame and HMM-APF tracking attempted using

[3] In Section 4.4.3 an alternative approach to reinforcement is investigated by using a time-reversed transition matrix to enable particles to move *backwards* along the activity manifold away from the current pose estimate, as well as forwards.

[4] Translation parameter variance was halved in the estimation of $\boldsymbol{P}_r^{\omega}$ to represent this fact.

100 particles, 15 states, and $\rho_T = 0.6$. The scenario is one of an unknown subject performing known activity.

Results are shown in Fig. 4.11 and demonstrate qualitatively satisfactory tracking of an unknown walking subject from stereo range data. As anticipated, the depth cue is noisy and ambiguous (see also discussion in Section 3.5.2.1) but although some mis-tracking is seen – for example the right leg in the top right image of Fig. 4.11 is attracted by the nearby background clutter – a reasonable pose estimate is always recovered within a few frames. Although the quality of pose recovery is similar to that presented in other studies (e.g. [UFF06b]), perhaps the most impressive aspect of the result is maintenance of a good global translation and rotation estimate for the subject as they move through the room.

One concern when tracking with strong priors on dynamics and pose is that observations are in fact redundant and the particle set will move through the correct state space trajectory regardless. A particularly interesting example is given by Sidenbladh *et al.* [SBF00] (see in particular their Fig. 7) where the particle set is found to reconstruct the poses of a person walking in a straight line with surprising accuracy despite the *complete absence* of observations (all particles are assigned an equal weighting). Tracking only breaks down when the subject turns to walk in a new direction. Having the subject in Fig. 4.11 perform a relatively sharp turn ensures that it is impossible for the activity prior alone to maintain tracking: in addition to the pose parameters, the 6 position parameters of the root must also be estimated via $\boldsymbol{P}_r^{\omega}$. This represents a challenging difference versus the more common scenario of a subject walking in a straight line across the field of view of a stationary camera, e.g. see the full body stereo results presented in the study by Urtasun and Fua [UF04].

4.4.2 Monocular Tracking

The HMM-APF algorithm was used to track the first 150 frames of the *walk* sequence tested by Bălan *et al.* [BSB05] using 40 particles and the weighted

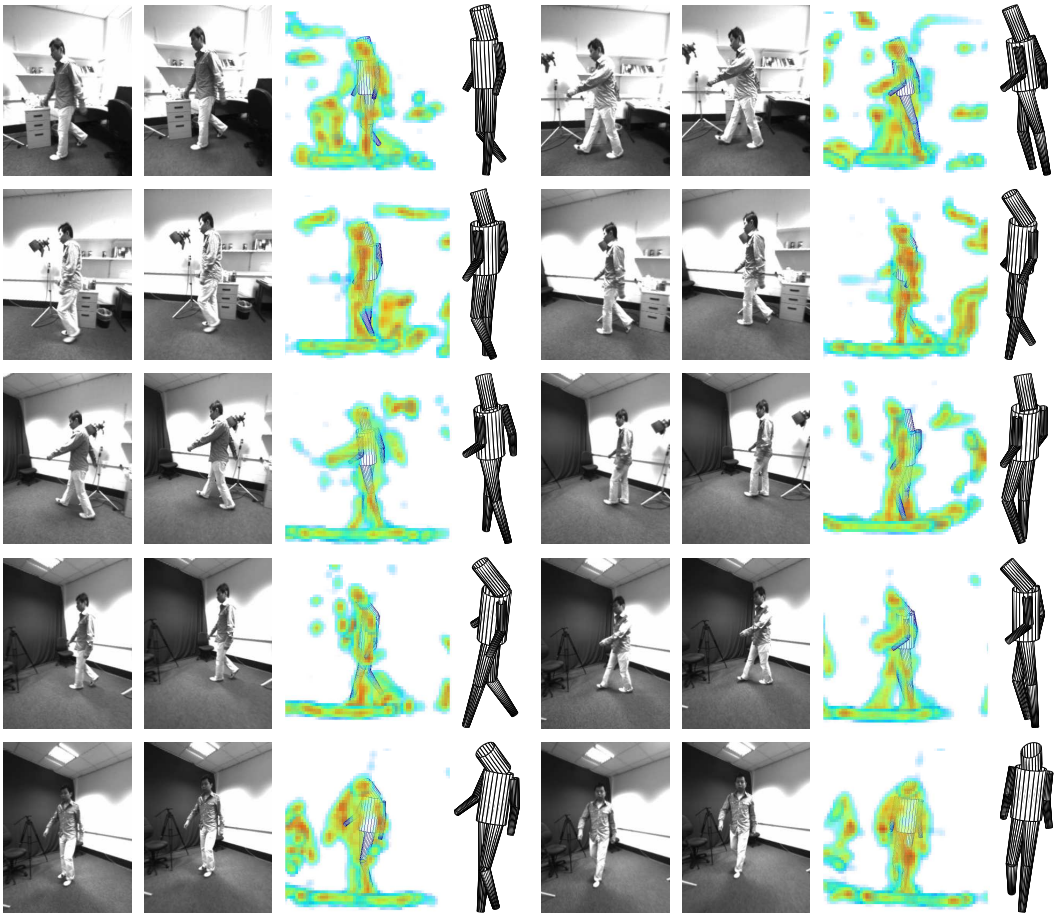


FIGURE 4.11: Narrow-baseline stereo tracking results with $N = 100$.

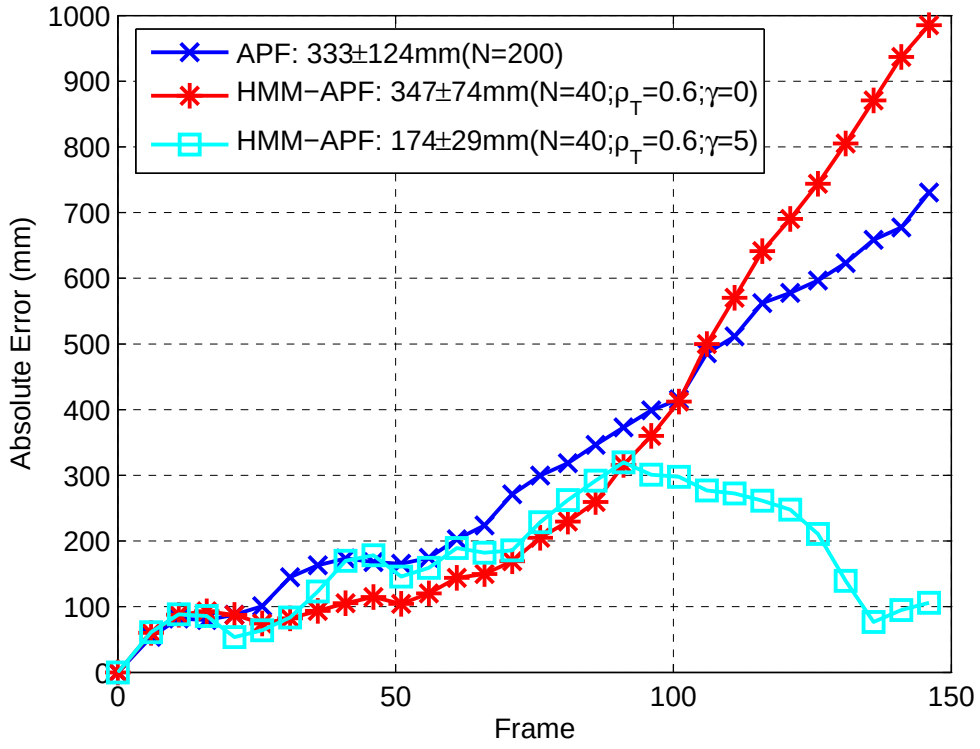
SSD score Σ^s_W defined in Section 4.3.2. The scenario is one of a known subject performing known activity. Results are shown in Fig. 4.12 with standard APF using five times as many particles (200 per layer), $\boldsymbol{P}^{\omega}_r$ and $\boldsymbol{P}^{y}_r$ for the propagation of particles, and the original edge-plus-silhouette weighting function described in Section 3.5.1 included for comparison. For each setting, tracking of the sequence was attempted 10 times and the average expected error at each frame across the 10 runs is plotted. The average expected error was also calculated across each entire run and the mean and standard deviation of error across the 10 runs is shown in the legends. For the *relative* error calculation in Fig. 4.12(b), the global coordinates of the virtual and MoCap pelvis markers were set equal before computing the average marker error. Typical qualitative results for standard APF and HMM-APF are shown in Fig. 4.13 and Fig. 4.14, respectively.

In the case of monocular tracking, maintaining an accurate estimate of the subject's global coordinates is very challenging. The body model tends to "sit back", ensuring it is enveloped by image evidence and scoring well in terms of the silhouette-based objective function. This can be seen in Fig. 4.12(a) for standard APF and for HMM-APF with $\gamma = 0$ where the high absolute errors are due, overwhelmingly, to error in estimating the subject's global position within the room. Enforcing agreement between the silhouette sizes by setting $\gamma = 5$ causes the body model to move with the subject as they start to walk towards the camera (around frame 90 in Fig. 4.12(a), top right image in Fig. 4.14). Error arising from inaccuracy in the global coordinates is difficult to eliminate entirely, with absolute error still reaching around 300mm for $\gamma = 5$, but (with dynamics inflated by $\rho_T = 0.6$) correct pose recovery is now observed across all 10 runs, giving a mean expected relative error of 55 ± 5mm, see Fig. 4.12(b).

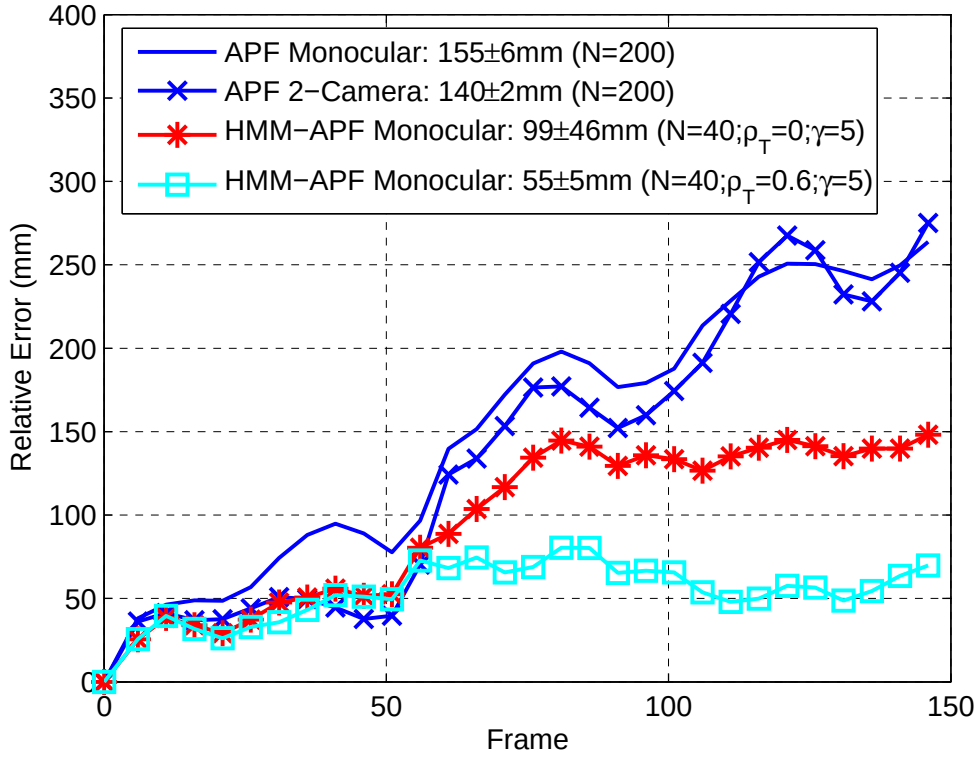
These findings are interesting as they show that, where activity is known, HMM-APF has the capacity to outperform standard APF quite considerably on monocular data. Results have been presented in such a way as to facilitate direct comparison with the extensive quantitative evaluation of standard APF presented in [BSB05]. Ultimately, however, the approach presented relies too heavily on good quality segmentation. Even though data is captured in a controlled environment, demanding agreement between F_s and F_z does not allow the subject's *global* location to be accurately inferred; although relative error is consistently low, absolute error features a considerable peak. Using range data (see also Section 4.4.1) may overcome this problem but this cannot be demonstrated *quantitatively* without a synchronised record of MoCap ground truth. Further discussion is given in Section 4.5 and an alternative approach to monocular tracking in Chapter 6.

4.4.3 Wide-Baseline Stereo Tracking

HMM-APF clearly shows potential for the recovery of known activity from limited observation data. However, the comparison with a high-dimensional approach such as standard APF – intended for the recovery of freeform motions –

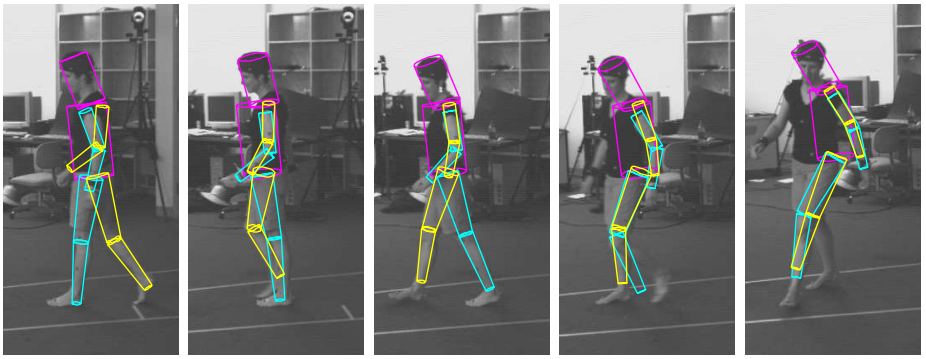


(a) Absolute error.

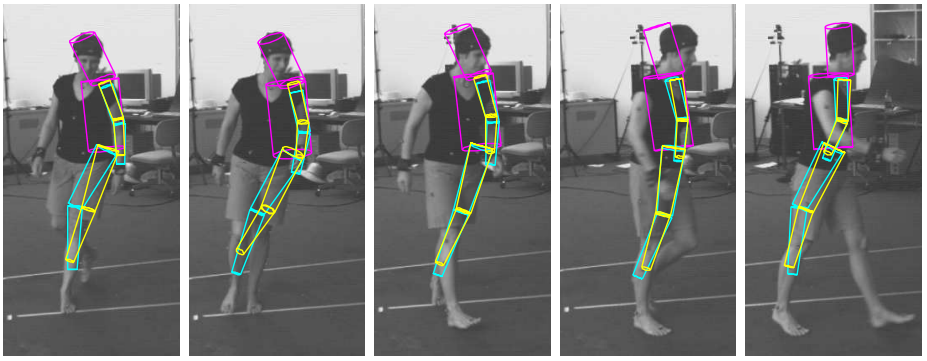


(b) Relative error.

FIGURE 4.12: Monocular tracking results for HMM-APF versus standard APF across 10 runs: (a) average expected absolute error; (b) average expected relative error.

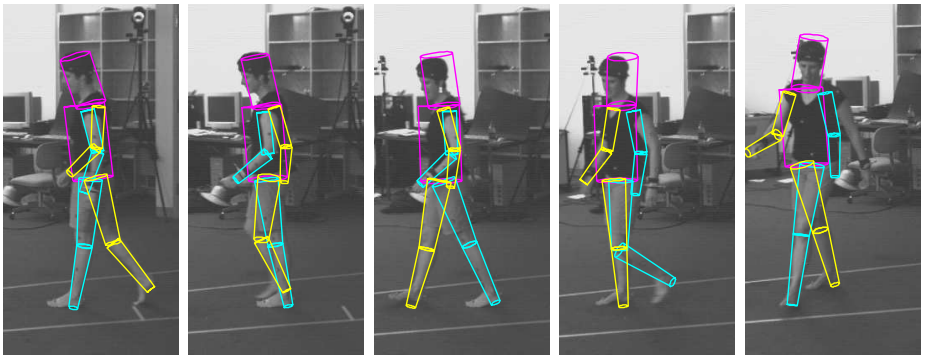


(a) $t = 15$ (b) $t = 30$ (c) $t = 45$ (d) $t = 60$ (e) $t = 75$

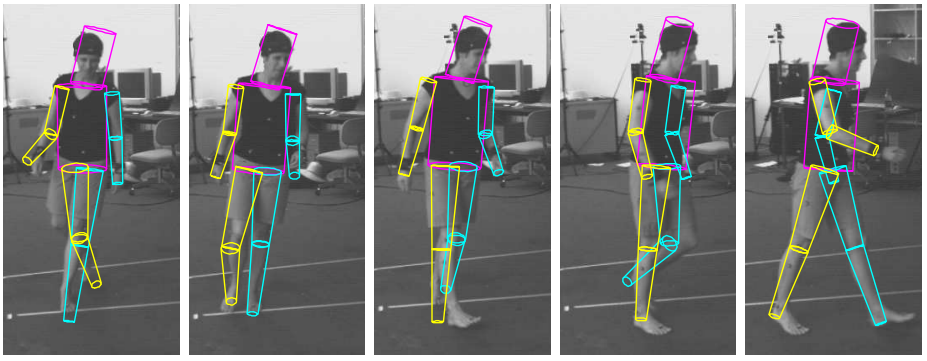


(f) $t = 90$ (g) $t = 105$ (h) $t = 120$ (i) $t = 135$ (j) $t = 150$

FIGURE 4.13: Monocular tracking using standard APF with $N = 200$.



(a) $t = 15$ (b) $t = 30$ (c) $t = 45$ (d) $t = 60$ (e) $t = 75$



(f) $t = 90$ (g) $t = 105$ (h) $t = 120$ (i) $t = 135$ (j) $t = 150$

FIGURE 4.14: Monocular tracking using HMM-APF with $N = 40$.

1. The position of the (n)th particle in the rth layer is given by the position and latent parameters, $\underline{s}_{t,r}^{(n)} = [\underline{\omega}_{t,r}, \underline{x}_{t,r}]$.

2. The particle's position parameters $\underline{\omega}_{t,r}$ are updated by the addition of the Gaussian random variable $\underline{n}_r^{\omega}$,

$$\underline{\omega}'_{t,r} = \underline{\omega}_{t,r} + \sum_1^{T_r} \underline{n}_r^{\omega}. \tag{4.11}$$

3. The particle's latent parameters $\underline{x}_{t,r}$ are updated T_r times by the addition of the Gaussian random variable $\underline{n}_r^{x}$,

$$\underline{x}'_{t,r} = \underline{x}_{t,r} + \sum_1^{T_r} \underline{n}_r^{x}. \tag{4.12}$$

4. The new estimates are then used to create a particle in a new set

$$[\underline{\omega}'_{t,r}, \underline{x}'_{t,r}] = \begin{cases} \underline{s}_{t,r-1}^{(n)} & \text{if } r > 0; \\ \underline{s}_{t+1,R}^{(n)} & \text{if } r = 0. \end{cases} \tag{4.13}$$

FIGURE 4.15: Dispersion of a single particle for known activity: latent APF.

is perhaps inappropriate. Here a more competitive baseline is proposed: latent APF. Latent APF uses the Gaussian random variable $\underline{n}_r^x$ as a dynamical model in the same latent pose space as HMM-APF. The latent APF particle dispersion process is detailed in Fig. 4.15 and a visualisation of its application to a *walk* observation is given in Fig. 4.2(b). Notice that in contrast to HMM-APF (depicted in Fig. 4.2(a)) particles *are* now free to move anywhere in latent space, independent of the path traced out by training data. Particles are dispersed not by an HMM, but by an aggregated dynamical model found by finite differencing latent data.

One problem for the use of dynamic T_0 is that there is no clear analogue for use in other dispersion methods. This has two implications: (i) that direct comparisons between dynamical models are difficult to draw; (ii) that the integration of multiple dynamical models is difficult (this is the objective of Chapter 5). To facilitate comparison between HMM-APF and latent APF, T_0 is held constant throughout tracking, and the performance of both schemes on much longer *HumanEva-I* activity sequences is investigated for a range of different choices of T_0.

Each approach was tested on a *walk* and *jog* sequence from the *HumanEva-I*

Validation partition using two wide-baseline cameras and the complementary SSD scores Σ^s and $\Sigma^{\bar{s}}$ proposed in Section 4.3.3. The *walk* and *jog* activities are of particular interest as they are the two known activities in the *HumanEva-II Combo* sequences studied in Chapter 5. The *walk* sequence of subject S1 and the *jog* sequence of subject S3 were chosen as they are the longest in the *HumanEva-I* dataset. Pose and position vectors were extracted from S1 and S3's portions of the *HumanEva-I Training* partition, finite differencing used to estimate the Gaussian random variables $\underline{n}_r^x$ and $\underline{n}_r^\omega$, and PCA applied to recover latent pose spaces and associated HMMs. The scenario is one of a known subject performing known activity.

For every value of T_0, the whole of each sequence was tracked ten times using the two cameras C1 and C2. The lowest number of particles tested by Sigal *et al.* [SBB10] were used; 50 particles over 5 annealing layers. In Fig. 4.16 average 3D absolute expected error results are presented for each sequence using both HMM-APF (with and without the time reversed matrix, $\hat{\boldsymbol{A}}$) and latent APF. Error bars show the standard deviation in average error across the ten runs at each T_0 value.

Average latent APF errors decrease with the number of time steps up to around $T_0 = 4$, mirroring the benefit of inflating dynamics seen in Figs. 4.9(b) and 4.12(b). However tracking failures still take place, as evidenced by the large standard deviations in error. For HMM-APF, robust tracking without failures is achieved even at low T_0 values, suggesting that a tightly constrained dynamical model in addition to a 2-camera symmetric objective function considerably reduces ambiguity.

Without the use of the time reversed transition matrix $\hat{\boldsymbol{A}}$, HMM-APF performance slowly degrades with increased T_0 when processing the faster *jog* activity. In contrast, HMM-APF with $\hat{\boldsymbol{A}}$ correctly tracks both of the sequences across the range of T_0 values, producing low average errors and low standard deviations in error across each batch of ten runs. The difference in performance between

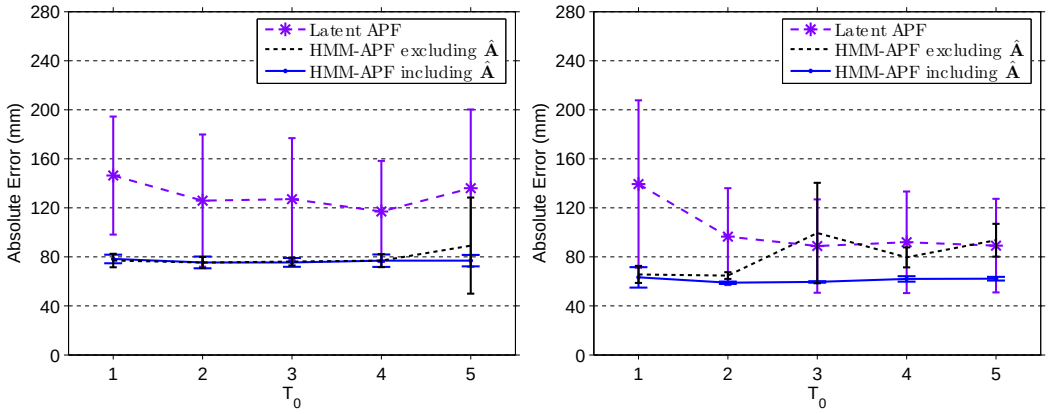


FIGURE 4.16: 3D absolute error results for HMM-APF versus latent APF: (left) *walk*; (right) *jog*. HMM-APF with time reversed transition matrix $\hat{\boldsymbol{A}}$ recovers low error pose estimates across the range of T_0. Latent APF produces higher average errors, with optimal performance occurring at around $T_0 = 4$.

HMM-APF and latent APF suggests that, in addition to the latent pose space, the choice of dynamical model is important for producing robust tracking.

4.5 Discussion and Conclusions

HMM-APF appears to support robust tracking from a number of different sources of observation data. Inflation of dynamics proved particularly important for ambiguous observation data as evidenced by tracking failures for low ρ_0 in Fig. 4.9(b) and Fig. 4.12(b). This appears to be less important when using two cameras and the symmetric observation function, see Fig. 4.16 but inflation is strongly advocated regardless; particles should be spread as widely as possible to facilitate recovery from errors if they do occur. Use of the reverse transition matrix $\hat{\boldsymbol{A}}$ ensures there is no degradation in performance for HMM-APF up to and including $T_0 = 5$ for both *walk* and *jog* activities. Inflation of dynamics also ensures that optimal performance is obtained from the latent APF baseline, occurring at around $T_0 = 4$. Tracking errors continue to occur, however, confirming the importance of a well-constrained dynamical model – the HMM – in addition to a latent pose space.

The results using range data are interesting, but the absence of a synchronised record of ground truth limits their usefulness. Synchronised narrow-baseline stereo and MoCap data capture – similar to that presented in *HumanEva* – is an objective of future work (see also Section 7.1.2.1). Qualitative results suggest that a relatively noisy estimate of 2.5D surface data is a sufficient cue for human motion tracking where the activity class is known. This is the only experiment where the subject is unknown – that is, no training data is available for them and the dimensions of the body model were not tailored to their physical appearance. Although it is the use of an activity model that facilitates tracking, the constraints this places on the state space are visible in stylistic intra-activity differences between the training and test subject, e.g. in the bend of the arms as they swing forward in Fig. 4.11. This result indicates the specificity of HMM-APF. Addressing this limitation is the primary objective of Chapter 5.

The use of chamfer volumes should offer an advantage over monocular approaches in terms of absolute tracking error as the true 3D position of the subject relative to the sensor is estimated. The method can be easily applied to data captured using other range sensors, such as time-of-flight cameras and should perform well outdoors and in other more natural scenes where backgrounds are changing and lighting and shadows are not controlled. This is in contrast to the use of silhouettes. To achieve monocular tracking considerable emphasis had to be given to agreement between observation and hypothesis foreground area. When this agreement is not enforced tracking breaks down, e.g. compare results for $\gamma = 5$ and $\gamma = 0$ in Fig. 4.12(a). Performance will inevitably suffer as the quality of extracted silhouettes degrades.

In longer sequences such as the *HumanEva-I* videos tested in Section 4.4.3, the quality of silhouettes varies quite considerably. Accuracy of segmentation changes as the subject moves through the capture area, occluding different regions of the background model and casting shadows over their surroundings. For this reason use of the more principled symmetric objective function proposed in Section 4.3.3 is pursued for the more challenging sequences addressed in Chapter 5.

The monocular case is revisited in Chapter 6 by using a tracker that models the *subject's* appearance rather than that of their surroundings.

Chapter 5

Known and Unknown Activity

In this chapter the low-dimensional generative tracking approach defined in Chapter 4 is integrated with a simultaneous high-dimensional generative tracking approach. The associated inference tasks have quite different levels of difficulty and are assigned differently sized particle quotas to reflect this fact. A "particle stacking" method is described to ensure fair but efficient exploration of each space. A method for drawing a variable number of samples at subsequent annealing layers based on the emerging picture of activity model membership is proposed. The resulting algorithm is demonstrated tracking known and unknown human motions in the HumanEva-II Combo sequences using a variable number of particles and fewer than four cameras.

5.1 Introduction and Related Work

Existing generative tracking approaches can be broadly divided between two groups: those that attempt to solve an estimation problem in the body model's ambient pose space (e.g., [DBR00, CGH05, BEB08]), and those that attempt it in a low-dimensional embedding of the ambient pose space learned from training data (e.g., [SBF00, LYST06, TLS05]). High-dimensional applications of particle-based estimation – including particle filtering [AMGC02], annealed particle filtering [DBR00], adaptive diffusion [DR05], and partitioned sampling [MI00] – have required large particle numbers and a minimum of four cameras [SBB10, BSB05, BEB08]. While such approaches are computationally demanding, requiring a

large number of objective function evaluations against each camera observation, they have been successful in recovering freeform motions without restriction on activity class.

An alternative to searching the ambient pose space is to learn a low-dimensional latent pose space from training data. This was the approach undertaken in Chapter 4. The generative estimation task has been attempted in linear PCA spaces recovered from MoCap data using both particle filtering [SBF00] and deterministic optimisation [UFF06b]. Similar techniques have also been applied in non-linear latent pose spaces recovered using "piecewise linear" PCA [BS00], locally linear coordination (LLC) [LYST06], the Laplacian eigenmaps latent variable model (LELVM) [LPS07], and the Gaussian process latent variable model (GP-LVM) [TLS05]. In contrast to high-dimensional approaches, the use of a latent pose space has allowed for robust tracking from fewer cameras, at reduced computational expense. The main drawback of these approaches is their inability to generalise (see also Section 3.3.6). Although some pose spaces have been shown to account for intra-activity variations in style [UFF06a], none are able to account for new activities not featured in the training set.

This chapter attempts to combine the competing benefits – flexibility and efficiency – of these two generative tracking scenarios within a single approach. The approach presented is partly inspired by the use of mixed-state particle filters to track with multiple dynamical models [IB98c], but additionally adapts the number of particles needed. Variable particle numbers have previously been adopted to minimise an error estimate between the true posterior and the sample-based approximation [Fox01]. However, here their numbers are varied based on the difficulty of the estimation task given a particular activity model. The approach is similar in style to the variable-mass particle filter for vehicle tracking [KM08], where variable particle numbers may be allocated to competing dynamical models based on arbitrary criteria.

The main contributions of this chapter are as follows:

- Definition of two further activity models to complement HMM-APF (Chapter 4). Known activity *transitions* are modelled by permitting particles to flow between activity manifolds in a joint-activity latent pose space (Section 5.3.3). Unknown activities are modelled using Gaussian noise to propagate particles in the high-dimensional ambient pose space (Section 5.3.1).

- Proposal of an approach to combine a number of different activity models within the APF framework (Section 5.4). A *particle stacking* approach allows for the simultaneous consideration of multiple activity models described by different dynamical models spanning pose spaces of different dimensionality.

- The resulting estimation tasks are quite different in terms of difficulty, and they are assigned differently sized particle quotas to reflect this. A variable number of particles are resampled at each annealing layer based on the emerging picture of activity class membership. This allows for the recovery of known activities using only a small number of particles in a latent pose space, and unknown activities using a large number of particles in the full pose space (Section 5.4.2).

- Evaluation of the proposed scheme on *HumanEva-II* data. Robust tracking *and classification* is demonstrated on the *HumanEva-II Combo* sequences, which contain known activities, known activity transitions and unknown activity (Section 5.5). The proposed approach allows for a reduction of over 50% in the number of objective function evaluations required during known activity tracking.

The resulting algorithm, which is termed multiple activity model annealed particle filtering (MAM-APF), is an attempt to combine the best aspects of both generative approaches: faster recovery of known activity with few particles where possible, but the flexibility to work for longer with more particles to recover unknown activities where necessary.

5.2 Dimensionality Reduction

The focus of this chapter is to address the inflexibility of the work presented in Chapter 4 by combining latent space estimation for known activity with ambient space estimation of unknown activity. To this end PCA is chosen to perform dimension reduction. This is because particles must be free to flow between ambient and latent pose spaces during tracking, and the inexpensive bi-directional mapping offered by PCA is ideal for this purpose. For GP-LVMs learning itself is expensive and once complete calculation of the GP mapping between new points in the latent pose space and the ambient high-dimensional pose space has complexity quadratic in the number of training points. Further, the GP mapping is not bi-directional, and additional steps, such as the use of "back constraints" [LQC06], must be taken to enable mapping from new points in the ambient space to new points in the latent space. Work on generalising to novel poses using non-linear latent variable models learned from small amounts of training data is presented in Chapter 6.

5.3 Activity Model Definitions

In this section the various techniques described in Chapter 3 are combined to define three separate activity models. These are intended for use in particle dispersion during three different scenarios: (i) unknown activities, (ii) known activities, (iii) known activity transitions. Just as in Chapter 4, the inflation of dynamical models is undertaken to encourage recovery from errors [ST03a, Smi08]. In anticipation of this fact, each of the following subsections describes how the dynamical model may be used to produce $p(\underline{s}_{t-1+T_0}|\underline{s}_{t-1})$ where $T_0 \geq 1$, when creating a new particle set for the next frame with Eq. 3.14.

Although a dynamic inflation method has been investigated in Section 4.2.1.2, it has no analogue outside the HMM framework. As the intention here is the combine a number of different activity models, dynamics are inflated uniformly

across all activity models using a constant value for T_0 (see also Section 4.2.1.1). In line with the APF dispersion scaling in Eq. 3.8, the number of synthesised time steps is rescaled after each annealing layer using the survival rate α_r, to give

$$T_r = \lceil \alpha_R \times ... \times \alpha_r \times T_0 \rceil . \tag{5.1}$$

Note that setting $T_0 = 1$ causes $T_r = 1$ for all r, in which case no inflation is in effect and Section 5.3.1 describes standard APF [DBR00].

5.3.1 Unknown Activities

To track an unknown activity – that is, an activity for which no training data is available – the ambient 42D pose space must be explored. Although this search is expensive due to the high-dimensionality of the search space, it places no restriction on the activity class. This is the original aim of standard APF, and the techniques covered in Section 3.2.2. By dispersing particles in the ambient pose space using a Gaussian random variable it is, in theory, possible to recover any pose. Fig. 5.1 describes the dispersion of particles in the ambient pose space for unknown activity tracking.

Following Bălan *et al.* [BSB05] a check is used to find particles that describe poses where limbs intersect either with each other, or with the floor. Rather than simply discarding these hypotheses, however, resampling of the previous set continues until a complete set of "good" poses has been found. The intersection test looks for *any* intersection (regardless of its extent) between pairs of cones in the subject's body model. In practice many natural poses (including those in the *HumanEva-I* training set) were found to violate this intersection condition. This is because the rigid primitive shapes provide poor models of the deformable skin and muscle that surrounds bones in the human body. Training activities such as *jog*, where the arms are held close to the torso, regularly lead to slight intersections. To avoid the exclusion of these poses a set of truncated "bones"

1. The position of the (n)th particle in the rth layer is given by the position and pose parameters, $\underline{s}_{t,r}^{(n)} = [\underline{\omega}_{t,r}, \underline{y}_{t,r}]$.

2. The particle's position parameters $\underline{\omega}_{t,r}$ are updated T_r times by the addition of the Gaussian random variable $\underline{n}_r^{\omega}$,

$$\underline{\omega}'_{t,r} = \underline{\omega}_{t,r} + \sum_1^{T_r} \underline{n}_r^{\omega}. \tag{5.2}$$

3. Similarly, the particle's pose parameters $\underline{y}_{t,r}$ are updated T_r times by the addition of the Gaussian random variable $\underline{n}_r^{y}$

$$\underline{y}'_{t,r} = \underline{y}_{t,r} + \sum_1^{T_r} \underline{n}_r^{y}. \tag{5.3}$$

4. The new estimates are then used to create a particle in a new set

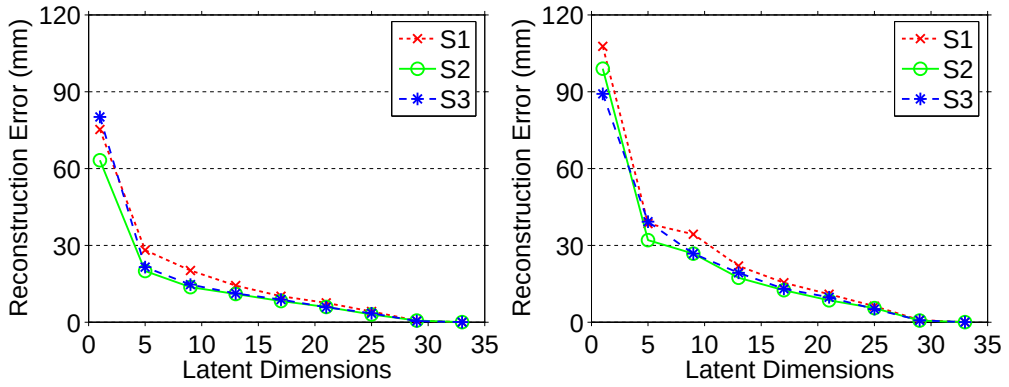
$$[\underline{\omega}'_{t,r}, \underline{y}'_{t,r}] = \begin{cases} \underline{s}_{t,r-1}^{(n)} & \text{if } r > 0; \\ \underline{s}_{t+1,R}^{(n)} & \text{if } r = 0. \end{cases} \tag{5.4}$$

FIGURE 5.1: Dispersion of a single particle for unknown activity tracking.

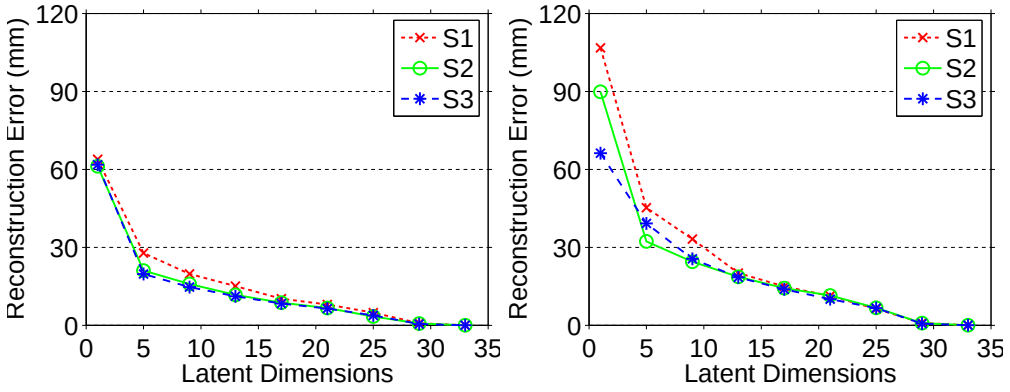
cylinders with the subjects' limb widths scaled by a factor of 0.8 were instead tested using the strict intersection conditions.

5.3.2 Multiple Known Activities

The HMM-APF approach described in Chapter 4 may be extended to model two activities in a *joint* latent pose space. Equal lengths of pose vector training data for each of two activities are concatenated before the application of PCA. The resulting latent variables are then divided equally and used to train two separate activity HMMs. The increase in reconstruction error due to modelling *walk* and *jog* activities in a joint latent pose space rather than individual latent pose spaces is small, see left hand side of Fig. 5.2 (and Fig. 3.7 for comparison). Particle dispersion takes place just as described in Section 4.2.1 and Fig. 4.1, but with every particle assigned to the single most likely parent state from *either* of the two activity HMMs.



(a) *Walk.*



(b) *Jog.*

FIGURE 5.2: Activity reconstruction errors for joint-activity latent pose spaces: (a) *walk*; (b) *jog*. Errors are shown for individual-subject spaces (left) and joint-subject spaces (right).

5.3.3 Known Activity Transitions

Where two (or more) segmented activities are modelled by the latent pose space, a transition activity model is introduced to permit particles to flow along *transition lines* between the most likely parent states in *each* of the activity HMMs. In the absence of any *HumanEva-I* training data, transition lines are useful in capturing the transition between *walk* and *jog* in the *HumanEva-II Combo* sequences. Fig. 5.3 describes the dispersion of particles along transition lines for the tracking of transitions between known activities. Example transition lines are depicted in Fig. 5.4, and the associated poses in Fig. 5.5.

1. The position of the (n)th particle in the rth layer is given by the position and latent parameters, $\underline{s}_{t,r}^{(n)} = [\underline{\omega}_{t,r}, \underline{x}_{t,r}]$.

2. The particle's position parameters $\underline{\omega}_{t,r}$ are updated T_r times by the addition of the Gaussian random variable $\underline{n}_r^{\omega}$,

$$\underline{\omega}'_{t,r} = \underline{\omega}_{t,r} + \sum_1^{T_r} \underline{n}_r^{\omega}. \tag{5.5}$$

3. The particle's latent parameters are allocated to the parent state most likely to have emitted them via $p_i(\underline{x})$. This may come from either activity HMM. They are then shifted to lie at the closest point on a line connecting the parent state's mean $\underline{\mu}_{i_1}$ with the mean of the particle's most likely parent state in the *other* activity HMM, $\underline{\mu}_{i_2}$. This line is referred to as the *transition line*.

4. The new estimate is then updated T_r times by dispersal along the transition line by a zero mean scalar Gaussian random variable $n_r^{\uparrow}$,

$$\underline{x}'_{t,r} = \underline{x}_{t,r} + \sum_1^{T_r} n_r^{\uparrow} \times \hat{\underline{u}} \tag{5.6}$$

where the unit vector $\hat{\underline{u}}$ is given by

$$\hat{\underline{u}} = (\underline{\mu}_{i_2} - \underline{\mu}_{i_1})/||(\underline{\mu}_{i_2} - \underline{\mu}_{i_1})||. \tag{5.7}$$

The variance of $n_r^{\uparrow}$ is chosen to be equal to the single largest element of the parent state's observation density covariance matrix at layer r,

$$n_r^{\uparrow} \sim N(0, ||\mathbf{\Sigma}_{i,r}||_{\max}). \tag{5.8}$$

5. The new estimates are then used to create a particle in a new set

$$[\underline{\omega}'_{t,r}, \underline{x}'_{t,r}] = \begin{cases} \underline{s}_{t,r-1}^{(n)} & \text{if } r > 0; \\ \underline{s}_{t+1,R}^{(n)} & \text{if } r = 0. \end{cases} \tag{5.9}$$

FIGURE 5.3: Dispersion of a single particle for known activity transitions.

5.4 Combining Activity Models (MAM-APF)

At every time step t, standard APF attempts to recover the body model pose that maximises the objective function [DBR00]. In contrast to standard particle filtering – where the posterior $p(\underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t)$ is propagated between time steps – the annealing process recovers a set of particles that are densely concentrated about a particular pose solution. To produce the initialising particle set for the

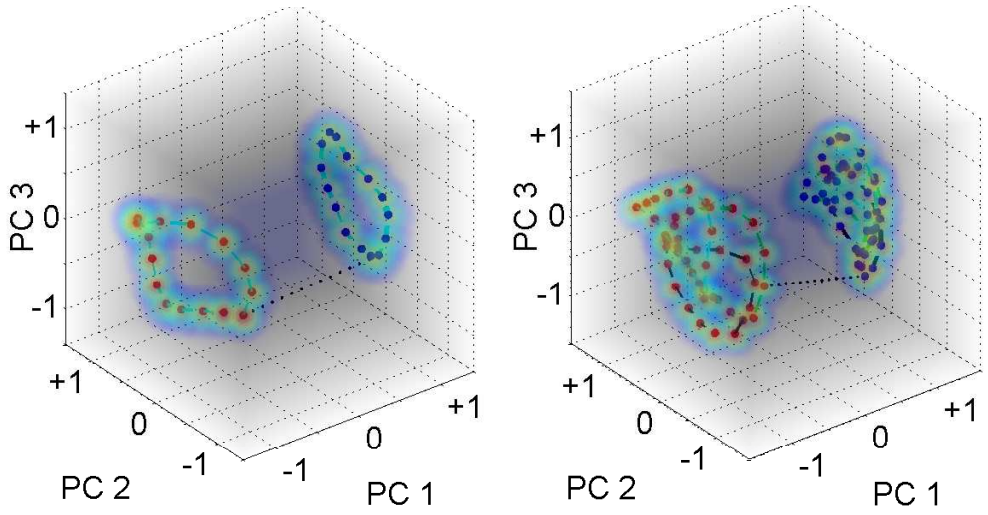


FIGURE 5.4: Joint-activity latent pose spaces: (left) individual-subject space; (right) joint-subject space. Example transition lines are also plotted (black points). The associated transition poses are shown in Fig. 5.5.

next time step, a dynamical model is used to disperse particles with maximum levels of diffusion, see Eq. 3.14 and the bottom of Fig. 3.2.

Each of the three activity models described in Section 5.3 is a candidate for the performance of this dispersion step. Their competing predictive properties are well summarised with reference to the "streetlight effect" [DTS+05]. Given a fixed allocation of N particles, the activity models for known activity and known activity transitions are analogous to narrow and bright streetlights illuminating small regions of the ambient pose space (via the latent pose space) with high sample density. The number of particles required to recover a solution is small, but if the true solution lies outside this region then the search is a futile endeavour. Alternatively, the activity model for unknown activities is analogous to a wide and dim streetlight illuminating a high-dimensional volume of the ambient pose space with low sample density. This streetlight should guarantee illumination encompasses the true solution, but the number of particles used must be large in order to ensure it is successfully recovered.

In the remainder of this section a multiple activity model annealed particle filtering (MAM-APF) method is proposed for the simultaneous consideration of complimentary activity models. This is achieved by assigning each activity model

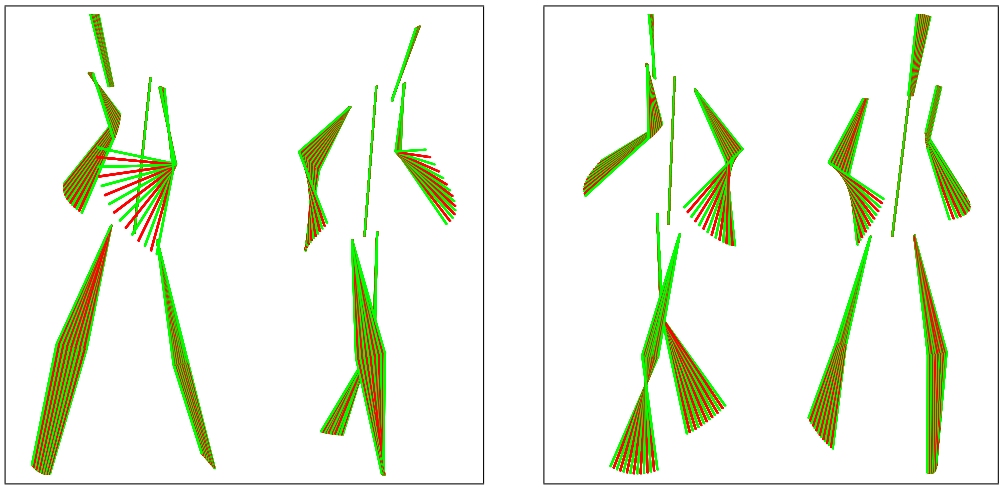


FIGURE 5.5: Modelling transitions between known activities. Poses reconstructed from the transition lines in Fig. 5.4: (left) individual-subject space; (right) joint-subject space. Each set of poses is shown from two rotated views.

its own unique quota of particles when re-initialising the particle set between frames. In each of the annealing layers that follow, a variable number of particles are drawn during resampling based on how well populated each activity model becomes. This approach ensures that enough particles are available to recover unknown activity via the ambient pose space, but that where known activities occur the latent pose space is not oversampled.

5.4.1 Simultaneous Activity Models

As a first step to supporting multiple activity classes during tracking, it is proposed that each of the three activity models described in Section 5.3 receive an equal allocation of N particles upon creation of the new (maximum dispersion) particle set at each frame. This constitutes an equal prior on each activity class. $3N$ particles are resampled from the particle set recovered at the previous time step $S^{\pi}_{t,0}$ and divided randomly between each of the three activity models to produce equal quotas of N particles. The activity models are then used to disperse their particle allocations, producing $S_{t+1,R}$. The result is a maximally diffused particle set that represents the predictions of all three activity models, which may be evaluated and resampled over successive annealing layers to recover a pose that maximises the objective function.

Particles are augmented by their activity model index $a_t^{(n)} = 1, 2, 3$ and are fully specified by the three parameters $(\underline{s}_{r,t}^{(n)}, \pi_{r,t}^{(n)}, a_t^{(n)})$. This index persists throughout the annealing run at each time step and ensures the particle is dispersed using its corresponding activity model at each layer. At each subsequent resampling stage just N particles are resampled. These particles may belong to any activity model and no quotas are enforced. By setting the particle number low, known activity and known activity transitions can be reliably and efficiently tracked in the latent pose space. However, this risks losing track where unknown activity occurs and the true pose can only be found by searching the ambient pose space. Conversely, by choosing N large enough to support ambient pose space search the latent pose space is oversampled during known activity, thus sacrificing any potential gain in efficiency.

5.4.2 Variable Particle Numbers

In order to increase the efficiency of the search, the approach described in Section 5.4.1 is modified to allow differently sized particle quotas to be allocated to each activity model. The activity models for known activities and known activity transitions (whose dynamical models span a low-dimensional latent pose space) are assigned a quota of $N_1 = N_2 = N_{\text{min}}$ particles each. The activity model for unknown activity (whose dynamical models spans the high-dimensional ambient pose space) is assigned a quota of $N_3 = N_{\text{max}}$ particles. The quotas reflect how many particles are required for each scheme to assume complete responsibility for tracking.

For creation of the new (maximum dispersion) particle set at each time step, every particle is dispersed by its respective activity model. The result is (as in Section 5.4.1) a maximally diffused particle set that represents the predictions of all three activity models. However, the equal prior on activity classes no longer holds, and the particle set is not suitable for resampling. For example, take the case where after dispersion takes place, every particle achieves the same objective

function score. If the distribution of particles between dynamical models is uneven due to the quota allocation, then the resampled particle set will contain the same disparity. This is despite the fact that each model's predictions explained the current observation equally well.

To address this problem two distinct measures are introduced: *effective particle* number; and *unique particle* number. $N_{\max}/N_{\min}$ effective particles are "stacked" at each of the $2N_{\min}$ unique particle locations in the latent pose space to give an equal number of effective particles in each activity model. By placing multiple particles at the same point, one is effectively returned to the approach described in Section 5.4.1, but only *one* objective function evaluation is required per stack. Resampling from this new particle set is no longer biased in favour of schemes with larger quota allocations, and subsequently resampled particle sets in the annealing layers that follow do not require stacking. A maximally dispersed particle set is shown in the bottom right of Fig. 5.6 with unique particle numbers shown in the legend followed by effective particle numbers in brackets.

Rather than resampling a fixed number of particles from the maximally diffused particle set, a variable number of particles are resampled based on activity model membership. With every particle that is resampled from a particular activity model $a = 1, 2, 3$, the value $N_{\max}/N_a$ is added to a `counter` parameter. Sampling continues to take place until the `counter` value reaches $N_{\max}$. The set of survival rates (see Eq. 3.7) are used to reshape the weighting distribution at each layer, just as in standard APF. The difference is that resampling of the distribution may terminate early: a maximum of $N_{\min}$ particles can be resampled from the latent pose space, or $N_{\max}$ particles from the ambient pose space. In general, a mixture of particles from the competing activity models are resampled. See for example the particles resampled from different activity models in layers $r = 4, 3, 1$ of Fig. 5.6; note that the total of their `counter` contributions (shown in brackets in the legends) always meets or just exceeds $N_{\max} = 250$.

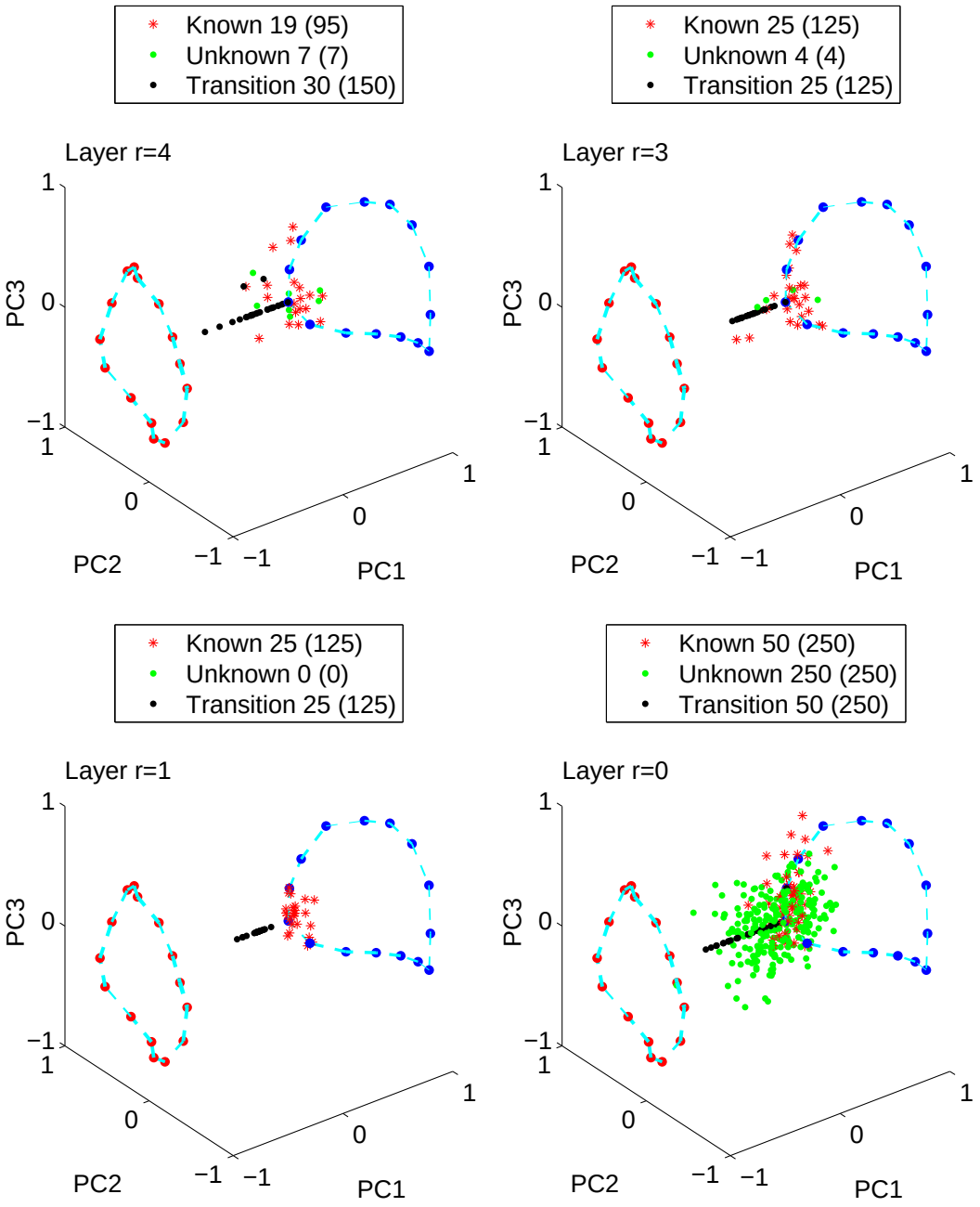


FIGURE 5.6: 3D view of MAM-APF particle dispersion over 5 layers for $T_0 = 4$ in the latent pose space. Layer $r = 2$ is omitted to maximise figure size. The observed pose is a *walk* pose. Multiple activity models are employed for each of known activity (red pluses), unknown activity (green points) and transitions (black points). Unknown activity hypotheses are projected into the latent pose space for visualisation. The numbers of resampled particles from each activity model are shown in the legends, with the `counter` contribution in brackets. For final layer $r = 0$ (maximal) dispersion the unique particle numbers are shown with effective particle numbers in brackets.

5.5 Experiments

In Section 5.5.1 each of the three activity models described in Section 5.3 are combined within an MAM-APF framework to recover the *HumanEva-II Combo* sequences which contain both known and unknown activity with transitions. The symmetric objective function of Section 4.3.3 is used and comparison is drawn with the use of standard APF [DBR00] using the same objective function. Body model dimensions are available for both subjects S2 and S4, but known activity training data is only available for S2. The scenario is therefore one of sequences of known and unknown activities performed by both known and unknown subjects.

For a particular activity, pose configurations in the *HumanEva-I* training data are given by series of global position vectors $\Omega = \{\underline{\omega}_1, ..., \underline{\omega}_M\}$ and pose vectors $Y = \{\underline{y}_1, ..., \underline{y}_M\}$ giving the relative joint rotations between limbs. The position vectors comprise $D_\omega = 6$ parameters, 3 rotational and 3 translational and the pose vectors comprise $D_y = 36$ Euler angles, every joint being permitted 3 degrees of freedom. A latent pose space dimensionality of $D_x = 4$ was again chosen for all experiments, resulting in a corresponding set of latent variables $X = \{\underline{x}_1, ..., \underline{x}_M\}$ related to the original pose vectors Y through a linear mapping.

HMMs of the form $\lambda = \{S, \boldsymbol{A}, \underline{a}, p_i(\underline{x})\}$ were estimated from latent variables using the steps described in Section 3.4.2 and were reestimated from a new initialisation before each individual tracking experiment. A time-reversed transition matrix $\hat{\boldsymbol{A}}$ was then calculated using Eq. 3.38, where the invariant distribution $\underline{\psi}$ is estimated by making 10^3 transitions via the original transition matrix, $\boldsymbol{A}$. Finite differencing of training data as described in Section 3.4.1 was used to estimate the covariance matrices, $\boldsymbol{P}_r^x$ and $\boldsymbol{P}_r^y$, used for dispersion at each layer. Note that these variables additionally facilitate tracking by standard APF.

During tracking five annealing layers and a constant survival rate of $\alpha_R = ... = \alpha_0 = 0.5$ were again adopted from the literature (Section 3.2.2 gives a discussion of the implications of varying these values). The complementary SSD scores Σ^s and $\Sigma^{\bar{s}}$ proposed in Section 4.3.3 were used in the calculation of particle weights.

In all experiments, the 3D absolute error between the expected tracking pose (see also Eq. 3.13) and a ground truth MoCap pose is calculated at each frame using Eq. 3.47.

5.5.1 Known and Unknown Activity using MAM-APF

The *HumanEva-II Combo* sequence for subject S2 was tracked using MAM-APF. Pose and position vectors for *walk* and *jog* were extracted from S2's portion of the *HumanEva-I Training* partition and PCA applied to recover a joint-activity latent pose space and associated HMMs for known activity tracking, see Fig. 5.4 (left). $N_{\min} = 50$ particles were assigned to each of the latent pose space activity models (known activity, known activity transitions) and $N_{\max} = 250$ particles to the ambient pose space activity model (unknown activity).

S2's *Combo* sequence was tracked five times from cameras C1 and C2 with $T_0 = 3$. 3D absolute tracking errors were calculated for each run using the online evaluation system [SB06a]. The final weighted particle set at each frame, $S^{\pi}_{t,0}$, was also used to perform a classification task. First each particle's activity model index was considered and the current pose classified as *unknown* if more than half belonged to the unknown-activity activity model. Otherwise every particle's parent HMM state in the latent pose space was found and the current pose classified as *walking* if more than half were assigned to λ_{walk} and *jogging* if more than half were assigned to λ_{jog}.

This test described above is a simple multiple-HMM classification task performed on each particle in isolation. It asks the question: given we observed this particular datapoint, which of the two HMMs was most likely to have emitted it? The details of classification between multiple HMMs are given in Appendix C and examples of their application to *sequences* of human poses in Appendix D. If there is uncertainty in the classification result then one option is to consider a longer state history including pose estimates at $t-1, t-2, \ldots$. Appendix D considers this case but finds only modest improvements in classification accuracy. The

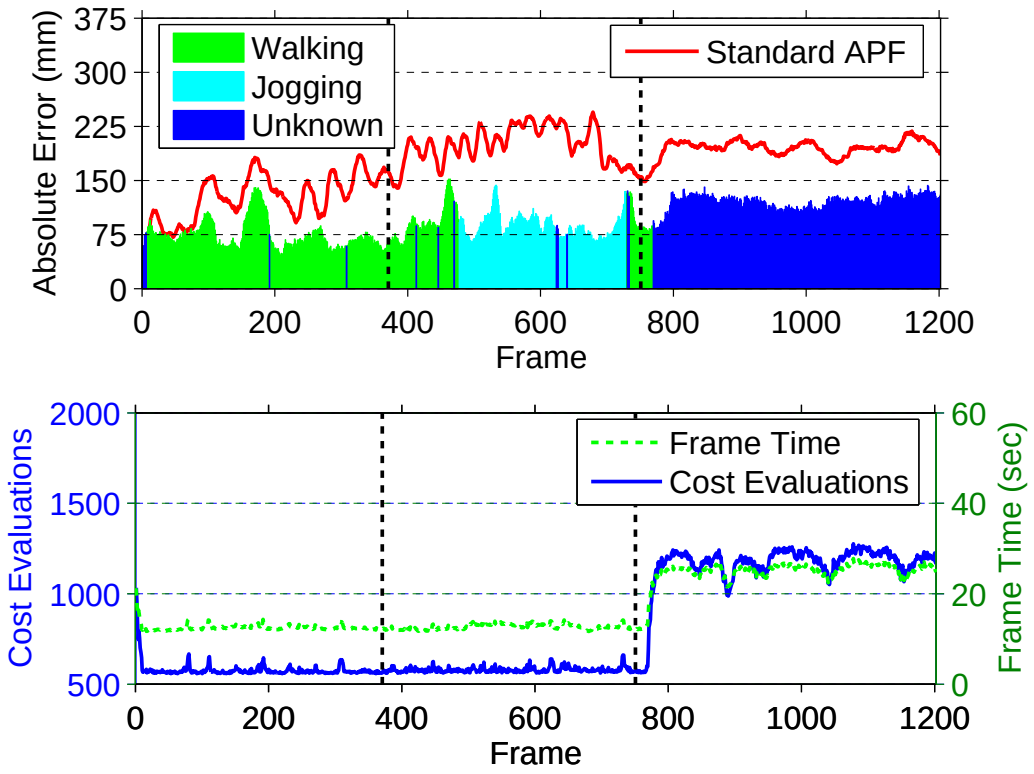
classification test proposed above was therefore not extended to include previous pose estimates as this is likely to introduce a classification lag during activity transitions, the possibility of genuine activity transformation being disregarded in the light of a consistent pose history.

Fig. 5.7 shows the average expected tracking error across the five runs at each frame, with the colour set according to the mode classification result across the five runs. The average number of objective function evaluations made at each frame and the average processing time required are also shown. The number of objective function evaluations is also equivalent to the number of unique particles used per frame (see also Section 5.4.2). Note that objective function evaluations remain low throughout the known activities before rising to recover the unknown activity. The subject's posture as they prepare to balance on one foot around frame 750 is indeed well described by a *walk* pose. Images showing the expected tracking pose superimposed on the image observations of HumanEva-II camera C1 are shown in Fig. 5.10.

5.5.2 Unknown Subjects

In the second experiment MAM-APF was used to track an unknown subject. A joint-activity joint-subject space was recovered from the training data of all three *HumanEva-I* subjects and a separate HMM trained for each subject's performance of each activity. By capturing the variation between subjects' performances, the aim was to maximise the ability of the latent pose space to generalise to new styles of known activity. The resulting activity model – shown on the right of Fig. 5.4 – was then used to track the *HumanEva-II Combo* sequence for the unknown subject S4, for whom no training data is available.

Using the same parameters as in Section 5.5.1, known activities were consistently and accurately recovered. However, the unknown *balance* segment proved more difficult. A failure mode – in which the subject's legs switch places – was regularly recovered at around frame 950. This appeared to be caused by strong shadows

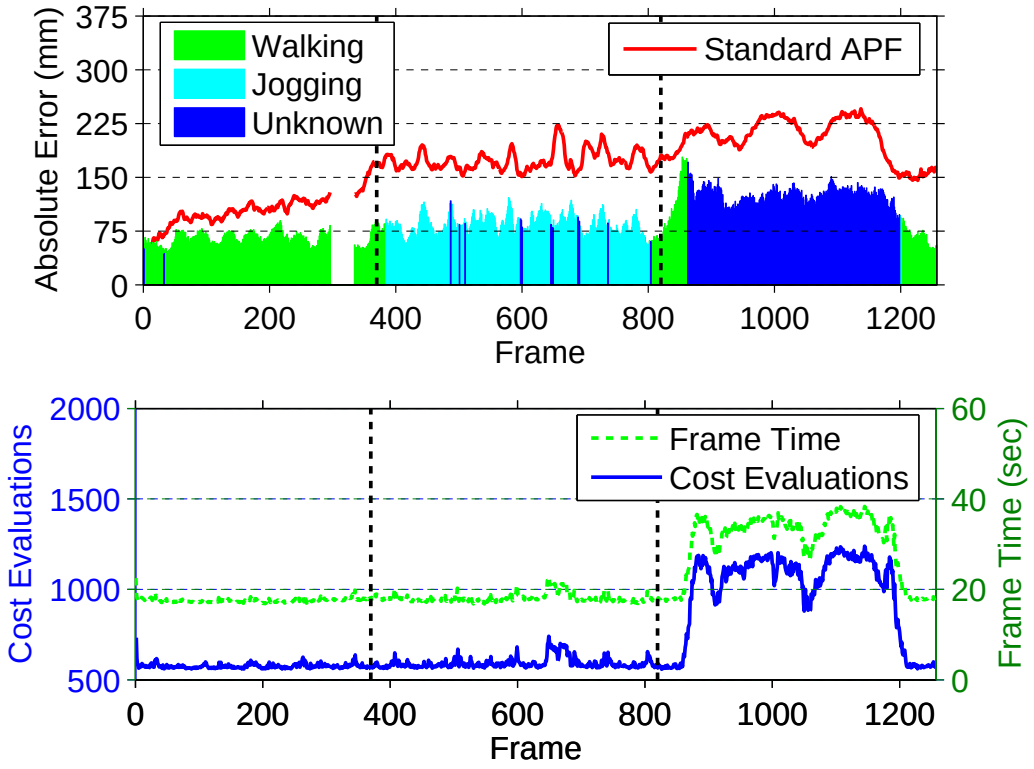


	Standard APF			MAM-APF		
	Walk	*Jog*	*Balance*	*Walk*	*Jog*	*Balance*
3D absolute err.	127 ± 32	199 ± 26	194 ± 12	74 ± 20	90 ± 19	119 ± 13
Objective evals.	1250	1250	1250	578 ± 42	579 ± 15	1154 ± 139

FIGURE 5.7: MAM-APF tracking results for S2's *HumanEva-II Combo* sequence: (top) 3D absolute error results averaged over five separate runs and colour coded by mode activity classification result; (middle) average number of objective function evaluations per frame and Matlab processing time.

cast onto the floor by the subject's lower legs and (incorrectly) included in the observation foreground mask, V^s. Neither increases in T_0 nor doubling of the unknown particle quota to $N_{\max} = 500$ enabled consistent recovery of the correct pose, and so a third camera was used (C1-C3) to obtain robust results. This failure mode highlights a potential drawback of the symmetric objective function, without which there would be no *requirement* to explain artefacts in the silhouette image.

Fig. 5.8 shows the average tracking error across the five runs at each frame. Just as for the known subject in Section 5.5.1, MAM-APF consistently outperforms the standard APF baseline and uses only half as many particles during the known activities. The final *walking* segment is a correct known activity classification, as



	Standard APF			MAM-APF		
	Walk	*Jog*	*Balance*	*Walk*	*Jog*	*Balance*
3D absolute err.	103 ± 24	173 ± 14	203 ± 28	65 ± 10	84 ± 14	117 ± 25
Objective evals.	1250	1250	1250	576 ± 14	592 ± 31	978 ± 234

FIGURE 5.8: MAM-APF tracking results for S4's *HumanEva-II Combo* sequence: (top) 3D absolute error results averaged over five separate runs and colour coded by mode activity classification result, frames 298-335 are ignored as accurate ground truth is not available; (middle) average number of objective function evaluations per frame and Matlab processing time.

the subject leaves their *balance* pose (around frame 1200) and starts walking out of the capture area. Images showing the expected tracking pose superimposed on the image observations of *HumanEva-II* camera C1 are shown in Fig. 5.11. Section 5.5.3 presents further work on the correction of activity misclassifications.

5.5.3 Projection-Reconstruction Error

Confusion between the two known activities is not a problem for the classification approach and there is therefore no need to pass longer state histories to the two HMMs (see also the investigation in Appendix D). It is possible, however, for the

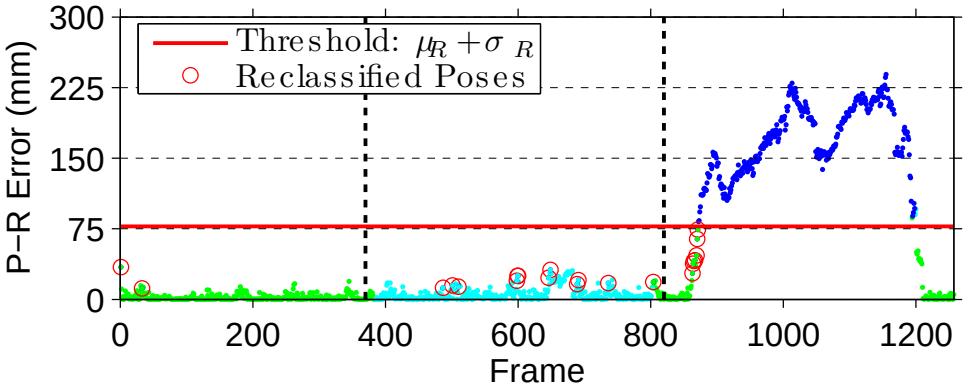


FIGURE 5.9: Average projection-reconstruction (P-R) errors for the S4 *Combo* results. By additionally requiring that "unknown" poses exceed a threshold P-R error, the misclassifications seen in Fig. 5.8 can be automatically corrected. Here the threshold is set as the average known activity reconstruction error for the latent pose space ($\mu_R = 58$mm) plus one standard deviation ($\sigma_R = 20$mm).

unknown activity model to do the work of the known activity model by recovering known poses from the ambient space, see the occasional unknown (blue) frames during the *walk* and *jog* phases in Figures 5.7 and 5.8. Such "overlap" between activity classes can be identified by reconstructing a recovered unknown pose from its projected coordinate in the latent space and then calculating a projection-reconstruction (P-R) error between the original and the reconstruction (using Eq. 3.47). Misclassifications can be automatically identified and reclassified by requiring that unknown poses exceed a lower bound on P-R error given the latent pose space. For example, in Fig. 5.9 all unknown expected poses that have a P-R error below the latent pose space's average known activity reconstruction error are reclassified by comparison of the associated particle sets between the two known activity HMMs (as described in Section 5.5.1). Alternatively, P-R error thresholding could be used as a prior on particle dispersion by the unknown activity model, resampling until all ambient pose space configurations are "novel" given the latent pose space.

5.6 Discussion and Conclusions

MAM-APF gives equal consideration to the predictions of multiple activity models at each frame. The difficulty of the associated estimation tasks is quite different and this has allowed the recovery of known and unknown activities using a variable number of particles. Here (and also in Chapter 7, Section 7.2) further discussion is given to some specific aspects of the proposed approach.

5.6.1 Tracking Performance

MAM-APF is able to reliably recover known activities from the *HumanEva-II Combo* sequences with fewer than four cameras and a reduced number of particles. This is in contrast to standard APF, see the quantitative comparisons in Figs. 5.7 and 5.8 and the investigation by Sigal *et al.* [SBB10]. MAM-APF is also able to increase particle numbers to recover the *balance* phases with its unknown activity model. Estimating freeform motion in the high-dimensional ambient pose space with a generic dynamical model is inherently more challenging, and the average 3D absolute error rises by 30-50mm. In general, however, a good track is maintained throughout the sequences, see Figs. 5.10 and 5.11.

The recovery of a failure mode when using 2 cameras to track S4's *balance* phase illustrates a potential danger of using the annealing methodology where image evidence is ambiguous: if an incorrect mode is recovered, tracking may never be regained. However, it should be noted that particle filtering has been found to perform significantly worse than standard APF on the *Combo* sequences [SBB10], despite its capacity to approximate a multimodal posterior over time. Furthermore, robust 2-camera tracking of S4's *balance* phase is likely to be possible if some consideration is given to the effects of shadows cast by the lower legs. The addition of feet to the body model may be helpful e.g. [SBB10], or more sophisticated background subtraction methods could be adopted [HD04].

5.6.2 Classification

The MAM-APF approach naturally lends itself to sequence classification based on the activity model membership of particles. Figs. 5.7 and 5.8 show the algorithm is able to correctly classify frames from the *Combo* sequences into their particular activity classes with reasonably few exceptions. The dashed vertical lines in these figures represent the ground truth activity segmentations defined by Sigal *et al.* [SBB10], and used in the calculation of the error tables. Misclassifications are generally due to the unknown activity model recovering a known activity pose. Sigal *et al.* [SBB10] note that S4's *jog* phase displays a greater variation in performance style. This may explain why a slightly higher number of S4's *jog* poses were recovered by the unknown activity model than for S2. No problems were experienced with false known activity transitions, e.g. [DLC08a].

Rather than clear and instantaneous changes between activities, the *Combo* sequences feature a number of slow activity transitions (relative to sampling rate) where intermediate poses do not feature in the *HumanEva-I Training* dataset (in which activities are segmented). S2's transition from *walk* to *jog* takes place over a period of approximately one second, starting with an abrupt rise in the forward swing of the left forearm that appears to increase vertical displacement (frames 380-400) before the subject eventually settles into a jog by around frame 440. In the absence of training data, it is the transition activity model that facilitates the maintenance of a track, and permits the recovery of intermediate poses from the space in between the two activity manifolds (e.g. see Fig. 5.5). During this period however, the mode classification result of MAM-APF remains as *walking* up until the *jog* gait is fully established at around frame 440. More accurate identification and recovery of activity transitions themselves is a potentially interesting future research topic.

5.6.3 Computational Cost

The computational cost of generic particle filtering is proportional to the number of particles used. For APF it is proportional to the number of particles used across all annealing layers. Total runtime is dominated by the evaluation of the objective function for each particle. As the objective function must be evaluated for each observation, computation time is also proportional to the number of cameras. For the standard APF baseline computation times are constant at around 25 and 40 seconds per frame for two and three cameras, respectively.

This work has addressed the high and fixed computational cost of particle-based inference by varying the number of particles depending on their activity class membership. For *Combo* sequences, the number of objective function evaluations remains low throughout the known activities of *walk* and *jog* as poses are recovered from the latent pose space. The number of evaluations then rises as the ambient pose space is explored to recover the unknown *balance* activity, see tables in Figs. 5.7 and 5.8. Computation times fall by around 50% to 15 and 20 seconds per frame when tracking known activity from two and three cameras, respectively.

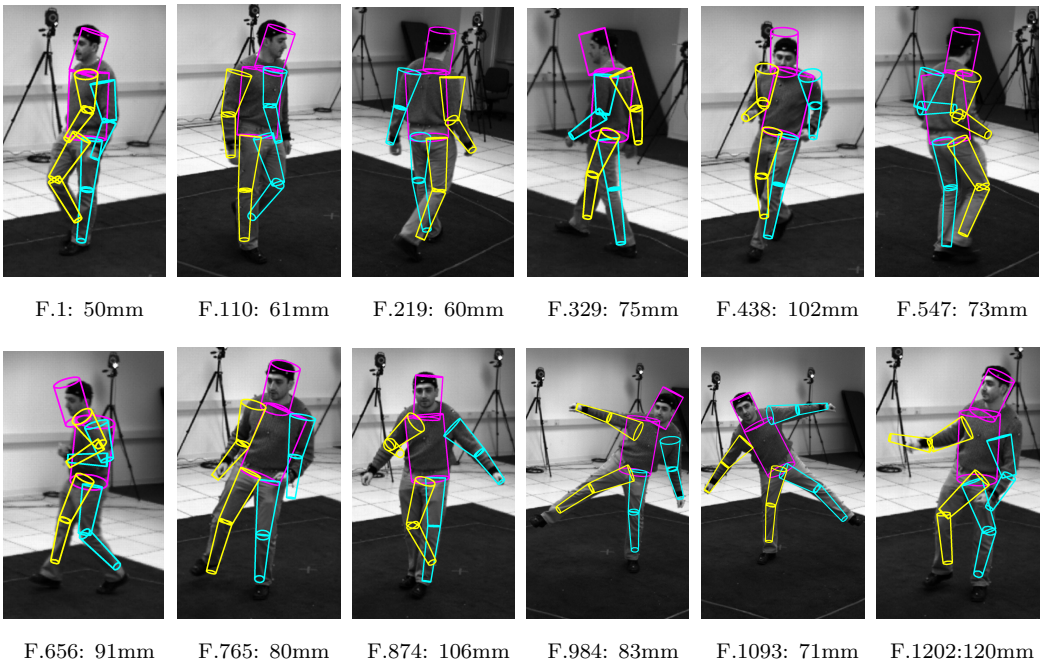


FIGURE 5.10: Tracking results for the S2 Combo sequence using two cameras.

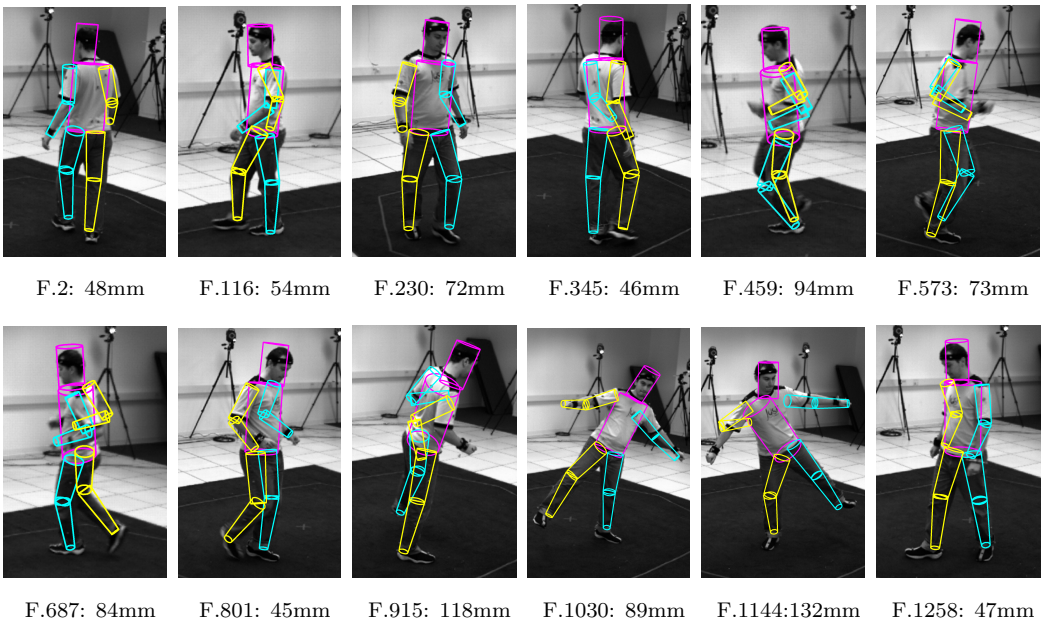


FIGURE 5.11: Tracking results for the S4 Combo sequence using three cameras.

Chapter 6

Composite Activity

In Chapter 4 a full body or "global" latent pose space is recovered from training data. Sampling such a space results in coordinated full body poses. In Chapter 5 this global latent space is combined with an ambient state space search in which each parameter of the body model is fully independent and free to produce uncoordinated poses. This chapter investigates the use of a hierarchy of latent variables during inference: the hierarchical Gaussian process latent variable model (H-GPLVM) [LM07]. A particle-based approach is used to "back off" through the model and exploit progressively greater independence between body parts to recover unknown activities. At its top level the H-GPLVM's root node is effectively a single low-dimensional global pose space approximating poses from the training set. At the bottom level its leaf nodes define conditional distributions over the high-dimensional ambient state space. As such, it can be used as a route between the two pose spaces. The extent to which the final pose estimate is constrained by the known activity training data depends on the extent to which correlations between the latent model's nodes are respected during the descent. Long range correlations can be used to infer the positions of occluded limbs, alternatively they may be disregarded in order to recover novel unknown activity poses.

6.1 Introduction and Related Work

Low-dimensional models of activity can be employed to effectively constrain the search task in generative human motion tracking. This was demonstrated using a form of APF guided by an HMM in Chapter 4 and has also been achieved

using deterministic optimisation and particle filtering (e.g., [LYST06, LPS07, RRR08a, SJ04, UFF06a, UFHF05]). Some of these approaches show a capacity to generalise to variations in style, or unknown subjects. Hou *et al.* [HGC+07] tracked an unknown subject performing jumping jacks. Urtasun *et al.* were able to track unknown walking subjects [UFF06b], the golf swings of unknown subjects [UFHF05] and an exaggerated walking style with increased stride length and rigid limbs [UFF06a]. However, when the activities to be tracked deviate significantly from those in the training data, these full-body or "global" models are unable to cope and tracking fails (e.g., [SJ04]). The central argument put forward in this chapter is that some capacity to relax the constraints of full-body models and exploit conditional independencies in the kinematic tree is desirable.

Chapter 5 introduced a combined low-dimensional and high-dimensional tracking approach to address the problem of unknown activity tracking. The *balance* portion of the *HumanEva-II Combo* sequences provides a somewhat extreme example of unknown activity that is unusual in the context of the *HumanEva-I* training partition but also in the more general sense. Departure from training activities may be far more subtle. A useful example is provided by activity combinations such as *walk whilst waving*. Although the component parts of an activity may be present in training data – e.g. *walk* activity + *wave* activity – the global nature of the latent pose space precludes tracking. If observations are sufficiently rich then tracking may be achieved by relying on a separate generic high-dimensional approach (e.g. [GD96, MI00, DBR00, ST03a] and the work presented in Chapter 5) but each pose parameter is fully independent and the prior model of correlations provided by activity training data is sacrificed. Similarly, part-based models (e.g. [FH05, LC04]) could be used to find kinematically feasible 2D solutions but do not capture long range correlations, only those between neighbouring limbs[1]. It is unclear how either approach might cope with occlusion, for example.

The findings of this chapter show that with a learned hierarchical model of body coordination for multiple activities, one can recover novel poses that comprise

[1]Loopy graphical models [SBR+04] can be used to permit more expressive constraints on 3D pose but greatly increase the cost of inference.

aspects of different activities, or composite activity. An H-GPLVM [LM07] (see also Section 3.3.5) is constructed by learning separate low-dimensional models for the variation in individual body parts and then augmenting them with further latent variable models capturing their coordination. Using a form of annealed particle filtering that includes a crossover operator (Section 6.3), it is shown that the H-GPLVM learned from two or more activities can be used to recover novel test poses. The approach presented is intended to be a compromise between the restrictions of a low-dimensional full-body activity model and the challenges of searching the high-dimensional state space of the body model.

Inference proceeds gradually (via a number of GPs) from the top to the bottom of the hierarchy. The intention is a gradual progression between a single global latent pose model (root node), through a number of increasingly short-ranged part-based models (intermediate and leaf nodes), to the original ambient pose space; the dimensionality of the state vector increasing with depth. It is the use of a crossover operator that allows for the recombination of different pose elements at each layer. Without it poses are always limited by the global poses that have performed well at the top of the hierarchy. This is a useful contrast with other hierarchical models, e.g. that of Karaulova *et al.* [KHM00] where a single global pose is then refined by "fine tuning" in a number of part based latent models or the related approach taken by Raskin *et al.* [RRR09] (see also Section 2.3.2.2).

The proposed approach allows the recovery of activity combinations such as *walk whilst waving* but can also recover less intuitively obvious composite poses, see for example Section 6.5.1. The hierarchical decomposition of articulated human motion data is useful because unlike other tracking targets – e.g. the deformable materials considered by Salzmann *et al.* [SUF08] using "flat" part-based models – there is benefit in appealing to the longer-range correlations between local models. For example, the position of an occluded arm is inferred using such correlations in Section 6.5.4. Disregarding the higher levels of the hierarchy and performing a "leaves-only" search would result in a randomly flailing arm, rather than one that is correlated with the visible upper body (see also Fig. 6.12).

Although the approach presented in this chapter is tested on tracking-style problems, such as continuous video featuring people walking, it is in fact a 3D *pose estimation* technique that is independent at each frame. The pose is not manually initialised at the start frame, nor is the pose recovered at one frame used to initialise the search at the next frame. Instead of testing only poses that are close (in terms of the state space) to the previous estimate, a considerable level of pose diversity is intentionally introduced to permit the recovery of novel composite configurations not present in the training data.

Performing 3D pose estimation (rather than recovering small inter-frame changes in pose, or *tracking*) from silhouettes using generative "synthesise-and-test" techniques is known to be very challenging (see also the discussion given in Section 2.3). Experience with noisy range data (see also known activity tracking experiments in Section 4.4.1.2) suggests it may be *at least* as challenging without some attempt to define a *symmetric* objective function. For these reasons the techniques described in the remainder of this chapter are restricted to "lifting" problems where 3D poses are inferred from known 2D joint locations, e.g. [UFHF05, UFF06a]. A possible future extension of the approach to tracking scenarios is suggested in Section 7.3.2.2.

The *HumanEva* sequences tested in previous chapters do not feature examples of composite activity. For example, the component joint angles of the *HumanEva-II balance* poses (alternate legs raised high with arms horizontally out at the sides) cannot be reconstructed as piece-wise combinations of *walk* and *jog* nor of any of the other *HumanEva-I* training activities: the individual body part configurations necessary are simply not present. Alternatively, although separate *walk* and *wave* activities are available, there is no *walk whilst waving* test data. For this reason 2D joint locations were extracted from new monocular test sequences featuring simple composite activity by using the $\mathcal{WSL}$ tracker [JFEM03] (see also Section 3.5.3.1) and training data was selected from the larger CMU MoCap repository [CMU].

The main contributions of this chapter are as follows:

- Implementation of the "back off" procedure suggested by Lawrence and Moore [LM07] using a form of annealed particle filtering with crossover operator (Section 6.3).
- Demonstration of the composite nature of a number of activities from the CMU MoCap database (Section 6.5.1 and Section 6.5.2).
- Monocular tracking of composite activity from 2D $\mathcal{WSL}$ data (Section 6.5.3).
- Recovery of occluded limbs by terminating back off above the hierarchy's leaf nodes (Section 6.5.4).

6.2 Hierarchies of Latent Variables

The GP-LVM [Law05] (see also Section 3.3.4.2) represents high-dimensional data through a low-dimensional latent model, and a non-linear Gaussian Process (GP) mapping from the latent space to the data space. This makes it ideal for the representation of human motion data. The GP-LVM exploits a probabilistic interpretation of PCA as a product of independent GP models over features, each with a linear covariance function [Law05]. By the consideration of non-linear covariance functions, such as a radial basis function kernel, non-linear latent variable models can be formulated. Optimising the latent variables (initialised with PCA) and kernel parameters given the set of high-dimensional training points results in a probabilistic model of the original data.

The H-GPLVM [LM07] (see also Section 3.3.5) is a form of GP-LVM with a hierarchical latent representation. The leaves of the latent model comprise a latent model for each limb or distinct part of the body. That is, each leaf node is a GP-LVM for a single body part. To capture the natural coordination of body parts one can then model the joint distribution over latent positions in a subset of leaf nodes with a GP from a parent latent variable. The hierarchical decomposition used in this chapter is shown in Fig. 6.1 with the direction of the

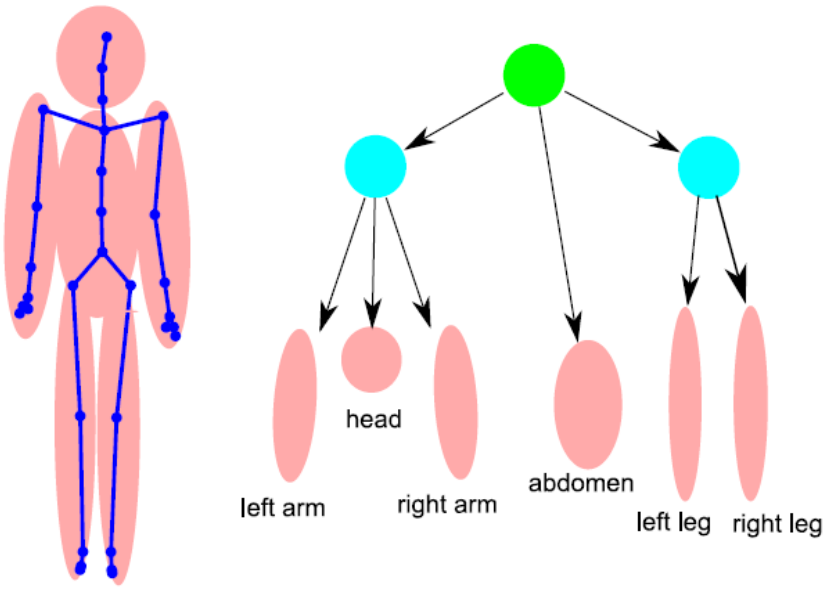


FIGURE 6.1: Skeleton and hierarchy of latent variables [LM07].

GP mappings between levels shown by arrows. Fig. 6.2 shows an H-GPLVM of the same form learned from motion capture data. The *left leg* and *right leg*, for example, are coordinated by the *lower body* latent variable. Given a lower-body latent position, there is a GP mapping (see also Section 6.2.1) to latent positions for the left and right legs, from which there are GP mappings to the joint angles of the two legs.

6.2.1 Data Generation

To aid the exposition of inference in the H-GPLVM latent positions in non-leaf nodes are sometimes referred to as specifying *partial* or *full-body* poses in the original ambient pose space. Strictly speaking, there is no direct connection between the two, and implicit in these statements is the assumption that the probabilistic mappings between parent and child are used to fully descend the hierarchy through the leaf nodes to the ambient pose space.

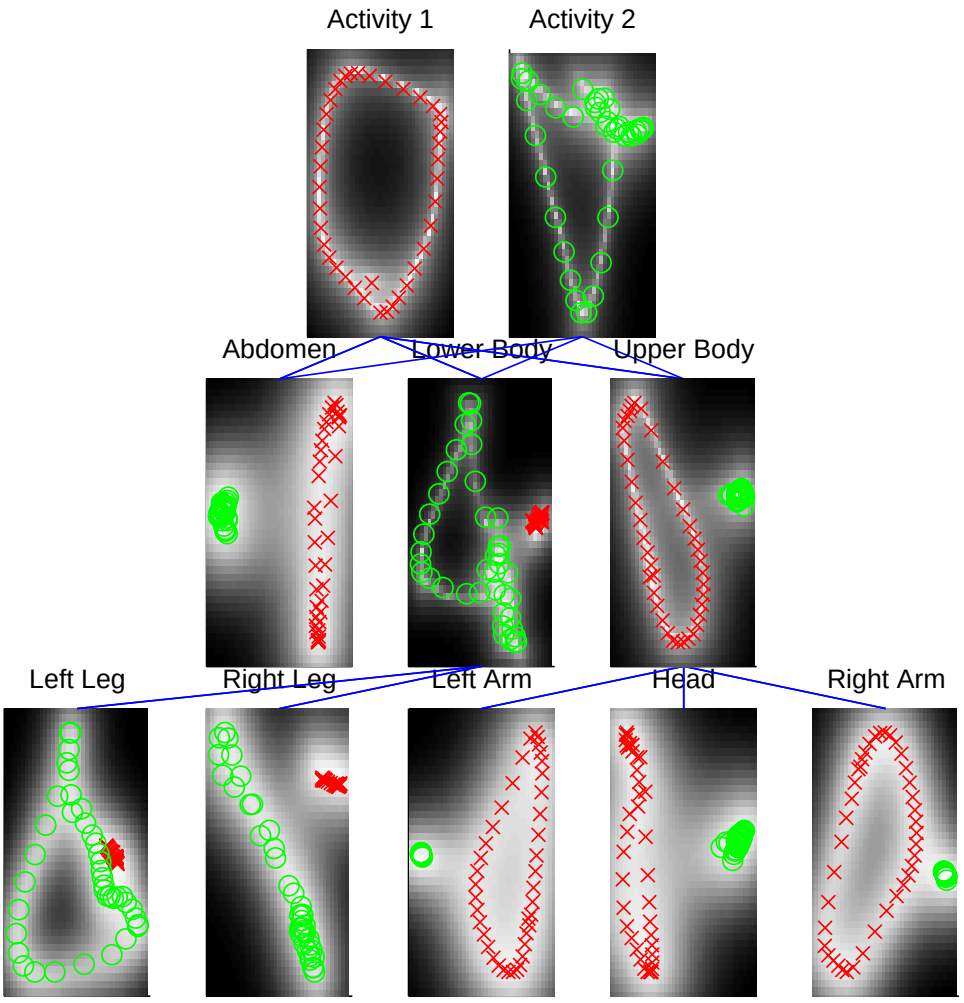


FIGURE 6.2: An H-GPLVM learned from two activities, namely *swing arms* and *walk with hurt stomach* (illustrated below in Fig. 6.7).

For GP-LVMs a new latent position $\underline{x}_*$ can be shown [Law05] to project into the data space as a Gaussian distribution

$$p(\underline{y}_*|\underline{x}_*) = N(\underline{y}_*|\underline{\mu}, \sigma^2\boldsymbol{I}). \tag{6.1}$$

Whose mean is

$$\underline{\mu} = \boldsymbol{Y}^\top \boldsymbol{K}^{-1} \underline{k}_{:,*} \tag{6.2}$$

where $\boldsymbol{K}$ is the kernel matrix developed from the training data and $\underline{k}_{:,*}$ is a column vector developed from computing the elements of the kernel matrix between the

training data and the new point $\underline{x}_*$. The variance is then given by

$$\sigma^2 = k(\underline{x}_*, \underline{x}_*) - \underline{k}_{:,*}^\top \boldsymbol{K}^{-1} \underline{k}_{:,*}. \tag{6.3}$$

Within the context of an H-GPLVM, $\underline{y}_*$ may describe a *further* set of concatenated latent coordinates defining positions in each of a node's children.

Fig. 6.3(a) illustrates the idea of implicit descent through the hierarchy. A fully coordinated *swing arms* pose has been generated by selecting a *single* latent point in the root node. The hierarchy is the same one shown in Fig. 6.2 with training data for *swing arms* and *walk with hurt stomach* depicted by red crosses and green circles, respectively. The latent point responsible for the pose is shown by a solid blue circle in the top left root node. Given this point, a *set* of dependent coordinates in the root node's immediate children can be found via the GP mapping described above (the mean position has been used in the figure, see Eq. 6.2). The mappings between parent and children can be used to recursively descend the levels of the hierarchy to the leaf nodes and then to the ambient space. The set of latent coordinates that arise from the single root node coordinate are also shown by solid blue markers, one in every dependent node. Fig. 6.3(b) shows a *walk with hurt stomach* pose reconstructed by descending from a single point in the top right root node.

6.3 Activity Model Definition

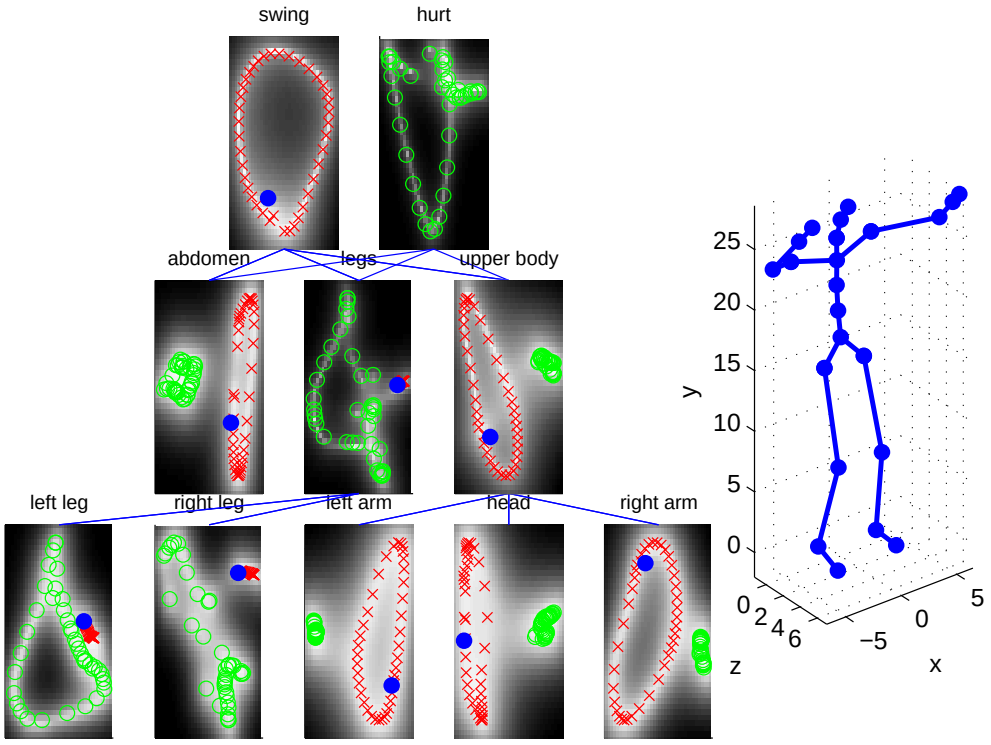
Rather than a single latent pose space as in previous chapters, the model of pose used in this chapter consists of a *set* of latent pose spaces. For example, for the model depicted in Fig. 6.2 there are ten latent spaces, the indices of which are given by the set $L = \{1, 2, ..., 10\}$. Pose hypotheses are defined by a position vector and a *collection* of latent vectors, each defining a coordinate in an *active* space. To avoid limiting the discussion to a particular hierarchical decomposition this is written simply as

$$\underline{s}_t = [\underline{\omega}_t, \{\underline{x}_t^i\}_{\forall i \in \mathcal{A}}], \tag{6.4}$$

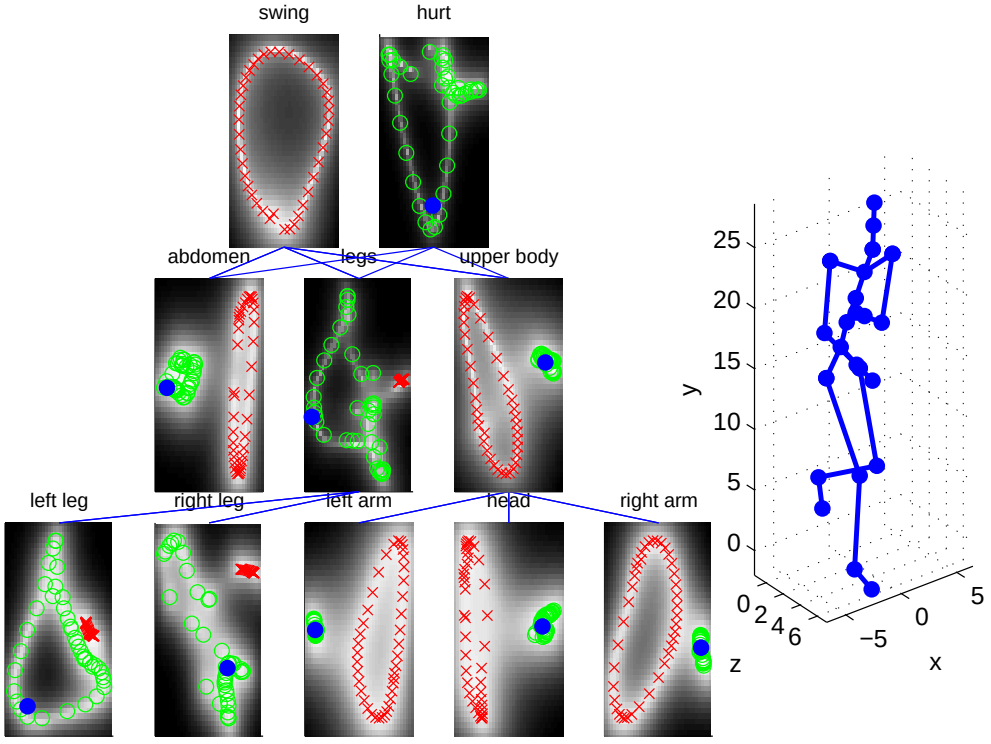
where $\mathcal{A} \subset L$ is the subset of active spaces. The only condition for a complete pose hypothesis is that every leaf node either be active, or have at least one active ancestor. The method for pose generation described in Section 6.3.1 guarantees this is always the case. In terms of the particle set it is helpful to imagine a set of N *related* particles in *every* active space, each defining a *partial pose*. The collection of $|\mathcal{A}| \times N$ locations combine to describe N full-body pose hypotheses, where $|\mathcal{A}|$ gives the total number of active spaces.

To exploit the hierarchical structure of the H-GPLVM Lawrence and Moore [LM07] suggest that a "back off" method inspired by language modelling might be used for the recovery of poses not featured in the training set. The idea is to descend the hierarchy and search nodes at the next level *independently*; this concept forms the basis for inference in this chapter. By shifting search down one level in the hierarchy the level of coordination amongst body parts can gradually be relaxed. While recovery of a novel test pose by inspection of full-body models at the root nodes may not be possible, a good fit might be obtained by backing off to the middle level nodes to optimise the abdomen, upper body and lower body independently.

Fig. 6.4(a) illustrates the introduction of independence between parts of the body. A *swing arms* pose has been instantiated from a single point in the top left root node, just as in Fig. 6.3(a). However, the point that results in the "legs" node of the middle level of the hierarchy has been deliberately shifted to lie near the *walk with hurt stomach* latent data. This new coordinate, its dependents, and the body parts that are affected are shown in magenta. The resulting composite pose (shown on the right of the figure) is the product of two *independent* latent coordinates. In Fig. 6.4(b) the pose has been further modified by the introduction of a third independent latent coordinate (shown in cyan) in the left arm's leaf node. By backing off to probe nodes at a given layer of the hierarchy independently, correlations present in the training can be broken, different activities recombined, and novel poses recovered.

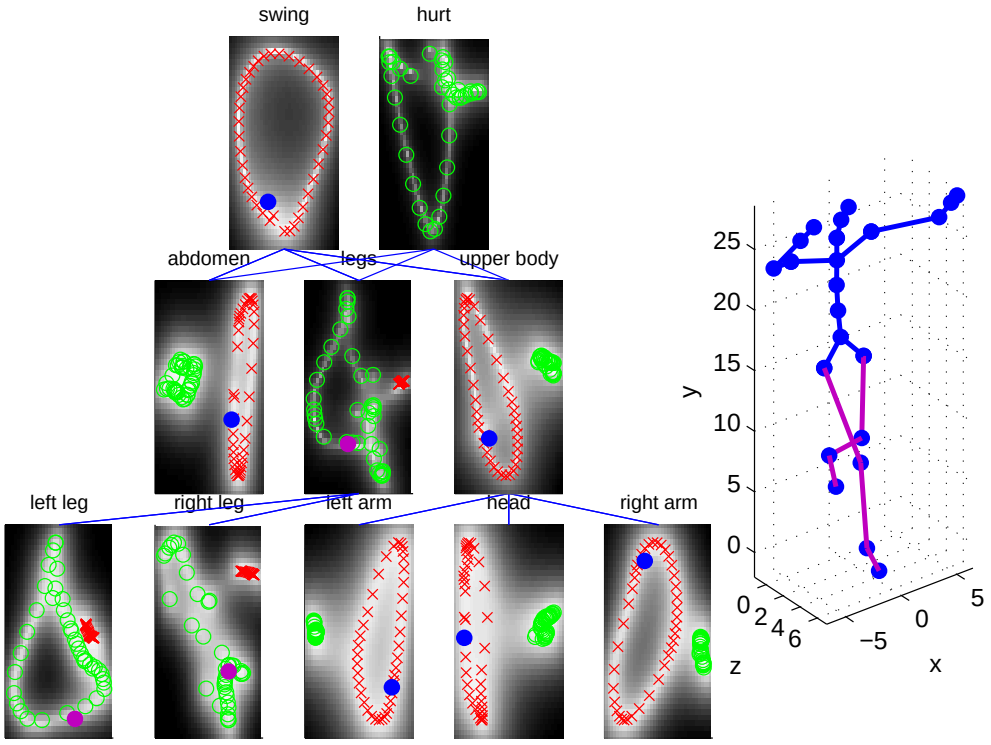


(a) Fully coordinated *swing arms* pose generated from a single point in the top left root node.

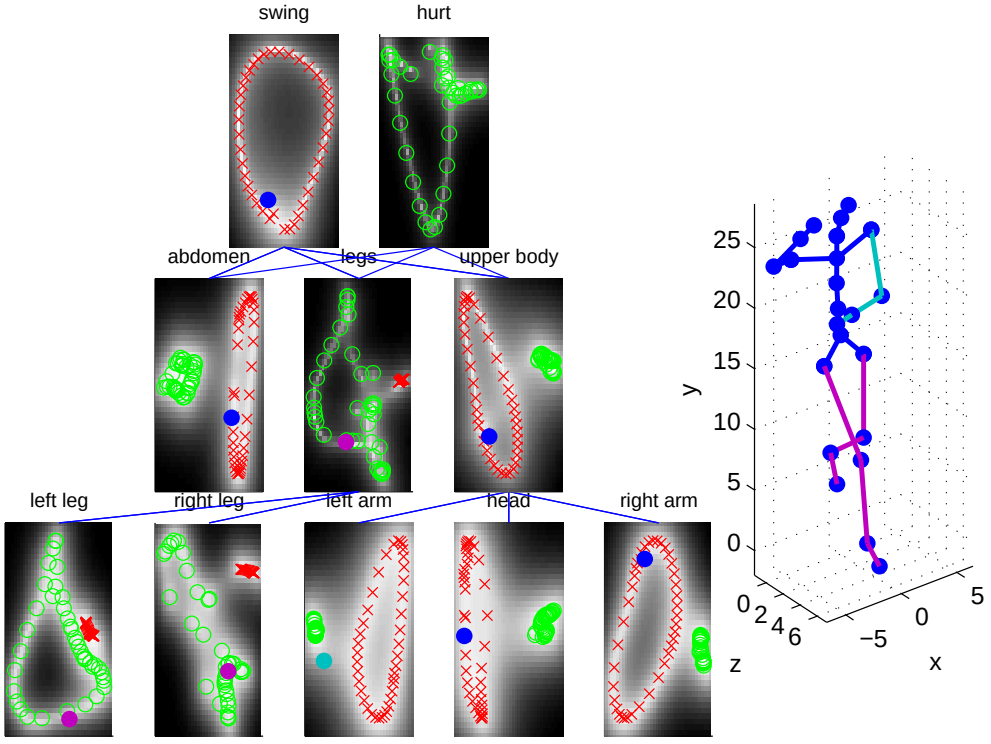


(b) Fully coordinated *walk with hurt stomach* pose generated from a single point in the top right root node.

FIGURE 6.3: Pose generation.



(a) Composite pose showing independence between lower and upper body.



(b) Extension of the pose in (a) to show independent movement of a single body part via a leaf node.

FIGURE 6.4: Composite pose generation.

To cope with novel poses not explicitly present in the training data, following Deutscher and Reid [DR05], a form of *crossover operator* is introduced to recombine the *building blocks* of particles that have performed well. This type of mechanism is ideal for the exploitation of reduced levels of coordination between limbs. For novel poses it will be necessary to retain full-body poses from the top level nodes of the hierarchy that are somewhat *flawed*. That is, even poses that show comparatively poor agreement with observation data may prove valuable in lower levels of the hierarchy, since they fit some but not all parts of the body well. The annealing schedule of APF is able to support a wide range of hypotheses in the early annealing layers before eventually concentrating on a particular solution in later layers.

6.3.1 Composite Activity

The particle-based search proceeds from the top to the bottom of the H-GPLVM over a number of annealing layers, with back off occurring after each resampling stage. The reader may find it helpful to refer to the hierarchical decomposition of two activities used in this chapter and depicted in Fig. 6.2, but the method is also applicable to other decompositions.

Given a new observation $\underline{z}_t$, maximal dispersion proceeds as follows. A set of N particles is initialised by uniformly sampling from the latent representatives of the training set at the root nodes. Each particle holds a latent position that may be used to completely descend the hierarchy and recover a full-body 3D pose. To allow these initial poses to depart from the training data, each particle's latent position is then perturbed with the addition of a zero mean Gaussian random variable $\underline{n}_0^x$ with covariance $\boldsymbol{P}_0^x$. Each corresponding pose $\underline{s}_{t,R}^{(n)}$ is then *evaluated* against the observation using $w_R(\underline{z}, \underline{s})$, and N particles are *resampled* with likelihood in proportion to their weights and with replacement.

Given the latent position held by each resampled particle, the H-GPLVM defines Gaussian conditional distributions over the child nodes in the level below (see

also Section 6.2.1). To exploit the potential for independence between these latent spaces (and therefore body parts) in the *dispersion* step, a new particle set is constructed by applying a crossover operator as follows. A single sample is drawn from the conditional distributions corresponding to each particle, yielding N new latent positions in each child node. N new particles, each holding a *set* of latent positions are then created by randomly sampling once from the new latent positions in each of the child nodes, without replacement. Subsequent annealing layers $r = R-1, ..., 1$ proceed to back off down the hierarchy in just the same way, but are initialised with the new particle set from the previous layer.

Where a pose observation features occlusion or self occlusion it is desirable to infer the location of an occluded limb from visible limbs based on their correlations within the training data. Where $\underline{z}_t$ takes the form of a set of labelled 2D features e.g. [JFEM03] or [Ram06] and one or more are absent, the implication for the search strategy is as follows: where image evidence for a subtree of the skeleton is missing, descent should not pass below that subtree's parent node. In Section 6.5.4 this principle is used to recover the occluded arm of a walking subject. Otherwise, back off ceases to take place only once the leaf nodes are reached.

In practice, the covariance of the GP mappings from parent to child in the H-GPLVM are often relatively small. This is due in part to the regularisation conditions (see Section 3.3.5) and in part to the use of only one activity cycle. To encourage individual body parts to depart from the training data and increase pose diversity, the covariance is artificially inflated to be equal to $\boldsymbol{P}_r^x$ where,

$$\boldsymbol{P}_r^x = \alpha_R \times ... \times \alpha_r \times \boldsymbol{P}_0^x. \tag{6.5}$$

This weighted dispersion term is also applied to latent positions in nodes where back off has ceased to take place. Individual particle dispersion is summarised in Fig. 6.5 and the application of the crossover operator to the particle set in Fig. 6.6.

1. The position of the (n)th particle in the rth layer is given by the position and latent parameters, $\underline{s}_{t,r}^{(n)} = [\underline{\omega}_{t,r}, \{\underline{x}_{t,r}^{i}\}_{i \in \mathcal{A}}]$.

2. The particle's position parameters $\underline{\omega}_{t,r}$ are updated by the addition of the Gaussian random variable $\underline{n}_{r}^{\omega}$,

$$\underline{\omega}'_{t,r} = \underline{\omega}_{t,r} + \underline{n}_{r}^{\omega}. \tag{6.6}$$

3. For $r = 0$:

   - The particle's latent parameter is initialised by activating a single root node, giving $\mathcal{A}'$, and sampling a single latent variable (training pose).
   - This is then perturbed by the Gaussian random variable $\underline{n}_{0}^{x}$ to give $\{\underline{x}_{t,r}^{i}\}'_{\forall i \in \mathcal{A}'}$.

4. For $r > 0$:

   - Each of the particle's latent parameters is used to descend to the next layer of the latent hierarchy using the mean mapping $\underline{\mu}$ (see also Eq. 6.2). This gives a new set of latent coordinates in a new set of active nodes, $\mathcal{A}'$. Descent does not take place at the leaf nodes or where the limb(s) controlled by the child node are occluded.
   - Each of the particle's latent coordinates is perturbed with the scaled Gaussian random variable $\underline{n}_{r}^{x}$ to give $\{\underline{x}_{t,r}^{i}\}'_{\forall i \in \mathcal{A}'}$. This allows body part configurations to depart from the training data.

5. The new estimates are then used to create a particle in a new set

$$[\underline{\omega}'_{t,r}, \{\underline{x}_{t,r}^{i}\}'_{\forall i \in \mathcal{A}'}] = \begin{cases} \underline{s}_{t,r-1}^{(n)} & \text{if } r > 0; \\ \underline{s}_{t+1,R}^{(n)} & \text{if } r = 0. \end{cases} \tag{6.7}$$

FIGURE 6.5: Dispersion of a single particle for composite activity tracking.

6.4 Objective Function

Hou *et al.* [HGC+07] have proposed a suitable objective function for particle-based inference from $\mathcal{WSL}$ data (see also Section 3.5.3.1). This is defined as the sum of the squared 2D Euclidean distances between corresponding pairs of joint centres in the observation $\underline{z} = \{\underline{w}_i\}_{i=1}^{\mathcal{M}}$ and in the hypothesised body model configuration $\underline{b}$,

$$\Sigma^{\mathcal{WSL}} = \sum_{i=1}^{\mathcal{M}} ||l_i(\underline{b}) - \underline{w}_i||^2 \tag{6.8}$$

where $l_i()$ returns the 2D location of the ith joint centre.

1. For $r > 0$: The latent parameters of each particle in the unweighted set define a total of $|\mathcal{A}| \times N$ latent coordinates across the A active nodes.
   - A single new particle is created by randomly sampling a single coordinate from each of the $|\mathcal{A}|$ active latent spaces.
   - Sampling continues *without* replacement until all latent coordinates have been resampled.
   - The result is a new particle set defining N unique full-body poses.

FIGURE 6.6: Action of the crossover operator on the particle set.

6.5 Experiments

In each experiment H-GPLVMs were trained (see also Section 3.3.5) from pose vectors $Y = \{\underline{y}_1, ..., \underline{y}_M\}$ recovered from MoCap data [CMU] and decomposed as shown in Fig. 6.2. The rotational components and vertical displacement of the position vector were moved into the pose vector, giving $D_y = 50$. Only in experiments where the subject walks across the image was a separate (horizontal) position parameter maintained, with $D_\omega = 1$. All latent spaces in the hierarchy have dimensionality $D_x = 2$.

In Section 6.5.1 MoCap test data was also used to investigate the performance of the H-GPLVM using a simple, well defined objective function. The score for each particle is calculated from the sum of the squared 3D Euclidean distances between a set of 15 markers on the wrists, elbows, shoulders, feet, knees, hips, head, neck and pelvis of the hypothesised skeleton and the test skeleton (this is simply Eq. 6.8 for the 3D case). The skeletons were identical in size, estimated from the MoCap data of CMU subject 35 [CMU]. The scenario is one of composite activity performed by a known subject.

In Section 6.5.3 and Section 6.5.4 a set of 2D feature tracks were obtained for a subset of these 15 joint locations for unknown subjects in monocular video sequences. These were again compared with the skeleton of CMU subject 35, this time using the sum of squared 2D Euclidean distances, $\Sigma^{\mathcal{WSL}}$ (defined in Section 6.4). To facilitate this comparison an orthographic camera projection was presumed and a single constant scaling factor was estimated by hand to give

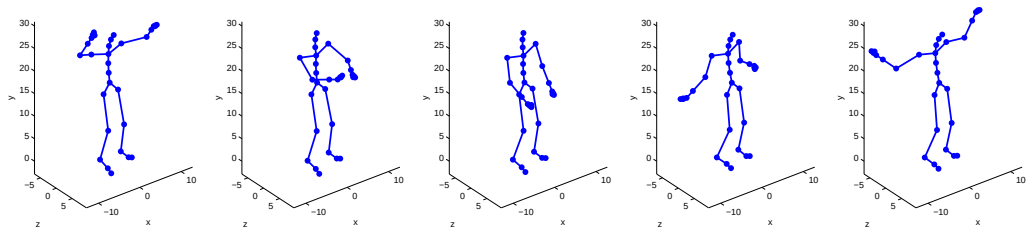
reasonable agreement in the height of the subject and skeleton. The scenarios are that of known and composite activity performed by an unknown subject.

To provide a baseline for comparison, a GPDM [WFH08] was also learned from each training data sequence and APF performed in the resulting latent spaces in an approach similar to [RRR08a]. GPDMs are an extension of the GP-LVM that incorporate dynamics, an extra GP being used to give a first order model of data dynamics in the latent space (see also Section 3.3.4.3). The smooth latent space recovered by a GPDM is suitable for exploration with particle filtering techniques where dispersion is based on a Gaussian random variable, here $\underline{n}_r^x$. All experiments used 100 particles and 10 annealing layers with a constant survival rate of $\alpha_R = ... = \alpha_0 = 0.5$. Rather than finite differencing latent data, latent noise covariance was set to a manually inflated value of $\boldsymbol{P}_0^x = 0.25\boldsymbol{I}$ to encourage pose diversity.

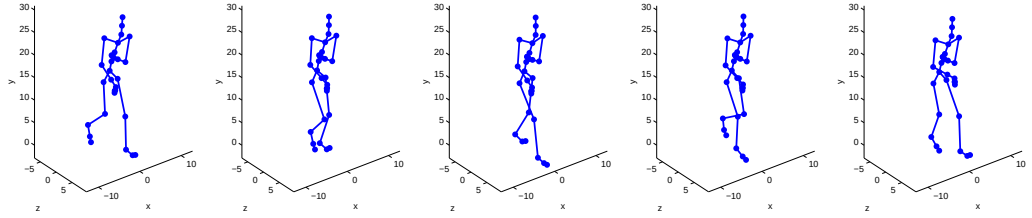
6.5.1 3D MoCap Data: *Walk*

An H-GPLVM (shown in Fig. 6.2) was trained using single 40 frame cycles of *swing arms* (CMU file 86_07.amc) and *walk with hurt stomach* (CMU file 91_26.amc) activity sequences. The model was then used to recover novel poses from a walking subject (CMU file 35_01.amc, 90 frames) using the 3D Euclidean distance objective function. The required departure from the training data is quite considerable, see Fig. 6.7. The GPDMs were unable to recover the walking poses with the particle set oscillating between the latent spaces of the two activities with constant frequency, jumping from the least worst *swing arms* pose to the least worst *walk with hurt stomach* pose, see Fig. 6.7(c).

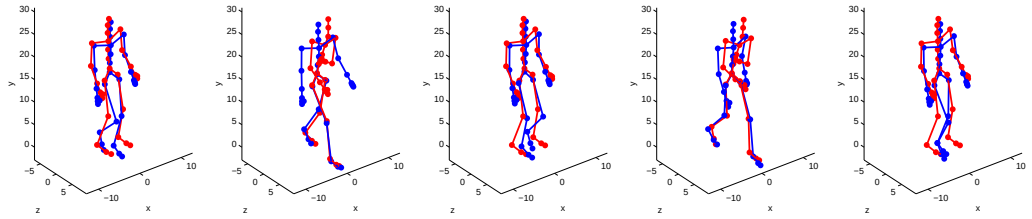
In contrast, the H-GPLVM was able to optimise limbs independently recovering good pose estimates at every frame, see Fig. 6.7(d). The required subdivision of the skeleton operates at two scales. The lower body is recovered from *walk with hurt stomach* pose data and the upper body from *swing arms*. The upper body is then further subdivided between the two arms. While the arms swing *in phase*



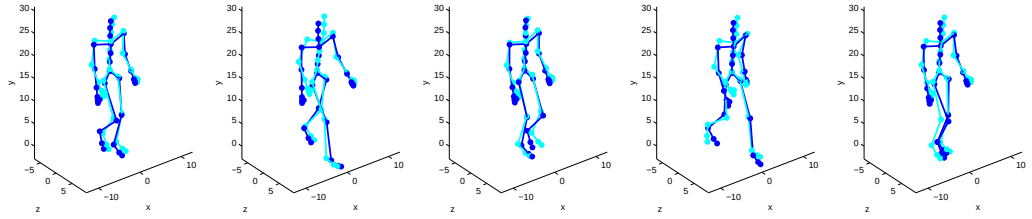
(a) Training data 1: *swing arms*.



(b) Training data 2: *walk with hurt stomach*.



(c) GPDMs: the models are unable to generalise to novel poses.



(d) H-GPLVM: back off and the addition of latent space noise allows the recovery of novel poses.

FIGURE 6.7: Training data and resulting pose estimation results for a *walk* sequence: (a-b) MoCap training data [CMU]; (c) GPDMs – particles oscillate between the best compromises in each latent space; (d) H-GPLVM – good pose recovery, note the opposing swing of the arms. Errors are plotted in Fig. 6.8.

in the training data, they are uncoupled to give the *out of phase* opposing swing seen in the walking data. Error values for pose estimation are shown in Fig. 6.8. The H-GPLVM consistently outperforms the GPDMs with average expected error across the sequence of 45.3mm versus 92.7mm for the GPDMs.

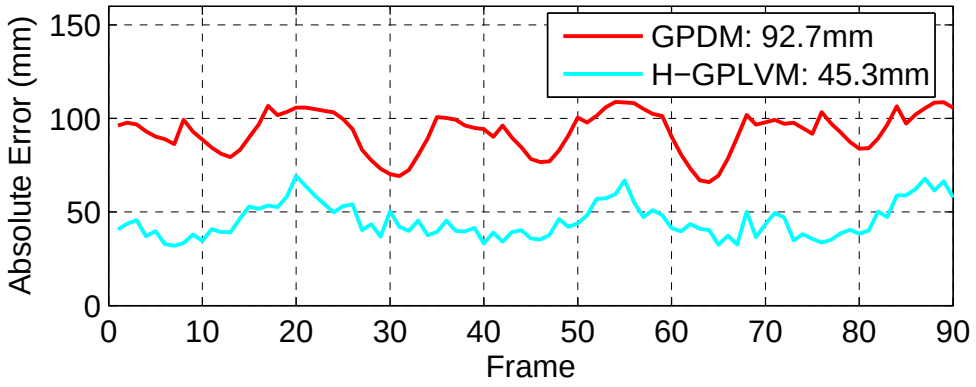


FIGURE 6.8: Expected errors for MoCap *walk* sequence in Fig. 6.7.

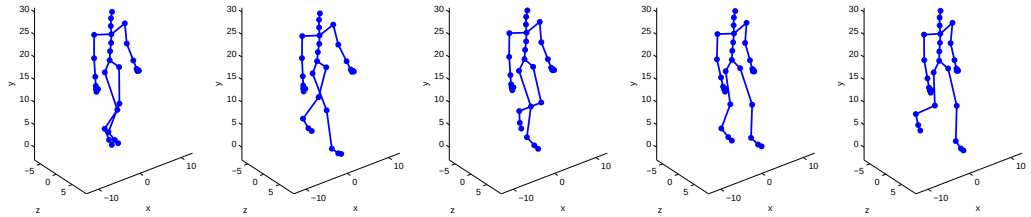
6.5.2 3D MoCap Data: *Walk whilst Waving*

To investigate performance on combined activities, an H-GPLVM was trained with single cycles of MoCap data from a person walking (see Fig. 6.9(a)) and a person standing and waving (see Fig. 6.9(b)). The model was then used to recover poses from a combined *walk whilst waving* test sequence. Searching the two GPDMs with APF recovered the best compromise at each frame, an accurate walking pose that ignored the waving hand. Test poses far exceeded the GPDMs' capacity to generalise, i.e. the variation is more than stylistic.

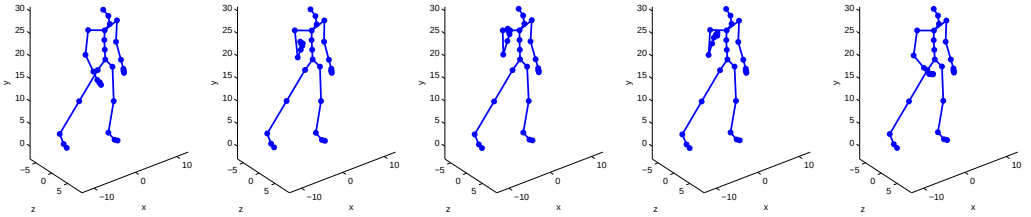
Searching the H-GPLVM resulted in a good expected pose estimate at each frame, see Fig. 6.9(c). The improvement in terms of joint location error is shown in Fig. 6.10. The H-GPLVM was able to significantly outperform the GPDMs during the wave with an average expected error across the sequence of 17.0mm versus 32.6mm.

6.5.3 2D $\mathcal{WSL}$ Data: *Walk whilst Waving*

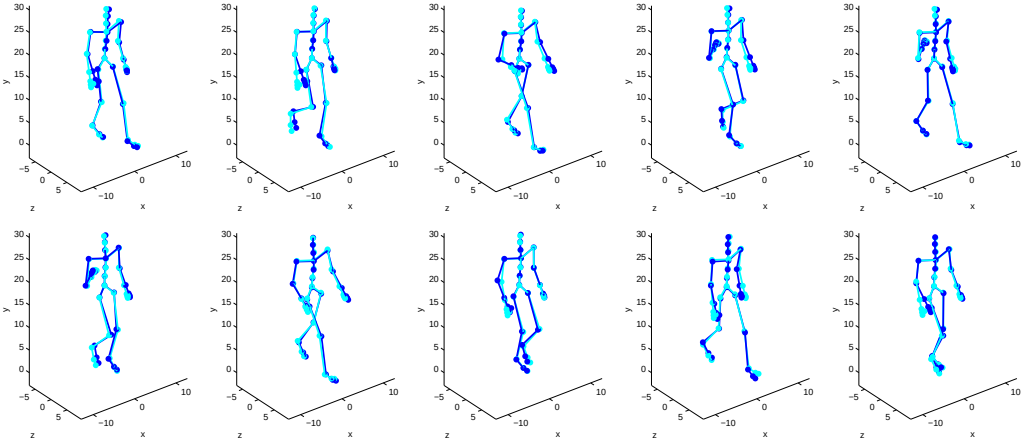
In order to test the H-GPLVM's ability to recover combined poses from 2D feature points the $\mathcal{WSL}$ tracker [JFEM03] was used to track 9 feature points on the body of a subject performing *walk whilst waving*. These comprised the hands, feet, knees, head, right shoulder and pelvis locations at each frame (see green squares in Fig. 6.11). An H-GPLVM was trained using single cycles of *slow*



(a) Training data 1: *walk* CMU file 35_01.amc



(b) Training data 2: *stand and wave* CMU file 113_27.amc



(c) H-GPLVM: "back off" permits combination of elements of each training set for successful pose estimation.

FIGURE 6.9: Training data and resulting pose estimation results for a *walk whilst waving* sequence: (a-b) MoCap training data [CMU]; (c) H-GPLVM tracking results. Errors are plotted in Fig. 6.10.

walk/stride (CMU file 08_11.amc) and *stand and wave* (CMU file 143_25.amc) activity sequences and used to recover the test poses with the 2D Euclidean distance objective function, $\Sigma^{\mathcal{WSL}}$. The tracking skeleton's pelvis was also allowed to translate horizontally to allow for a moving subject, and the extra particle parameter was dispersed with a scalar Gaussian random variable n_r^x and preserved between frames.

Results for the H-GPLVM and the GPDM baseline are shown in Fig. 6.11. The baseline recovers the best possible candidate from the two GPDMs at each frame, this is a *stand and wave* pose at every instant. The H-GPLVM is able to combine

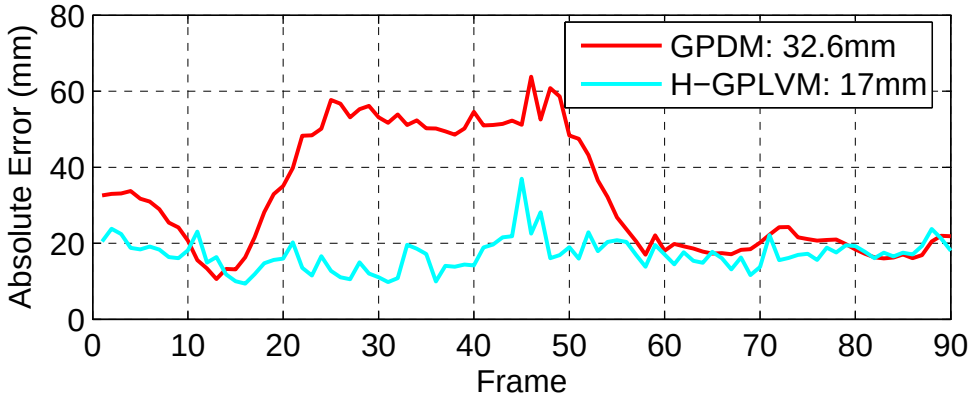


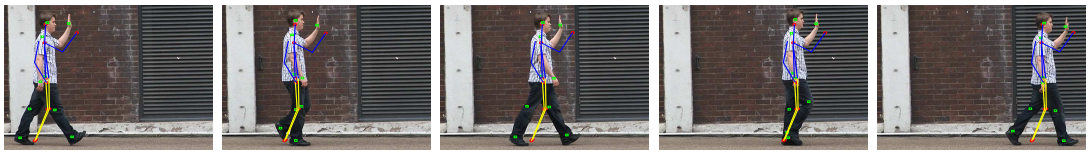
FIGURE 6.10: Expected errors for MoCap *walk whilst waving* sequence in Fig. 6.9. The GPDMs are unable to recover the combined activity poses and error is high during the wave (frames 15-60).

walking poses for the lower body and right arm with a waving pose for the left arm to give good 3D pose reconstruction throughout, see Fig. 6.11(c).

6.5.4 2D $\mathcal{WSL}$ Data: *Walk* with Occlusions

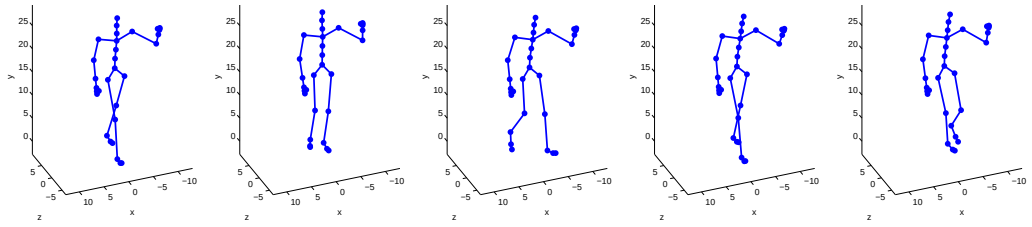
One advantage of learning global latent models of activity at the full-body scale is the ability to recover *known* poses given limited image evidence. For example, given a latent variable model learned from walking poses, walking sequences featuring occluded limbs have been reconstructed from a small set of 2D feature points [HGC+07, UFF06a, UFHF05]. In this section the H-GPLVM is shown to be "back compatible" with this kind of reconstruction of partially occluded known poses performed by an unknown subject.

An H-GPLVM was trained using a single cycle of *slow walk/stride* data (1 root node only) and used to reconstruct poses from the 2D $\mathcal{WSL}$ tracker data used by Urtasun *et al.* [UFHF05] (see green squares in Fig. 6.12). In contrast to Section 6.5.3, there is no data for the right arm and the right knee track is lost about half way through the sequence resulting in a challenging reconstruction problem. The placement of occluded limbs must be inferred from higher level correlations in the training data. In the case of a missing right arm, back off is terminated in the "upper body" node (see Section 6.3.1). While the legs are



(a) GPDMs: neither latent space contains the pose. The waving hand is recovered at the expense of the legs.

(b) H-GPLVM: back off allows the combination of training data for accurate pose recovery.



(c) H-GPLVM: inferred 3D poses from a different view point.

FIGURE 6.11: Pose estimation results using 2D $\mathcal{WSL}$ feature tracks from a monocular *walk whilst waving* sequence: (a) GPDM; (b) H-GPLVM; (c) rotated 3D view of poses. Training data is *slow walk/stride* and *stand and wave*.

independently optimised in the leaf nodes, the left arm, right arm and head are *jointly* optimised in the "upper body" node. The result is a right arm that is necessarily coordinated with the visible upper body.

Pose estimation results are shown in Fig. 6.12. To account for marker loss, a piecewise objective function was used to give no further increase in the contributions of markers separated by 30cm or more to the calculation of $\Sigma^{\mathcal{WSL}}$. Despite an absence of image evidence for the right arm, well coordinated walking poses were recovered at each frame with the occluded right arm oscillating out of phase with the visible left arm. This long-range skeletal correlation is a benefit of the hierarchical approach; in a "flat" part-based model, an occluded limb would be free to randomly flail.

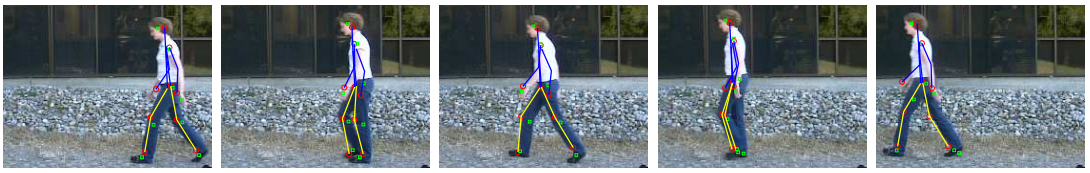


FIGURE 6.12: Pose estimation results using 2D $\mathcal{WSL}$ feature tracks from a monocular *walk* sequence [SBS02] using H-GPLVM. Note the position of occluded right arm is inferred from the visible upper body.

6.6 Discussion and Conclusions

This chapter has outlined a middle ground between searching low-dimensional global pose models and searching in the original high-dimensional state space. This is achieved by descending through a hierarchical part based model: the H-GPLVM. Breaking correlations between local part-based models permits the recovery of novel composite activity poses. However, it is the retention of *long range* correlations in the higher levels of the hierarchy that permits known activity tracking through occlusion.

Just as in Chapter 5 the benefit of encountering known activity (e.g. see Section 6.5.4) could be reflected in terms of computational cost. This is by terminating back off as soon as one (or more) particles attain some minimum error threshold. For the $\mathcal{WSL}$ experiments presented here this additional step is trivial (checking for a lower bound on $\Sigma^{\mathcal{WSL}}$ calculations) and may also be possible if using bottom-up tracker output as input (see also Section 7.3.2.3). However, this is unlikely to be possible if using more general objective functions such as those presented in Section 4.3.

At its top level the H-GPLVM is akin to a set of GPDMs, one for each activity. But by "backing off" to benefit from progressively greater independence between body parts, and by making increasingly discerning comparisons with image evidence, the stochastic search algorithm presented is able to recover novel pose configurations. H-GPLVMs can be used to recover poses that are beyond the scope of other widely used global latent variable models such as the GPDM.

Chapter 7

Conclusions

A number have methods have been proposed for the recovery of known and unknown human motions. Here the contributions made in this thesis are reviewed, their relative merits discussed and future work proposed.

7.1 Known Activity

In Chapter 4 a method was described for known activity tracking – that is, tracking of activities for which training data is available. The approach is discussed further below and the need for future work highlighted.

7.1.1 Contributions

HMM-APF entails the use of an HMM to disperse particles across a latent pose space as part of an annealed particle filtering framework. PCA is used for dimensionality reduction of MoCap training data and an associated dynamical model is recovered by learning an HMM from the resulting latent variables. The method has been found to be capable of recovering activity from less than three cameras; specifically, from monocular, narrow-baseline stereo and wide-baseline stereo observations. Furthermore, these results are achieved using only a small allocation of particles.

This performance contrasts with high-dimensional standard APF which fails when tracking human motions from three or fewer cameras using much larger particle allocations. This has been confirmed by a number of quantitative investigations into particle filtering and APF [BSB05, SBB10] and related variants [BEB08]. It is also the finding of this thesis, see for example Figures 5.7 and 5.8 where standard APF has been combined with the symmetric objective function used in this work (see also Section 4.3.3) but is still unable to recover *walk* and *jog* activities reliably.

The observed improvements in performance are in line with other work on latent models for known activity tracking, see for example Section 2.3.2.2. The drawback that unites all these various approaches is an inability to generalise to (even modestly) novel activities. It is this limitation that motivates the work presented in Chapters 5 and 6.

7.1.2 Future Work

7.1.2.1 Quantitative Evaluation of Range Data Tracking

Narrow-baseline stereo data (see also Section 3.5.2) presents a particularly interesting observation format for the application of known activity tracking (see also Section 4.4.1). This is primarily because it has the potential to remove the need for background subtraction and with it the requirement that camera position, background appearance and lighting conditions do not change. Section 4.4.1.2 demonstrated qualitatively satisfactory tracking of a walking subject from a *moving* stereo camera, without the need for background subtraction.

Stereo cameras are becoming relatively cheap and simple to calibrate e.g. [I2I] but applications to human motion tracking are still relatively rare (see also Section 3.5.2.2). It is likely that the difficulty of quantifying tracking accuracy and the lack of any shared datasets within the community represents a barrier to progress. In the cases of monocular and multi-camera tracking, freely available datasets containing video with synchronised motion capture ground truth – e.g.

HumanEva [SB06a] – have contributed to the advancement of the state of the art by simplifying the necessary evaluation and comparison tasks. The production of a similar resource featuring range images of human movement with synchronised MoCap ground truth is experimentally challenging, but could play a similarly important role in the recovery of human movement from stereo.

7.1.2.2 Temporal Diversity in Known Activity

The use of dynamic T_0 where known activity dynamics are captured by an HMM deserves further investigation. One barrier to this is the absence of a clear analogue for the inflation of other forms of dynamical model. The method described in Section 4.2.1.2 is therefore overlooked for use in MAM-APF (see also Chapter 5). However, when tracking known activity exclusively, the ability to consider the next *spatially* significant change in pose rather than only the next *temporal* change is likely to prove useful. The *walk* and *jog* activities processed in Chapter 4 involve reasonably constant motion[1] but activities that result in more markedly self-referential states are perhaps a more interesting candidate for investigation.

One example is given by sparring activities where a relatively static *guard* pose is occasionally interrupted by explosive bursts of motion, such as a *punch* being thrown. Here training data results in a highly self-referential *guard* state s_i that, during inference, all but a small fraction of particles will fail to escape; $A_{ii} \approx 1$. Tracking from anything other than rich observation data is therefore challenging. During such activities it may be beneficial to insist on exploring the next *spatially* distinct pose with particles – that is, an early *punch* state with the arm starting to extend. This can be achieved using the transition temperature, ρ_T introduced in Section 4.2.1.2.

[1] The rate of "flow" between hidden states is not constant, however. For example, there is a momentary lull in the walking gait when both feet are in contact with the floor.

7.2 Known and Unknown Activity

In generative tracking, the use of high-dimensional activity models has previously allowed the recovery of freeform human motions without limitations on activity class. Drawbacks have included the need for sufficiently rich observations from multiple cameras, and a high and fixed computational cost during tracking. As an alternative, many approaches (including HMM-APF, presented in Chapter 4) have adopted a low-dimensional activity model to recover certain classes of activity from fewer cameras and at reduced computational cost. The drawback being that training data must be available for every activity that is to be tracked. To address these limitations Chapter 5 introduced a generative tracking approach that gives equal consideration to the predictions of both low- and high-dimensional activity models at each frame.

7.2.1 Contributions

MAM-APF combines a number of different activity models within the APF framework. Each activity model is aimed at solving a particular class of tracking problem. For example, a novel method is introduced for the recovery of activity transitions by using particles to explore "transition lines" between different manifolds in a joint-activity latent pose space. The estimation tasks associated with each activity model are quite different in terms of difficulty, and differently sized particle quotas are assigned to them to reflect this. An equal prior over activity models is efficiently ensured using a particle stacking technique.

In a simple (single layer) particle filter the multiple activity model technique can bring no *computational* advantage, but by drawing a variable number of samples based on the emerging picture of activity model membership across a number of annealing layers significant gains in efficiency can be made. The final distribution of particles between activity models can also be used as a classifier for each observation. MAM-APF provides good segmentations of sequences featuring multiple known and unknown activities with transitions. The algorithm is an attempt to

combine the best of two generative tracking approaches: faster recovery of known activity with few particles where possible, but the flexibility to work for longer with more particles to recover unknown activities where necessary.

7.2.2 Future Work

7.2.2.1 Many Known Activities

In Chapter 5 a joint-activity pose space was adopted to recover known activity transitions, and a joint-subject pose space to generalise to unknown subjects. The *HumanEva-II* data set has permitted quantitative investigation, but it would be interesting to extend the approach to support larger numbers of known activities in the future. Where more activities are used to create a joint pose space, an HMM-guided particle-based approach is well placed to explore the resulting activity manifolds.

Where activities contain poses that are close in latent space (and therefore in ambient space) probabilistic classification between nearby HMM states can be used to select the correct HMM for propagation. Achieving classification based only on a single pose (first order dynamics) may be challenging, however. For example, there may be genuine "junctions" in the latent space due to two or more activities sharing a similar component pose. The investigation presented in Appendix D finds that even the consideration of long state histories does not always guarantee disambiguation of a pose between activity classes. This motivates the multiple hypothesis particle-based approach to estimation: an ensemble of pose hypotheses drawn from noisy (Gaussian) state observation densities will naturally divide between competing (nearby) HMM states for subsequent propagation. For example, in Section 5.5.2 the proximity of the three subjects' latent data (see also Fig 5.4, right) leads particles to flow constantly between HMMs during tracking. Where HMMs represent substantially different but partially overlapping activities in a joint space, the correct HMM will assume complete control of tracking

(and the known activity particle quota) at such time as its future pose hypotheses begin to diverge from those of the others.

The dimensionality of a joint-activity space must grow with the number of activities modelled. Alternatively, a set of individual low-dimensional latent pose spaces could be used, one for each known activity. Here the use of transition lines is no longer possible. This is the approach taken in earlier work [DLC08a], but forcing an equal number of particles into each space means that computational cost increases with the number of activities. In contrast, results in Chapter 5 have shown it is not necessary to saturate *every* HMM with particles, only to select a single parent HMM state for each particle in the quota.

An infinity of points in the high dimensional pose space describing unrelated unknown poses do in fact project to latent coordinates close to known activity training data. This is because pose variations are concentrated in the orthogonal complement to the PCA subspace [MP97]. This is important in the context of MAM-APF where particles can flow between ambient and latent space. If multiple latent pose spaces are used, it may be insufficient simply to find the single most likely parent state in order to determine which known activity model contains the "closest" pose. A low cost solution is to reconstruct an unknown particle's pose from its latent coordinate in each space, and select the activity that gives the lowest projection-reconstruction error (see also Section 5.5.3).

7.2.2.2 Activity Class Transitions

In Chapter 5 an equal prior is placed over all activity models at all frames, anticipating the commencement of *any* class of activity with equal probability. This can be interpreted as a "flat" Markovian activity model transition matrix, e.g. see those used for dynamical model transitions by Isard and Blake [IB98c]. While it is prudent to continually cater for the possibility that known activity will start to transform into unknown activity, the reverse does not always hold. Where the projection-reconstruction error (see also Section 5.5.3) is consistently high given the latent pose space, it is natural to ask whether the projection is

appropriate at all. That is, if the unknown activity model (correctly) permits particles to move through the ambient pose space until they are "far" from all known activity poses, it may no longer be appropriate to force equal particle quotas into the known activity model.

The projection-reconstruction error could be used to dynamically adjust the prior on activity model transitions. In practice this would mean making adjustments to the probability of unknown-to-known activity class transitions based on how accurately the latent pose space is able to reconstruct the last expected pose. The potential computational saving is relatively modest – $2B_{\text{min}}$ unnecessary objective function evaluations at the first layer during unknown activity (around 5% of computation time per frame) – but the practice may also help to guard against false transitions.

7.3 Composite Activity

Appeals to low-dimensional models of pose are not unreasonable. It is true that during every day activity the range of typical human movements is surprisingly limited, especially given the range of *possible* movements e.g. see the CMU Mo-Cap database [CMU]. Although it may not be viable to learn low-dimensional activity models for *all* typical movements, an interesting alternative is to learn a compact subset of activities with which remaining activities "overlap". This is the approach taken in Chapter 6 where novel poses are recovered by gradually breaking down known activities into smaller part-based representations which are recombined using a crossover operator to create new poses.

7.3.1 Contributions

Section 6.5 demonstrates the recovery of unknown activity through the recombination of known activity via a hierarchical part-based representation of pose. The H-GPLVM provides a quite unique model of pose – a hierarchy of *latent*

rather than observable variables – but how best to conduct inference is not clear. Faced with complete global coordination of body parts at the root (useful for dealing with occlusion) and no coordination whatsoever at the leaves (useful for recovering novel poses), it is not obvious how to proceed. Chapter 6 implements the suggestion of Lawrence and Moore [LM07]: "backing off". This means moving from the top to the bottom of the hierarchy in stages, applying the models at each level *independently*. The result is a state vector that gradually increases its dimensionality *en route* from a single global latent pose space to the ambient pose space (see also Eq. 6.4).

The philosophy of annealing is to support a wide range of diverse pose hypotheses initially before gradually concentrating in on a globally optimal solution through increasingly discerning comparisons with image evidence (see also Fig. 3.2). This is ideal for the exploration of the H-GPLVM where it is important not to be drawn into a local optimum too quickly. For example, if the objective function scores achieved by full-body poses at the top level of the hierarchy are not cooled when an unknown pose is observed then resampling will overlook many *partially* accurate poses. Committing to the best full-body pose solutions reduces diversity and may preclude recovering the correct pose at lower levels. The ability to introduce and to support such diversity is critical to the success of the approach.

7.3.2 Future Work

7.3.2.1 Investigating Compositionality

Although Section 6.5 contains some interesting examples, the extent to which human activities more generally are composite is not investigated. Given a large enough database of examples this question can be addressed experimentally. The CMU MoCap database [CMU] is a suitable candidate and a quantitative analysis of joint angle data could permit the recovery of a set of "basis activities" that have maximum overlap with other movements. It would be interesting to know

the required size of this set and the computational cost of training and exploring an associated H-GPLVM in order to recover a range of composite activities.

7.3.2.2 Tracking Mode

The approach presented in Chapter 6 might also be extended to tracking scenarios. In contrast to pose estimation, the aim here is to recover small inter-frame changes in pose with a more conservative dynamical model. However, there remains the challenge of exploiting the various scales of correlation that are captured by the different levels of the H-GPLVM. One possibility is to apply the original technique to provide an initialisation by pose estimation at the first frame and then to cluster the resulting coordinates in each leaf node based on their nearest latent training variable. Where a particle describes points with common (or nearby) cluster indices in two or more sibling nodes, these values can be combined by *ascending* the hierarchy to the equivalent cluster in the parent node. Performing this step ensures that the poses at $t-1$ are retained as the starting point for particle dispersion at t, but also identifies long range correlations, enabling the application of dynamics at the appropriate level of the hierarchy.

7.3.2.3 Bottom-up Output as Top-down Input

As mentioned in Section 2.4.2 a potential focus for future research is to replace the 2D $\mathcal{WSL}$ tracker results used here and in [UFHF05, UFF06a] with the 2D joint location estimates of a bottom-up tracker e.g. [RFZ07]. This would remove the need to hand initialise (defining $\mathcal{WSL}$ ellipses in the first frame) and the work presented in Chapter 6 is a potential candidate for inferring occluded limb positions from long-range correlations in training data. A difficulty is that the input does not account for "sidedness" – that is, (unless perhaps clothing is asymmetrically coloured [RFZ07]) there is no notion of right and left for a given limb. Addressing how best to support and resolve this ambiguity would be an interesting future topic for investigation.

7.4 Concluding Remarks

This thesis has presented a collection of work aimed at bridging a gap between low-dimensional and high-dimensional generative tracking approaches. This has included: (i) defining novel low-dimensional activity models for *known activity* tracking; (ii) combining these with high-dimensional activity models for *unknown activity* tracking; (iii) gradually removing dependencies between partitioned low-dimensional pose models to recover *composite activity*. Each of these contributions has been tested within the estimation framework of the annealed particle filter [DBR00] and various different objective functions have been proposed for tracking from different forms of observation: monocular, narrow-baseline stereo, and wide-baseline stereo. These techniques have permitted the dynamic reduction of particle numbers during known activity, and the ability to track known poses through occlusion. Where observation data is sufficiently rich they have additionally permitted the recovery of composite poses by activity combination, and unknown activity poses by dynamically increasing the size of the particle set.

Appendix A

Bayesian Filtering

The Bayesian filtering equation is much cited but rarely derived in the tracking literature. For completeness it is included here; the derivation below is closely based upon that given by Sigal [Sig08] and elaborates each step for clarity.

The system state $\underline{s}_t$ at every discrete time instant t is exposed to some sensor to produce the corresponding observation $\underline{z}_t$. In the context of this thesis, the system state is a set of joint angles, the sensors are cameras and the observations are digital images. Observations are presumed to be independent of both each other and of the past and future state of the underlying dynamical process. The recursive Bayesian filtering equation can be derived by manipulation of the joint distribution $p(\underline{s}_0, \underline{s}_1, ..., \underline{s}_t | \underline{z}_0, \underline{z}_1, ..., \underline{z}_t)$ using conditional probability rules (Section A.1) and Bayes' rule (Eq. A.2).

A.1 Marginalisation

For two continuous random variables, X given Y, the marginal probability density function can be written as $p_X(x)$. This is

$$p_X(x) = \int_y p_{X,Y}(x, y)\, dy = \int_y p_{X|Y}(x|y) p_Y(y)\, dy \tag{A.1}$$

where $p_{X,Y}(x, y)$ gives the joint distribution of X and Y, while $p_{X|Y}(x|y)$ gives the conditional distribution for X given Y. The second expression comes from a more general rule about conditional probabilities:

$$p_{X|Y}(x|y) = \frac{p_{X,Y}(x, y)}{p_Y(y)}, \tag{A.2}$$

for $p_Y(y) \neq 0$.

A.2 Bayes' Rule

Bayes' rule is given by,

$$p_{X|Z}(x|z) \propto p_{Z|X}(z|x)p_X(x). \tag{A.3}$$

A.3 The Filtering Equation

The joint distribution can be integrated to marginalise past system states,

$$p(\underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t) = \int_{\underline{s}_0} \int_{\underline{s}_1} ... \int_{\underline{s}_{t-1}} p(\underline{s}_0, \underline{s}_1, ..., \underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t)\, d\underline{s}_0 d\underline{s}_1 ... d\underline{s}_{t-1}. \tag{A.4}$$

Then by making the first order Markov assumption that $\underline{s}_t$ depends only on $\underline{s}_{t-1}$,

$$p(\underline{s}_t|\underline{s}_0, \underline{s}_1, ..., \underline{s}_{t-1}) = p(\underline{s}_t|\underline{s}_{t-1}) \tag{A.5}$$

Eq. A.4 can be rewritten as

$$p(\underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t) = \int_{\underline{s}_{t-1}} p(\underline{s}_t, \underline{s}_{t-1}|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t)\, d\underline{s}_{t-1}. \tag{A.6}$$

Rewriting using Bayes' Rule (Eq. A.3) gives

$$p(\underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t) = \int_{\underline{s}_{t-1}} \frac{p(\underline{z}_0, \underline{z}_1, ..., \underline{z}_t|\underline{s}_t, \underline{s}_{t-1})p(\underline{s}_t, \underline{s}_{t-1})}{p(\underline{z}_0, \underline{z}_1, ..., \underline{z}_t)}\, d\underline{s}_{t-1}. \tag{A.7}$$

Where by assuming that the current observation is conditionally independent of the past observations given $\underline{s}_t$,

$$p(\underline{z}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_{t-1}, \underline{s}_0, \underline{s}_1, ..., \underline{s}_t) = p(\underline{z}_t|\underline{s}_t) \tag{A.8}$$

this can be rewritten as

$$p(\underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t) = \int_{\underline{s}_{t-1}} \frac{p(\underline{z}_t|\underline{s}_t, \cancel{\underline{s}_{t-1}})p(\underline{z}_0, \underline{z}_1, ..., \underline{z}_{t-1}|\cancel{\underline{s}_t}, \underline{s}_{t-1})p(\underline{s}_t, \underline{s}_{t-1})}{p(\underline{z}_t)p(\underline{z}_0, \underline{z}_1, ..., \underline{z}_{t-1})} \, d\underline{s}_{t-1}. \tag{A.9}$$

Where the simplifications are possible because the current observation $\underline{z}_t$ is independent of all past and future system states (Eq. A.8). Next, restating the final term in the numerator in terms of a conditional and a prior (Eq. A.1) gives

$$p(\underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t) = \int_{\underline{s}_{t-1}} \frac{p(\underline{z}_t|\underline{s}_t)p(\underline{z}_0, \underline{z}_1, ..., \underline{z}_{t-1}|\underline{s}_{t-1})p(\underline{s}_t|\underline{s}_{t-1})p(\underline{s}_{t-1})}{p(\underline{z}_t)p(\underline{z}_0, \underline{z}_1, ..., \underline{z}_{t-1})} \, d\underline{s}_{t-1}. \tag{A.10}$$

Rearranging to recognise the right hand side of Bayes' Rule (Eq. A.3)

$$p(\underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t) = \int_{\underline{s}_{t-1}} \frac{p(\underline{z}_t|\underline{s}_t)}{p(\underline{z}_t)} p(\underline{s}_t|\underline{s}_{t-1}) \frac{p(\underline{z}_0, \underline{z}_1, ..., \underline{z}_{t-1}|\underline{s}_{t-1})p(\underline{s}_{t-1})}{p(\underline{z}_0, \underline{z}_1, ..., \underline{z}_{t-1})} \, d\underline{s}_{t-1}, \tag{A.11}$$

recovers

$$p(\underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t) = \int_{\underline{s}_{t-1}} \frac{p(\underline{z}_t|\underline{s}_t)}{p(\underline{z}_t)} p(\underline{s}_t|\underline{s}_{t-1}) p(\underline{s}_{t-1}|\underline{z}_0, \underline{z}_1, ..., \underline{z}_{t-1}) \, d\underline{s}_{t-1}. \tag{A.12}$$

As $p(\underline{z}_t)$ is a constant and $p(\underline{z}_t|\underline{s}_t)$ independent of $\underline{s}_{t-1}$, one has that

$$\underbrace{p(\underline{s}_t|\underline{z}_0, \underline{z}_1, ..., \underline{z}_t)}_{\text{Posterior at time } t} = \frac{1}{C} \underbrace{p(\underline{z}_t|\underline{s}_t)}_{\text{Likelihood}} \int_{\underline{s}_{t-1}} \underbrace{p(\underline{s}_t|\underline{s}_{t-1})}_{\text{Dynamical model}} \underbrace{p(\underline{s}_{t-1}|\underline{z}_0, \underline{z}_1, ..., \underline{z}_{t-1})}_{\text{Posterior at time } t-1} \, d\underline{s}_{t-1}. \tag{A.13}$$

Appendix B

Probabilistic Interpretations of PCA

In the following subsections some of the key steps in the derivations of PPCA [TB99] and the GP-LVM [Law05] are reproduced. The reader may first wish to familiarise themselves with the results for the marginal probability density function for a continuous random variable in Section A.1.

B.1 Probabilistic PCA

Following Tipping and Bishop [TB99], a matrix of low-dimensional latent variables,

$$\boldsymbol{X} = [\underline{x}_1, ..., \underline{x}_N]^\top \tag{B.1}$$

is related to a matrix of concatenated high-dimensional pose vectors,

$$\boldsymbol{Y} = [\underline{y}_1, ..., \underline{y}_N]^\top, \tag{B.2}$$

through a set of linear mapping parameters corrupted by noise,

$$\underline{y}_n = \boldsymbol{W}\underline{x}_n + \underline{\eta}_n. \tag{B.3}$$

The mapping is given by $\boldsymbol{W} \in \Re^{D_y \times D_x}$ with D_y the dimension of the data space, D_x the dimension of the latent space and $\underline{\eta}_n$ a vector of noise terms. In PPCA the noise is taken to be Gaussian distributed,

$$p(\underline{\eta}_n|\beta) = N(\underline{\eta}_n|\underline{0}, \beta^{-1}\boldsymbol{I}), \tag{B.4}$$

with a mean of zero and a spherical covariance given by $\beta^{-1}\boldsymbol{I}$.

The conditional probability of the nth original pose datum given its corresponding latent datum and mapping can be written as

$$p(\underline{y}_n|\underline{x}_n, \boldsymbol{W}, \beta) = N(\underline{y}_n|\boldsymbol{W}\underline{x}_n, \beta^{-1}\boldsymbol{I}), \tag{B.5}$$

where the mean vector $\underline{\mu} = \boldsymbol{W}\underline{x}_n$ (from Eq. B.3) and the covariance matrix $\boldsymbol{\Sigma} = \beta^{-1}\boldsymbol{I}$ from (Eq. B.4) have been used to parameterise a Gaussian distribution. Then assuming independence across data points (and just multiplying probabilities together),

$$p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{W}, \beta) = \prod_{n=1}^{N} N(\underline{y}_n|\boldsymbol{W}\underline{x}_n, \beta^{-1}\boldsymbol{I}) \tag{B.6}$$

This result can then be manipulated to give both the PPCA result [TB99] and the GP-LVM result [Law05].

In PPCA the latent variables $\boldsymbol{X}$ are marginalised as nuisance parameters, and the mapping parameters $\boldsymbol{W}$ optimised by likelihood maximisation of $p(\boldsymbol{Y}|\boldsymbol{W})$. In this case Eq. B.6 is multiplied by a prior on $\boldsymbol{X}$ and integrated with respect to $\boldsymbol{X}$. The form of the prior is chosen to be Gaussian with zero mean and unit covariance by convention,

$$p(\boldsymbol{X}) = \prod_{n=1}^{N} p(\underline{x}_n) = \prod_{n=1}^{N} N(\underline{x}_n|\underline{0}, \boldsymbol{I}). \tag{B.7}$$

This leads to

$$\begin{aligned} p(\boldsymbol{Y}|\boldsymbol{W},\beta) &= \prod_{n=1}^{N} \int N(\underline{y}_n|\boldsymbol{W}\underline{x}_n, \beta^{-1}\boldsymbol{I}) N(\underline{x}_n|\underline{0},\boldsymbol{I})\, d\underline{x}_n \\ &= \prod_{n=1}^{N} N(\underline{y}_n|\underline{0},\boldsymbol{C}) \end{aligned} \tag{B.8}$$

where $\boldsymbol{X}$ has gone from the conditional and the final result can be recognised as a product of zero mean Gaussians with $\boldsymbol{C} = \boldsymbol{W}\boldsymbol{W}^\top + \beta^{-1}\boldsymbol{I}$. Tipping and Bishop give a proof that Eq. B.8 is maximised when $\boldsymbol{W}$ spans the principal sub-space of the ambient data.

B.2 Dual Probabilistic PCA

In contrast, derivation of the GP-LVM [Law05] begins with a *dual* probabilistic interpretation of PCA in which the mapping parameters $\boldsymbol{W}$ are marginalised (also using a Gaussian prior), and the latent variables $\boldsymbol{X}$ optimised by likelihood maximisation of $p(\boldsymbol{Y}|\boldsymbol{X})$. In this case Eq. B.6 is multiplied by a prior on $\boldsymbol{W}$ and integrated with respect to $\boldsymbol{W}$,

$$p(\boldsymbol{Y}|\boldsymbol{X},\beta) = \prod_{n=1}^{N} \int N(\underline{y}_n|\boldsymbol{W}\underline{x}_n, \beta^{-1}\boldsymbol{I}) p(\boldsymbol{W})\, d\boldsymbol{W}. \tag{B.9}$$

If the form of the prior is again chosen to be Gaussian and with zero mean and unit covariance,

$$p(\boldsymbol{W}) = \prod_{i=1}^{D_y} N(\underline{w}_i|\underline{0},\boldsymbol{I}) \tag{B.10}$$

where $\underline{w}_i$ is the ith row of the matrix $\boldsymbol{W}$ then the likelihood can be written as,

$$\begin{aligned} p(\boldsymbol{Y}|\boldsymbol{X},\beta) &= \prod_{i=1}^{D_y} \frac{1}{(2\pi)^{\frac{N}{2}}|\boldsymbol{K}|^{\frac{N}{2}}} \exp\left(-\frac{1}{2}\underline{y}_{:,i}^\top \boldsymbol{K}^{-1}\underline{y}_{:,i}\right) \\ &= \prod_{i=1}^{D_y} N(\underline{y}_{:,i}|\underline{0},\boldsymbol{K}). \end{aligned} \tag{B.11}$$

Where $\boldsymbol{W}$ has gone from the conditional, $\underline{y}_{:,i}$ is the ith column of $\boldsymbol{Y}$ and $\boldsymbol{K} = \boldsymbol{X}\boldsymbol{X}^\top + \beta^{-1}\boldsymbol{I}$. This result is termed dual probabilistic PCA (DPPCA).

Appendix C

HMM Training and Classification

This appendix contains equations for learning and classification using HMMs with single multivariate Gaussian observation functions. Expectation maximisation derivations are based on the paper by Rabiner [Rab89] and the particularities of the single Gaussian case follow the results given by Wilson and Bobick [WB01].

C.1 Training: the Baum-Welch Algorithm

The Baum-Welch algorithm requires calculation of the forward and backward variables for the data set $X = \{\underline{x}_1, ..., \underline{x}_M\}$. The forward variable for a state s_i at time m is the total probability of all paths through the model that emit the training data up to time m, $\{\underline{x}_1, ..., \underline{x}_m\}$ and finish in state s_i

$$\alpha_{m,i} = p_i(\underline{x}_m) \sum_{j=1}^{N} \alpha_{m-1,j} A_{ji} \tag{C.1}$$

where $\alpha_{1,i}$ is calculated using the distribution $\underline{a}$ i.e. $a_i \times p_i(\underline{x}_1)$. Similarly, the backward variable for a state s_i at time m is the total probability of all paths from state s_i that emit the rest of the training data $\{\underline{x}_{m+1}, ..., \underline{x}_M\}$

$$\beta_{m,i} = \sum_{j=1}^{N} \beta_{m+1,j} p_j(\underline{x}_{m+1}) A_{ij} \tag{C.2}$$

where $\beta_{M,i} = 1$. At any time m, the value $\alpha_{m,i}\beta_{m,i}$ gives the total probability of all paths through the model that produce the data X and pass through state s_i at time m. Furthermore, $\sum_{i=1}^{N} \alpha_{m,i}\beta_{m,i}$ is constant for all m and gives the probability of the sequence X given λ, or $p(X|\lambda)$. These results can be used to calculate the probability that the model was in state s_i when feature vector $\underline{x}_m$ was observed, given all the data

$$\gamma_{m,i} = \alpha_{m,i}\beta_{m,i} \Bigg/ \sum_{i=1}^{N} \alpha_{m,i}\beta_{m,i} \tag{C.3}$$

with which one can estimate the parameters of the Gaussian emission function $p(\underline{x})$ associated with each state s_i

$$\underline{\mu}_i = \sum_{m=1}^{M} \gamma_{m,i}\underline{x}_m \Bigg/ \sum_{m=1}^{M} \gamma_{m,i} \tag{C.4}$$

$$\Sigma_i = \sum_{m=1}^{M} \gamma_{m,i}(\underline{x}_m - \underline{\mu}_i)(\underline{x}_m - \underline{\mu}_i)^T \Bigg/ \sum_{m=1}^{M} \gamma_{m,i} \tag{C.5}$$

these are the first two maximisation steps.

In order to reestimate the matrix $\boldsymbol{A}$, it is necessary to consider the probability that a transition from state s_i to state s_j occurred between timesteps $m-1$ and m

$$\xi_{m,ij} = p(q_m = s_j, q_{m-1} = s_i|X, \lambda) = \frac{\alpha_{m-1,i}A_{ij}p_j(\underline{x}_m)\beta_{m+1,j}}{p(X|\lambda)} \tag{C.6}$$

where q_m is the active hidden state at time m. This is the total probability of all paths through the model which emit $\left\{\underline{x}_1, ..., \underline{x}_{m-1}\right\}$ and pass through state s_i at $m-1$ (given by α_{m-1}), multiplied by the transition-emission pair s_i transitions to s_j, s_j emits $\underline{x}_m$, multiplied by the total probability of all paths from state s_j that emit the remainder of the training data $\left\{\underline{x}_{m+1}, ..., \underline{x}_M\right\}$ (given by $\beta_{m,j}$), as a fraction of all paths through the model that emit the data.

Summing over the total number of state transitions gives the expected number of transitions from s_i to s_j

$$E_{ij} = \sum_{m=2}^{M} \xi_{m,ij} \tag{C.7}$$

as the expectation step. The final maximisation step is then

$$A_{ij} = E_{ij} \Big/ \sum_{j=1}^{N} E_{ij} \tag{C.8}$$

This process can then be iterated, with Eqs. C.4, C.5 and C.8 providing the new estimate for λ, until some convergence criteria is met. The elements of $\underline{a}$ may also be reestimated using $\gamma_{1,i}$, although this is not done in this work.

C.2 Classification

The definition of the forward variable α can be used to calculate the likelihood of a sequence of feature vectors given a particular set of model parameters. For a set of test data $X' = \{\underline{x}'_1, ..., \underline{x}'_M\}$ and model $\lambda = \{S, \boldsymbol{A}, \underline{a}, p_i(\underline{x})\}$,

$$p(X'|\lambda) = \sum_{i=1}^{N} \alpha_{M,i}. \tag{C.9}$$

Therefore, if an HMM is trained for each activity of interest, one can evaluate the likelihood that unseen test data was produced by each of the models and classify data as belonging to the model most likely to have emitted it.

Appendix D

HMMs for MoCap Data Classification

This appendix presents an investigation into the use of HMMs for modelling dynamics in low-dimensional embeddings of human activity data. HMMs provide a natural framework for modelling noisy observations of a stochastic process. A good dynamical model is important for efficient particle dispersion and HMMs are an interesting candidate for a number of reasons. First, hidden state observation densities can be used to define "valid" subregions of the embedding space, preventing the sampling of "illegal" poses. Second, movement between hidden states via the transition matrix provides reliable activity synthesis. *Finally, probabilistic* classification *of poses is possible by the evaluation of each state's observation density. This last characteristic of HMMs is of particular interest where a single subspace is used to model jointly a number of separate activities.*

D.1 Introduction

Particle-based inference requires a model of temporal dynamics for particle dispersion. As discussed in Section 3.4.1 this model may be very simple, for example a Gaussian random variable. However, more sophisticated models have the potential to improve tracking performance by anticipating future poses and propagating particles to pertinent regions of the pose space in a "smart sampling"

approach. First order Markov chains [HH98], second order autoregressive processes [AT04b] and higher-order variable length Markov models [CGH05] have all previously been used for this purpose (see Section 2.3.2.3 for further discussion). Each has been shown to perform well. Hidden Markov models (HMMs), however, have another potentially useful feature: the ability to *classify* poses.

Pose classification is useful where a single pose space is used to model a number of different activities. In this case it is desirable to classify a pose for two reasons. First, in the context of estimation, classification allows propagation of each particle by the correct activity HMM. Second, classification of the resulting pose allows a tracker additionally to label human activities. In light of these factors and in anticipation of the usefulness of a joint activity pose space, the ability of HMMs to classify human activities is further investigated in this appendix.

If activity classification is the *only* objective, there are a number of methods that may be preferred to the HMM. Section 4 of [WHT03] gives a comprehensive review of the various techniques that have been applied to the human action recognition task and a discussion of their relative merits. In particular, both template matching and neural networks have received much attention e.g. [BD96, GXT94], respectively. Template matching techniques offer low computational complexity and ease of implementation over state space approaches such as the HMM. However, they are typically more sensitive to noise and variation in the speed of movements [WHT03]. Neural networks have been found to give very similar results to the HMM on human motion classification problems [BMB+04].

It is the HMM's ability both to classify *and* to synthesise poses that sees it adopted in this thesis. This chapter presents a quantitative investigation into the difficulty of the classification task in a joint activity pose space. Depending on the activities that are present, the task is a challenging one and the results motivate the combination of HMMs with particle-based estimation. By sustaining multiple hypotheses – each a result of HMM synthesis – these techniques are able to support, and eventually to resolve, ambiguity in the classification task [DLC08a].

D.2 Related Work

In the first application of HMMs to human motion recognition, Yamato *et al.* [YOI92] classified a set of 6 different tennis strokes. Good known subject classification results (better than 90%) were achieved, but recognition rates drop considerably when the test subject is removed from the training data. This work is interesting for its use of hidden states with very short duration; 36 states were used to model sequences of between 23 and 70 symbols in length. Wilson and Bobick [WB95] adopted the HMM for recognition of simple gestures such as waving. The authors note that although gestures may appear to us as a well defined sequence of conceptual states, they may appear to sensors as a complex mixture of perceptual states. No topology shaping was therefore enforced[1] and the HMM was left potentially ergodic. The resulting model represents a *wave* activity; individual hidden states are particular physical configurations of the arm, observations are low resolution images of the arm captured from a fixed camera. The HMM construction – noisy observations of an underlying stochastic process – is a natural and intuitively appealing choice.

HMMs have also been adopted where a much tighter coupling between conceptual and perceptual states is possible. Campbell *et al.* studied observations from a vision system able to give accurate 3D estimates of a subject's hand positions. The authors view human gesture performance as a doubly stochastic system ideal for the application of HMMs: a human's intentions to produce movement are imprecisely realised (by their muscles) and the resulting pose configurations are then imprecisely measured (by sensors). They undertake a study of the best choice of features (e.g. hand position, velocity and acceleration) for classification of T'ai Chi moves performed by a known subject. In this work both hidden states and observations occupy the same domain. One can imagine the vector of observation parameters tracing out a trajectory through state space that passes through or nearby static hidden states belonging to one of a number of separate HMMs. If sensor errors are small (e.g. MoCap data of full body movement,

[1]For example, HMM structure (such as left-to-right) can be enforced by initialising state transition matrix entries to zero before Baum-Welch training.

or the use of a "dataglove" for gestures [LX96]) then natural differences in the performance of gesture may become the dominant source of variation.

Variations in the performance of gesture and activity are often most marked between different subjects. Classifying the gestures of unknown subjects – subjects for whom there is no training data – is therefore challenging, e.g. [YOI92]. Bowden [Bow99] has shown that extracting a richer high dimensional state vector and then performing dimensionality reduction with principal components analysis can help a model to generalise, alleviating the known subject requirement. Brand and Hertzmann [BH00] introduced stylistic HMMs (SHMMs) which specifically address this problem by attempting to recover the "essential structure" of data while disregarding its "accidental properties" in a separation of structure and style.

Brand [BOP96] comments on the shortcomings of HMMs for vision research, noting that many human activities are not well described by the Markov condition, as they feature multiple interacting processes. This fact has motivated adoption of higher order models such as the variable length Markov model by Galata *et al.* [GJH01]. Longer state histories are useful for encoding activity with correlations at different temporal scales. For example, they can be useful where an HMM overlaps to form a junction in the state space. Where *different* activities share conceptual states, classification of the associated perceptual state is unavoidably ambiguous. Consideration of previous states may alleviate the problem. An investigation into classification accuracy versus test data batch length is presented in Section D.5.3.

In the wider context of Bayesian tracking, however, the use of a higher order temporal model is not strictly compatible with the recursive filtering equation (see also Eq. 2.1). In Chapter 4 and Chapter 5 HMMs are adopted for particle dispersion and are successfully used to classify single poses in intra- and inter-activity scenarios, respectively. Ultimately it seems that the need for highly accurate

classifications is mitigated by the use of multiple hypothesis support during inference. That is, given enough particles the predictions of every competing HMM are well represented.

D.3 State Vector Definition

Given a sequence of MoCap frames $m = 1, ..., M$ for a particular activity a subset of feature points were extracted. These were the markers on the right shoulder, elbows, wrists, right hip, knees and ankles. Angles between right radius and right humerus, both radii, right femur and right tibia, and both tibia were then calculated. For example, the angle between the two radii bones may be calculated from the marker coordinates in the global coordinate system $\underline{c}_{\text{Relb}}$, $\underline{c}_{\text{Rwri}}$, $\underline{c}_{\text{Lelb}}$, $\underline{c}_{\text{Lwri}}$ by defining limb vectors $\underline{l}_{\text{Lrad}} = \underline{c}_{\text{Lwri}} - \underline{c}_{\text{Lelb}}$ and $\underline{l}_{\text{Rrad}} = \underline{c}_{\text{Rwri}} - \underline{c}_{\text{Relb}}$. The relationship

$$|\underline{l}_{\text{Lrad}}||\underline{l}_{\text{Rrad}}| \cos\theta = \underline{l}_{\text{Lrad}}.\underline{l}_{\text{Rrad}} \tag{D.1}$$

was then used to determine the angle θ between limbs. In this way, a state vector was compiled at each frame

$$\underline{x}_m = \begin{pmatrix} \theta_{\text{Rrad,Lrad}} \\ \theta_{\text{Rhum,Rrad}} \\ \theta_{\text{Rfem,Rtib}} \\ \theta_{\text{Rtib,Ltib}} \end{pmatrix}, \quad m = 1, ..., M. \tag{D.2}$$

As limbs are considered relative to one another, the state vector should remain consistent for a particular pose regardless of the subject's location in the world coordinate system, see Fig. D.2(a). In order to minimise ambiguity in the state space, the state vector was extended to contain a finite differencing estimate of $\Delta\underline{x}_m$ made using the previous timestep, i.e. $\Delta\underline{x}_m \approx \underline{x}_m - \underline{x}_{m-1}$.

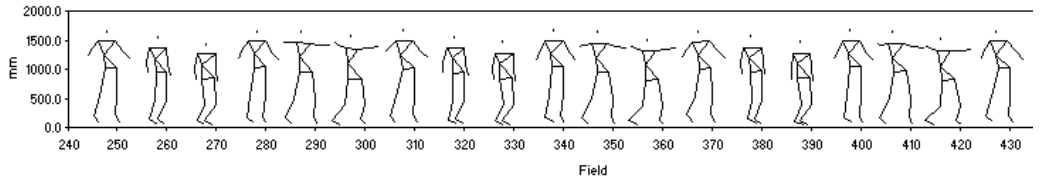
D.4 Learning HMMs

A set of 6 subjects were recorded performing 6 periodic activities using a Vicon MoCap system. These were walking on the spot, running on the spot, one-footed skipping, two-footed skipping and two types of star jump, see Fig. D.1. The Vicon system provided coordinates of markers attached to feature points on each subject, in the manner of a 3D moving light display (MLD) system. Feature points were located on the head, torso, shoulders, elbows, wrists, hips, knees and ankles. Each activity was performed by at least 3 individuals and state vectors were extracted at each frame as described in Section D.3. Each resulting sequence was divided into two halves, each of between 5 to 12 seconds at 60fps. One half was used for training, the other retained for testing.

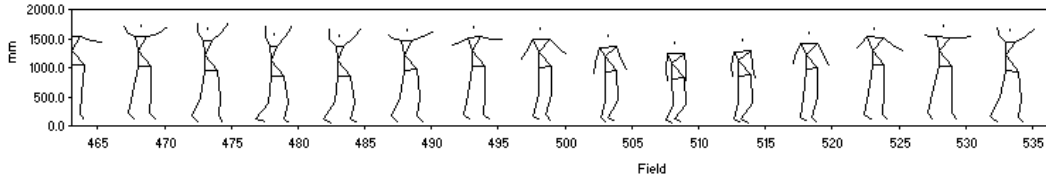
Each of the activities was represented by 30 states, each with a Gaussian observation density. Initial estimates of the state means and covariance matrices were found by K-means clustering. The transition matrix $\boldsymbol{A}$ was initialised randomly (with each row summing to 1) and the prior $\underline{a}$ set with every value equal to $1/N$, where N is the total number of states. Elements of $\underline{a}$ were not reestimated in order that test data could begin at any point during the activity unit with no probabilistic penalty. The transition probabilities and state means and covariances were reestimated using no more than 20 iterations of the Baum-Welch update equations (see Appendix C for details).

D.5 Experiments

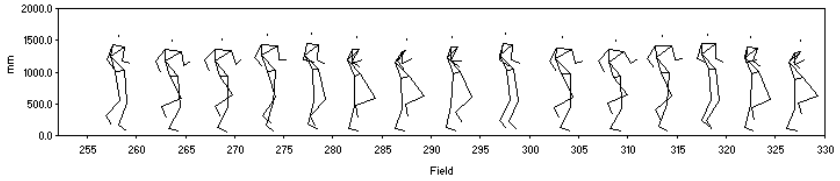
Using the "forward algorithm" it is relatively simple to calculate the possibility that test data was emitted by an HMM (see also Section C.2). By training a set of N HMMs using training data from a set of N activities it is possible to classify subsequent batches of training data between the N activities. Of particular interest is *how much* test data is required to achieve reliable classification.



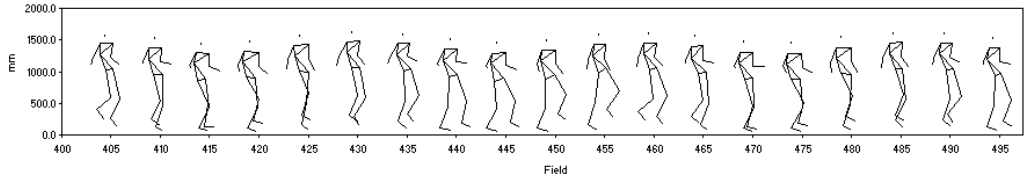
(a) jumping on the spot 1 (*jump1*)



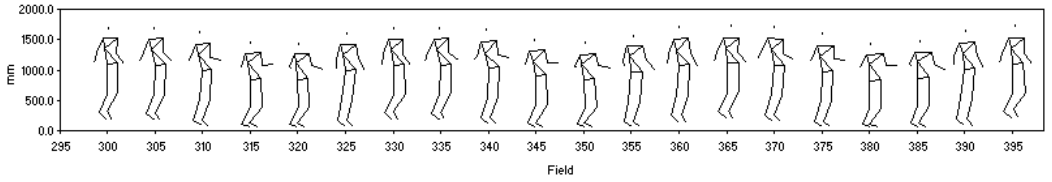
(b) jumping on the spot 2 (*jump2*)



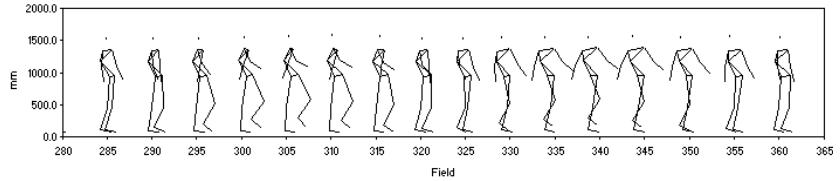
(c) Running on the spot (*run*)



(d) Skipping on the spot 1 (*skip1*)



(e) Skipping on the spot 2 (*skip2*)



(f) walking on the spot (*walk*)

FIGURE D.1: MoCap activity data.

D.5.1 Synthesis

Once trained, an HMM can be used to synthesise activity data. A starting state is chosen with probability proportional to the set of likelihoods $\underline{a}$, and a state vector

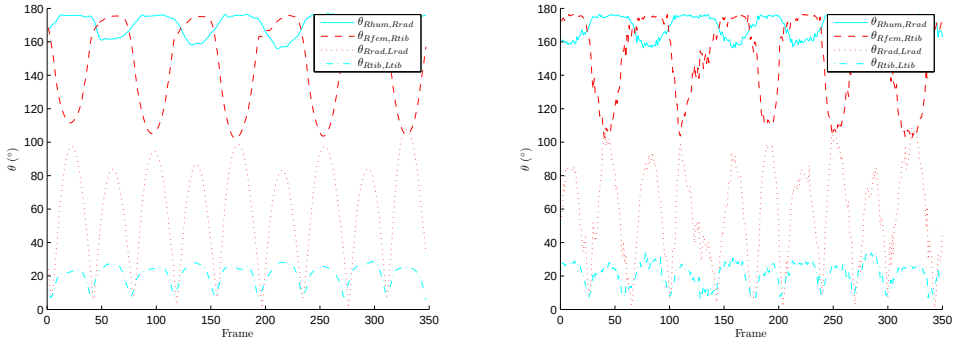


FIGURE D.2: *Walk* data: (a) an example of the state vector time series for a walking subject; (b) synthetic walking data produced by an HMM.

sampled from the chosen state's observation density. Subsequent transitions are then made via the recovered state transition matrix $\boldsymbol{A}$, each accompanied by the emission of a vector of state parameters. An example of synthetic *walk* data is shown in Fig. D.2(b).

D.5.2 Classification

Each subject's test data for each activity was tested separately. The probability $p(X_*|\lambda)$ was calculated 5 times for each test sequence $X_* = \{\underline{x}_1, ..., \underline{x}_M\}$, the Baum-Welch algorithm having been allowed to reconverge to a newly estimated set of parameters λ each time. Table D.1 summarizes the classification results for each batch of activity test data against each trained model. For cross comparison, the forward variable is calculated over the first 2.5 seconds of each test sequence. Classification results are concentrated on the diagonal and no misclassifications are made for four of the activities. In the cases of *jump1* and *skip1*, all off-diagonal classifications are due to just one test sequence in each batch, with all other sequences being correctly classified. Further discussion is given in Section D.6.

	λ_{jump1}	λ_{jump2}	λ_{run}	λ_{skip1}	λ_{skip2}	λ_{walk}
X_{jump1}	**17/20**	3/20	0/20	0/20	0/20	0/20
X_{jump2}	0/20	**20/20**	0/20	0/20	0/20	0/20
X_{run}	0/15	0/15	**15/15**	0/15	0/15	0/15
X_{skip1}	0/15	1/15	0/15	**12/15**	2/15	0/15
X_{skip2}	0/20	0/20	0/20	0/20	**20/20**	0/20
X_{walk}	0/15	0/15	0/15	0/15	0/15	**15/15**

TABLE D.1: Activity classification results.

D.5.3 Confusion Matrices

Fig. D.3 shows the forward variable for each activity model as a function of the number of frames of one subject's test *walk* sequence taken as input (m). *Walk* is not correctly established as the most likely activity until $m = 4$ and *jump2* temporarily overtakes it for $m = 27, 28, 29$. *Walk* subsequently remains the most likely interpretation. $p(X_{\text{walk}}|\lambda_{\text{run}})$ proved extremely unlikely, causing arithmetic overflow by $m = 2$ and is not plotted.

In order to determine how quickly reliable classification may take place across the activity cycles, each test sequence was divided into smaller segments for evaluation with the forward variable. Segment lengths of 2, 4, 8, 16, 32 and 64 frames were used and all possible continuous segments of this length tested, with data segments allowed to overlap, thus maximising the number of classification problems considered. The classification results were used to form a confusion matrix for each activity and these are shown in Table D.2.

D.6 Discussion and Conclusions

Given reasonably long batches of test data good classification of activity is achieved (see Table D.1). Classification between the broad activity types (*run*, *walk*, *skip*, *jump*) is reliable. Although subtle changes in activity proved more difficult – e.g. there is confusion between the two star jumps and one-footed and two-footed skipping – classification rates remained upwards of 80%. Reduction of test data segment length for reliably classified activities such as *jump2* and *skip2*

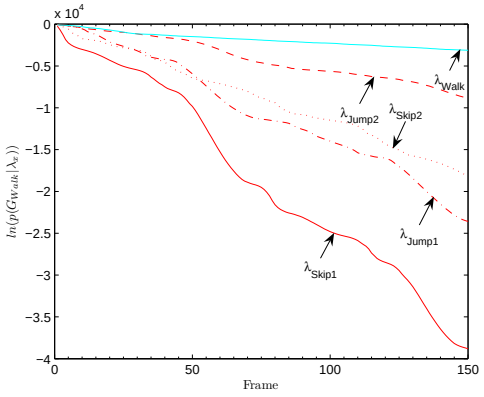


FIGURE D.3: Forward variable for one subject's test *walk* sequence for all activity models as a function of the number of frames (m).

produced a gradual spread in the distribution across activity columns of the confusion matrix (see Table D.2). However, correct classification remains above 80% using only two observations (separated by 1/60th of a second). These results demonstrate a surprising ability to discriminate between activities given only very small fragments of test data and a rather "bland" set of state parameters.

The two *jump* activities considered in this chapter share some of the same poses. It is therefore unavoidable that some test poses will divided between HMMs. In pure classification tasks this result may be unacceptable, and longer state histories don't guarantee improvement (see lower rows in Table D.2). Viewed in the context of multiple hypothesis tracking, however, this result is likely to be sufficient. An ensemble of observations (particles) will be divided between competing HMM states with, say, a 70-30 ratio for *jump1* and *jump2* activities. If the HMMs are then used for particle propagation, the future predictions of *both* activity models are represented.

(a) *jump1*

	λ_{jump1}	λ_{jump2}	λ_{run}	λ_{skip1}	λ_{skip2}	λ_{walk}
$M = 2$	**0.7190**	0.2810	0.0000	0.0000	0.0000	0.0000
$M = 4$	**0.7045**	0.2955	0.0000	0.0000	0.0000	0.0000
$M = 8$	**0.7062**	0.2938	0.0000	0.0000	0.0000	0.0000
$M = 16$	**0.7221**	0.2779	0.0000	0.0000	0.0000	0.0000
$M = 32$	**0.7511**	0.2489	0.0000	0.0000	0.0000	0.0000
$M = 64$	**0.7738**	0.2262	0.0000	0.0000	0.0000	0.0000

(b) *jump2*

	λ_{jump1}	λ_{jump2}	λ_{run}	λ_{skip1}	λ_{skip2}	λ_{walk}
$M = 2$	0.0397	**0.9460**	0.0000	0.0011	0.0132	0.0000
$M = 4$	0.0311	**0.9566**	0.0000	0.0000	0.0122	0.0000
$M = 8$	0.0238	**0.9706**	0.0000	0.0000	0.0057	0.0000
$M = 16$	0.0012	**0.9988**	0.0000	0.0000	0.0000	0.0000
$M = 32$	0.0000	**1.0000**	0.0000	0.0000	0.0000	0.0000
$M = 64$	0.0000	**1.0000**	0.0000	0.0000	0.0000	0.0000

(c) *run*

	λ_{jump1}	λ_{jump2}	λ_{run}	λ_{skip1}	λ_{skip2}	λ_{walk}
$M = 2$	0.0229	0.0249	**0.9254**	0.0191	0.0076	0.0000
$M = 4$	0.0077	0.0464	**0.9168**	0.0251	0.0039	0.0000
$M = 8$	0.0000	0.0614	**0.9287**	0.0099	0.0000	0.0000
$M = 16$	0.0000	0.0686	**0.9293**	0.0021	0.0000	0.0000
$M = 32$	0.0000	0.0531	**0.9215**	0.0254	0.0000	0.0000
$M = 64$	0.0000	0.1128	**0.8872**	0.0000	0.0000	0.0000

(d) *skip1*

	λ_{jump1}	λ_{jump2}	λ_{run}	λ_{skip1}	λ_{skip2}	λ_{walk}
$M = 2$	0.0267	0.1336	0.0095	**0.7977**	0.0324	0.0000
$M = 4$	0.0270	0.1351	0.0077	**0.8147**	0.0154	0.0000
$M = 8$	0.0237	0.1443	0.0040	**0.8162**	0.0119	0.0000
$M = 16$	0.0104	0.1432	0.0000	**0.8423**	0.0041	0.0000
$M = 32$	0.0046	0.1175	0.0000	**0.8641**	0.0138	0.0000
$M = 64$	0.0237	0.1834	0.0000	**0.7929**	0.0000	0.0000

(e) *skip2*

	λ_{jump1}	λ_{jump2}	λ_{run}	λ_{skip1}	λ_{skip2}	λ_{walk}
$M = 2$	0.0145	0.0193	0.0000	0.0386	**0.9277**	0.0000
$M = 4$	0.0081	0.0114	0.0000	0.0309	**0.9495**	0.0000
$M = 8$	0.0017	0.0017	0.0000	0.0017	**0.9950**	0.0000
$M = 16$	0.0000	0.0000	0.0000	0.0000	**1.0000**	0.0000
$M = 32$	0.0000	0.0000	0.0000	0.0000	**1.0000**	0.0000
$M = 64$	0.0000	0.0000	0.0000	0.0000	**1.0000**	0.0000

(f) *walk*

	λ_{jump1}	λ_{jump2}	λ_{run}	λ_{skip1}	λ_{skip2}	λ_{walk}
$M = 2$	0.0114	0.1641	0.0000	0.0000	0.0000	**0.8245**
$M = 4$	0.0089	0.1705	0.0000	0.0000	0.0000	**0.8206**
$M = 8$	0.0000	0.1667	0.0000	0.0000	0.0000	**0.8333**
$M = 16$	0.0000	0.1267	0.0000	0.0000	0.0000	**0.8733**
$M = 32$	0.0000	0.0655	0.0000	0.0000	0.0000	**0.9345**
$M = 64$	0.0000	0.0000	0.0000	0.0000	0.0000	**1.0000**

TABLE D.2: Activity classification rate versus data segment length.

Bibliography

[AMGC02] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. on Sig. Proc.*, 50(2):174–188, 2002.

[ARS08] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *CVPR*, pages 1–8, 2008.

[AT04a] A. Agarwal and B. Triggs. 3D human pose from silhouettes by relevance vector regression. In *CVPR*, pages 882–888, 2004.

[AT04b] A. Agarwal and B. Triggs. Tracking articulated motion with piecewise learned dynamical models. In *ECCV*, pages 54–65, 2004.

[AT06] A. Agarwal and B. Triggs. Recovering 3D human pose from monocular images. *PAMI*, 28(1):44–58, 2006.

[AUAD07] P. Azad, A. Ude, T. Asfour, and R. Dillmann. Stereo-based markerless human motion capture for humanoid robot systems. In *ICRA*, pages 3951–3956, 2007.

[BB06] A. O. Bălan and M. J. Black. An adaptive appearance model approach for model-based articulated object tracking. In *CVPR*, pages 758–765, 2006.

[BD96] A. Bobick and J. Davis. Real-time recognition of activity using temporal templates. In *IEEE Workshop on Applications of Computer Vision*, pages 39–42, 1996.

[BEB08] J. Bandouch, F. Engstler, and M. Beetz. Evaluation of hierarchical sampling strategies in 3D human pose estimation. In *BMVC*, pages 925–934, 2008.

[BH00] M. Brand and A. Hertzmann. Style machines. In *SIGGRAPH*, pages 183–192, 2000.

[BI98] A. Blake and M. Isard. *Active Contours*. Springer-Verlag, 1998.

[BM98] C. Bregler and J. Malik. Tracking people with twists and exponential maps. In *CVPR*, pages 8–15, 1998.

[BMB+04] I. Boesnach, J. Moldenhauer, C. Burgmer, T. Beth, V. Wank, and K. Bos. Classification of phases in human motions by neural networks and hidden Markov models. In *CCIS*, pages 976–981, 2004.

[BMS97] R. Bowden, T. A. Mitchell, and M. Sarhadi. Cluster based non-linear principal components analysis. *Electronics Letters*, 33(22):1858–1859, 1997.

[BMS98] R. Bowden, T. A. Mitchell, and M. Sarhadi. Reconstructing 3D pose and motion from a single camera view. In *BMVC*, pages 904–913, 1998.

[BMS00] R. Bowden, T. Mitchell, and M. Sahardi. Non-linear statistical models for the 3D reconstruction of human pose and motion from monocular image sequences. *IVC*, 18(9):729–737, 2000.

[BN03] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, June 2003.

[BOP96] M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complex action recognition. In *CVPR*, pages 994–999, 1996.

[Bow99] R. Bowden. *Learning non-linear models of shape and motion*. PhD thesis, Brunel University, 1999.

[Bow00] R. Bowden. Learning statistical models of human motion. In *CVPR: Workshop on Human Modeling, Analysis and Synthesis*, pages 10–17, 2000.

[Bra99] M. Brand. Shadow puppetry. In *ICCV*, pages 1237–1244, 1999.

[BS00] R. Bowden and M. Sarhadi. Building temporal models for gesture recognition. In *BMVC*, pages 32–41, 2000.

[BSB05] A. O. Bălan, L. Sigal, and M. J. Black. A quantitative evaluation of video-based 3D person tracking. In *VS-PETS*, pages 349–356, 2005.

[CBA+96] L. Campbell, D. Becker, A. Azarbayejani, A. Bobick, and A. Pentland. Invariant features for 3D gesture recognition. In *F&G*, pages 157–162, 1996.

[CCVC07] V. Camomilla, A. Ceratti, G. Vannozzi, and A. Cappozzo. An optimized protocol for hip joint centre determination using the functional medthod. *Journal of Biomechanics*, 39(6):1096–1106, 2007.

[CGH05] F. Caillette, A. Galata, and T. Howard. Real-time 3-D human body tracking using variable length Markov models. In *BMVC*, pages 469–478, 2005.

[CMG+10] S. Corazza, L. Mündermann, E. Gambaretto, G. Ferrigno, and T. Andriacchi. Markerless motion capture through visual hull, articulated ICP and subject specific model generation. *IJCV*, 87(1–2):156–169, 2010.

[CMU] Carnegie Mellon University graphics lab motion capture database. http://mocap.cs.cmu.edu/.

[CR99] T.-J. Cham and J. M. Rehg. A multiple hypothesis approach to figure tracking. In *CVPR*, pages 239–245, 1999.

[Dau09] B. Daubney. *Using low-level motion for high-level vision.* PhD thesis, Bristol University, 2009.

[DBR00] J. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *CVPR*, pages 2126–2133, 2000.

[Dem03] D. Demirdjian. Enforcing constraints for human body tracking. In *WOMOT*, 2003.

[Dem04] D. Demirdjian. Combining geometric- and view-based approaches for articulated pose estimation. In *ECCV*, pages 183–194, 2004.

[DLC08a] J. Darby, B. Li, and N. P. Costen. Behaviour based particle filtering for human articulated motion tracking. In *ICPR*, pages 1–4, 2008.

[DLC08b] J. Darby, B. Li, and N. P. Costen. Human activity tracking from moving camera stereo data. In *BMVC*, pages 865–874, 2008.

[DLC08c] J. Darby, B. Li, and N. P. Costen. Tracking a walking person using activity-guided annealed particle filtering. In *F&G*, pages 1–6, 2008.

[DLC+09] J. Darby, B. Li, N. P. Costen, D. J. Fleet, and N. D. Lawrence. Backing off: hierarchical decomposition of activity for 3D novel pose recovery. In *BMVC*, pages 1–11, 2009.

[DLC10] J. Darby, B. Li, and N. P. Costen. Tracking human pose with multiple activity models. *Pattern Recognition*, 43(2010):3042–3058, 2010.

[DNBB99] J. Deutscher, B. North, B. Bascle, and Andrew Blake. Tracking through singularities and discontinuities by random sampling. In *ICCV*, pages 1144–1149, 1999.

[DR05] J. Deutscher and I. Reid. Articulated body motion capture by stochastic search. *IJCV*, 61(2):185–205, 2005.

[DTS+05] D. Demirdjian, L. Taycher, G. Shakhnarovich, K. Grauman, and T. Darrell. Avoiding the "streetlight effect": Tracking by exploring likelihood modes. In *ICCV*, pages 357–364, 2005.

[EL04] A. Elgammal and C.-S. Lee. Inferring 3D body pose from silhouettes using activity manifold learning. In *CVPR*, page 681688, 2004.

[ETL07] C. H. Ek, P. H. S. Torr, and N. D. Lawrence. Gaussian process latent variable models for human pose estimation. In *MLMI*, pages 132–143, 2007.

[EVA10] Special issue on evaluation of articulated human motion and pose estimation. *IJCV*, 87(1–2):1–190, 2010.

[FE73] M. Fischler and R. Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computer*, 22(1):67–92, 1973.

[FH05] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61(1):55–79, 2005.

[Fox01] D. Fox. KLD-sampling: Adaptive particle filters. In *NIPS*, pages 713–720, 2001.

[GD96] D. Gavrila and L. S. Davis. 3D model-based tracking of humans in action: a multi-view approach. In *CVPR*, pages 73–80, 1996.

[GdBUP95] L. Goncalves, E. di Bernado, E. Ursella, and P. Perona. Monocular tracking of the human arm in 3D. In *ICCV*, pages 764–770, 1995.

[GJH01] A. Galata, N. Johnson, and D. Hogg. Learning variable length markov models of behaviour. *CVIU*, 81(3):398–413, March 2001.

[GMHP04] K. Grochow, S. L. Martin, A. Hertzmann, and Z. Popović. Style-based inverse kinematics. *ACM Trans. Graph.*, 23(3):522–531, 2004.

[GSD03] K. Grauman, G. Shakhnarovic, and T. Darrell. Inferring 3D structure with a statistical image-based shape model. In *ICCV*, pages 641–647, 2003.

[GXT94] Y. Guoa, G. Xu, and S. Tsuji. Understanding human motion patterns. In *ICPR*, pages 325–329, 1994.

[HD04] N. R. Howe and A. Deschamps. Better foreground segmentation through graph cuts. Technical report, Smith College, 2004. http://arxiv.org/abs/cs.CV/0401017.

[HGC+07] S. Hou, A. Galata, F. Caillette, N. Thacker, and P. Bromiley. Real-time body tracking using a Gaussian process latent variable model. In *ICCV*, pages 1–8, 2007.

[HH97] T. Heap and D. Hogg. Improving specificity in PDMs using a hierarchical approach. In *BMVC*, pages 80–89, 1997.

[HH98] T. Heap and D. Hogg. Wormholes in shape space: Tracking through discontinuous changes in shape. In *ICCV*, pages 344–349, 1998.

[HLF00] N. R. Howe, M. E. Leventon, and W. T. Freeman. Bayesian reconstruction of 3D human motion from single-camera video. In *NIPS*, pages 820–826, 2000.

[HLWJ08] L. Han, W. Liang, X. Wu, and Y. Jia. human action recognition using discriminative models in the learned hierarchical manifold space. In *F&G*, pages 1–6, 2008.

[How07] N. Howe. Silhouette lookup for monocular 3D pose tracking. *IVC*, 25(3):331–341, 2007.

[HUF04] L. Herda, R. Urtasun, and P. Fua. Hierarchical implicit surface joint limits to constrain video-based motion capture. In *ECCV*, pages 405–418, 2004.

[I2I] nVela. Hydra–Stereo Webcam. http://nvela.co.uk, Viewed August 2009.

[IB98a] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *IJCV*, 29(1):5–28, 1998.

[IB98b] M. Isard and A. Blake. Icondensation: Unifying low-level and high-level tracking in a stochastic framework. In *ECCV*, pages 893–908, 1998.

[IB98c] M. Isard and A. Blake. A mixed-state CONDENSATION tracker with automatic model-switching. In *ICCV*, pages 107–112, 1998.

[IF01] S. Ioffe and D. A. Forsyth. Probabilistic methods for finding people. *IJCV*, 43(1):45–68, 2001.

[Isa03] M. Isard. PAMPAS: Real-valued graphical models for computer vision. In *CVPR*, pages 613–620, 2003.

[JFEM03] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi. Robust online appearance models for visual tracking. *PAMI*, 25(10):1296–1311, 2003.

[Jia09] H. Jiang. Human pose estimation using consistent max covering. In *ICCV*, pages 1–8, 2009.

[JTH99] N. Jojic, M. Turk, and T. S. Huang. Tracking self-occluding articulated objects in dense disparity maps. In *ICCV*, pages 123–130, 1999.

[KB04] K. Konolige and D. Beymer. *SRI Small Vision System Calibration Addendum to the User's Manual*. SRI International, November 2004.

[KF00] O. King and D. A. Forsyth. How does CONDENSATION behave with a finite number of samples? In *ECCV*, pages 695–709, 2000.

[KHM00] I.A. Karaulova, P.M. Hall, and A.D. Marshall. A hierarchical model of dynamics for tracking people with a single video camera. In *BMVC*, pages 352–361, 2000.

[KM96] I. A. Kakadiaris and D. Metaxas. Model-based estimation of 3D human motion with occlusion based on active multi-viewpoint selection. In *CVPR*, pages 81–87, 1996.

[KM08] G. Kravaritis and B. Mulgrew. Variable-mass particle filter for road-constrained vehicle tracking. *EURASIP J. on Adv. in Sig. Proc.*, pages 1–13, 2008.

[Kon97] K. Konolige. Small vision systems: hardware and implementation. In *ISRR*, pages 111–116, 1997.

[Law05] N. D. Lawrence. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. *JMLR*, 6:1783–1816, 2005.

[LC04] M. Lee and I. Cohen. Proposal maps driven MCMC for estimating human body pose in static images. In *CVPR*, volume 2, pages 334–341, 2004.

[LH05] X. Lan and D. Huttenlocher. Beyond trees: Common factor models for 2D human pose recovery. In *ICCV*, pages 470–477, 2005.

[LM07] N. D. Lawrence and A. J. Moore. Hierarchical Gaussian process latent variable models. In *ICML*, pages 481–488, 2007.

[LPS07] Z. Lu, M. C. Perpinan, and C. Sminchisescu. People tracking with the Laplacian eigenmaps latent variable model. In *NIPS*, pages 1705–1712, 2007.

[LQC06] N. D. Lawrence and J. Quiñonero-Candela. Local distance preservation in the GP-LVM through back constraints. In *ICML*, pages 513–520, 2006.

[LTS07] R. Li, T.-P. Tian, and S. Sclaroff. Simultaneous learning of nonlinear manifold and dynamical models for high-dimensional time series. In *ICCV*, pages 1–8, 2007.

[LX96] C. Lee and Y. Xu. Online, interactive learning of gestures for human/robot interfaces. In *ICRA*, pages 2982–2987, 1996.

[LYST06] R. Li, M-H. Yang, S. Sclaroff, and T-P. Tian. Monocular tracking of 3D human motion with a coordinated mixture of factor analyzers. In *ECCV*, pages 137–150, 2006.

[May79] P. S. Maybeck. *Stochastic models, estimation and control*. Academic Press, New York, 1979.

[MG01] T. B. Moeslund and E. Granum. A survey of computer vision-based human motion capture. *CVIU*, 81(3):231–268, 2001.

[MHK06] T. B. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *CVIU*, 104(2):90–126, 2006.

[MI00] J. MacCormick and M. Isard. Partitioned sampling, articulated objects, and interface-quality hand tracking. In *ECCV*, pages 3–19, 2000.

[MM06] G. Mori and J. Malik. Recovering 3D human body configurations using shape contexts. *PAMI*, 28(7):1052–1062, 2006.

[MN78] D. Marr and H. K. Nishihara. Representation and recognition of the spatial organization of three dimensional structure. *Proceedings of the Royal Society of London B*, 200:269–294, 1978.

[MOB05] A.S. Micilotta, E.J. Ong, and R. Bowden. Detection and tracking of humans by probabilistic body part assembly. In *BMVC*, pages 429–438, 2005.

[MOB06] A.S. Micilotta, E.J. Ong, and R. Bowden. Real-time upper body detection and 3D pose estimation in monoscopic images. In *ECCV*, pages 139–150, 2006.

[Møl93] M. F. Møller. A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, 6(4):525–533, 1993.

[MP97] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *PAMI*, 19(7):696–710, 1997.

[MP06] K. Moon and V. Pavlović. Impact of dynamics on subspace embedding and tracking of sequences. In *CVPR*, pages 198–205, 2006.

[NBIR00] B. North, A. Blake, M. Isard, and J. Rittscher. Learning and classification of complex dynamics. *PAMI*, 22(9):1016–1034, 2000.

[Nor98] J. R. Norris. *Markov Chains*. Cambridge University Press, 1998.

[NTTC05] R. Navaratnam, A. Thayananthan, P. H. S. Torr, and R. Cipolla. Hierarchical part-based human body pose estimation. In *BMVC*, pages 1–10, 2005.

[OG99] E.-J. Ong and S. Gong. A dynamic human model using hybrid 2D-3D representations in hierarchical PCA space. In *BMVC*, pages 33–42, 1999.

[O'H78] A. O'Hagan. Curve fitting and optimal design for prediction. *Journal of the royal statistical society*, 40(B):1–42, 1978.

[OMBH06] E.-J. Ong, A. S. Micilotta, R. Bowden, and A. Hilton. Viewpoint invariant exemplar-based 3D human tracking. *CVIU*, 104(2):178–189, 2006.

[PF02] E. Poon and D. J. Fleet. Hybrid monte carlo filtering: Edge-based people tracking. In *IEEE Workshop on Motion and Video Computing*, pages 151–158, 2002.

[PF03] R. Plänkers and P. Fua. Articulated soft objects for multiview shape and motion capture. *PAMI*, 25(9):1182–1187, 2003.

[Pop07a] R. Poppe. Evaluating example-based pose estimation: Experiments on the humaneva sets. In *EHuM2*, pages 1–8, 2007.

[Pop07b] R.W. Poppe. Vision-based human motion analysis: an overview. *CVIU*, 108(1-2):4–18, 2007.

[PP06] R. Poppe and M. Poel. Comparison of silhouette shape descriptors for example-based human pose recovery. In *F&G*, pages 541–546, 2006.

[PRCM99] V. Pavlović, J. M. Rehg, T. J. Cham, and K. P. Murphy. A dynamic Bayesian network approach to figure tracking using learned dynamic models. In *ICCV*, pages 94–101, 1999.

[PRM00] V. Pavlović, J. M. Rehg, and J. MacCormick. Learning switching linear models of human motion. In *NIPS*, pages 626–632, 2000.

[QDLM08] S. Quirion, C. Duchesne, D. Laurendeau, and M. Marchand. Comparing GPLVM approaches for dimensionality reduction in character animation. *Journal of WSCG*, 16(1-3):41–48, 2008.

[Rab89] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[Ram06] D. Ramanan. Learning to parse images of articulated bodies. In *NIPS*, pages 1129–1136, 2006.

[RB99] J. Rittscher and A. Blake. Classification of human body motion. In *ICCV*, pages 634–639, 1999.

[RFZ07] D. Ramanan, D. A. Forsyth, and A. Zisserman. Tracking people by learning their appearance. *PAMI*, 29(1):65–81, 2007.

[RKM08] B. Rosenhahn, R. Klette, and D. Metaxas, editors. *Human Motion - Understanding, Modeling, Capture and Animation*, volume 36 of *Computational Imaging and Vision*. Springer, Dordrecht, The Netherlands, 2008.

[RRR08a] L. Raskin, E. Rivlin, and M. Rudzsky. Using Gaussian process annealing particle filter for 3D human tracking. *EURASIP J. on Adv. in Sig. Proc.*, pages 1–13, 2008.

[RRR+08b] G. Rogez, J. Rihan, S. Ramalingam, Orrite C., and P. H. S. Torr. Randomized trees for human pose detection. In *CVPR*, pages 1–8, 2008.

[RRR09] L. Raskin, E. Rivlin, and M. Rudzsky. 3D human body-part tracking and action classification using a hierarchical body model. In *BMVC*, pages 1–11, 2009.

[RS00] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, December 2000.

[RS01] R. Rosales and S. Sclaroff. Learning body pose via specialized maps. In *NIPS*, pages 1263–1270, 2001.

[RST94] D. Ron, Y. Singer, and N. Tishby. The power of amnesia. In *NIPS*, pages 176–183, 1994.

[RW06] C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. MIT Press, 2006.

[SB01] H. Sidenbledh and M. J. Black. Learning image statistics for bayesian tracking. In *ICCV*, pages 709–716, 2001.

[SB06a] L. Sigal and M. J. Black. HumanEva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Technical Report CS-06-08, Brown University, Providence, RI, 2006.

[SB06b] L. Sigal and M. J. Black. Measure locally, reason globally: Occlusion-sensitive articulated pose estimation. In *CVPR*, pages 2041–2048, 2006.

[SB06c] L. Sigal and M. J. Black. Predicting 3D people from 2D pictures. In *AMDO*, pages 185–195, 2006.

[SB10] L. Sigal and M. J. Black. Guest editorial: state of the art in image- and video-based human pose and motion estimation. *IJCV*, 87(1–2):1–3, 2010.

[SBB07] L. Sigal, A. Bălan, and M. J. Black. Combined discriminative and generative articulated pose and non-rigid shape estimation. In *NIPS*, 2007.

[SBB10] R. Sigal, A. Bălan, and M. J. Black. HumanEva: Synchronised video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *IJCV*, 87(1–2):4–27, 2010.

[SBF00] H. Sidenbladh, M. J. Black, and D. J. Fleet. Stochastic tracking of 3D human figures using 2D image motion. In *ECCV*, pages 702–718, 2000.

[SBIH10] L. Sigal, M. J. Black, M. Isard, and H. Haussecker. Loose-limbed people: Estimating human pose and motion using non-parametric belief propagation. *IJCV*, 2010. (Submitted).

[SBR+04] L. Sigal, S. Bhatia, S. Roth, M. J. Black, and M. Isard. Tracking loose-limbed people. In *CVPR*, pages 421–428, 2004.

[SBS02] H. Sidenbladh, M. J. Black, and L. Sigal. Implicit probabilistic models of human motion for synthesis and tracking. In *ECCV*, pages 784–800, 2002.

[SF95] E. P. Simoncelli and W. T. Freeman. The steerable pyramid: a flexible architecture for multi-scale derivative computation. In *ICIP*, pages 444–447, 1995.

[SFAH92] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger. Shiftable multi-scale transforms. *IEEE transactions on informations theory*, 38(2), 1992.

[Sig08] L. Sigal. *Continuous-state Graphical Models for Object Localization, Pose Estimation and Tracking*. PhD thesis, Brown University, 2008.

[SJ04] C. Sminchisescu and A. Jepson. Generative modeling for continuous non-linearly embedded visual inference. In *ICML*, pages 759–766, 2004.

[SKLM05] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas. Discriminative density propagation for 3D human motion estimation. In *CVPR*, pages 390–397, 2005.

[SKM06a] C. Sminchisescu, A. Kanaujia, and D. Metaxas. Conditional models for contextual human motion recognition. *CVIU*, 104(2):210–220, 2006.

[SKM06b] C. Sminchisescu, A. Kanaujia, and D. Metaxas. Learning joint top-down and bottom-up processes for 3D visual inference. In *CVPR*, pages 1743–1752, 2006.

[Smi08] C. Sminchisescu. 3D human motion analysis in monocular video: techniques and challenges. In B. Rosenhahn, R. Klette, and D. Metaxas, editors, *Human Motion - Understanding, Modeling, Capture and Animation*, volume 36 of *Computational Imaging and Vision*, pages 185–211. Springer, Dordrecht, 2008.

[ST02a] C. Sminchisescu and B. Triggs. Building roadmaps of local minima of visual models. In *ECCV*, pages 566–582, 2002.

[ST02b] C. Sminchisescu and B. Triggs. Hyperdynamics importance sampling. In *ECCV*, pages 769–783, 2002.

[ST03a] C. Sminchisescu and B. Triggs. Estimating articulated human motion with covariance scaled sampling. *IJRR*, 22(6):371–391, June 2003.

[ST03b] C. Sminchisescu and B. Triggs. Kinematic jump processes for monocular 3D human tracking. In *CVPR*, pages 69–76, June 2003.

[SUF08] M. Salzmann, R. Urtasun, and P. Fua. Local deformation models for monocular 3D shape recovery. In *CVPR*, 2008.

[SVD03] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter sensitive hashing. In *ICCV*, page 750757, 2003.

[Tay00] C. J. Taylor. Reconstruction of articulated objects from point correspondences in a single image. *CVIU*, 80(3):349–363, 2000.

[TB99] M. E. Tipping and C. M. Bishop. Probabilistic principal components analysis. *Journal of the Royal Statistical Society*, 6(3):611–622, 1999.

[TLS05] T. P. Tian, R. Li, and S. Sclaroff. Articulated pose estimation in a learned smooth space of feasible solutions. In *CVPR Learning Workshop*, page 50, 2005.

[UF04] R. Urtasun and P. Fua. 3D human body tracking using deterministic temporal motion models. In *ECCV*, pages 92–106, 2004.

[UFF06a] R. Urtasun, D. J. Fleet, and P. Fua. 3D people tracking with Gaussian process dynamical models. In *CVPR*, pages 238–245, 2006.

[UFF06b] R. Urtasun, D. J. Fleet, and P. Fua. Temporal motion models for monocular and multiview 3D human body tracking. *CVIU*, 104(2):157–177, 2006.

[UFGP08] R. Urtasun, D. J. Fleet, A. Geiger, and Popović. Topologically-constrained latent variable models. In *ICML*, pages 1080–1087, 2008.

[UFHF05] R. Urtasun, D. J. Fleet, A. Hertzmann, and P. Fua. Priors for people tracking from small training sets. In *ICCV*, pages 403–410, 2005.

[Urt06] R. Urtasun. *Motion models for robust 3D human body tracking*. PhD thesis, Ecole Polytechnique Fédérale de Lausanne, 2006.

[Vid] Videre Design. http://www.videredesign.com, Viewed July 2008.

[VSJ08] M. Vondrak, L. Sigal, and O. C. Jenkins. Physical simulation for probabilistic motion tracking. In *CVPR*, pages 1–8, 2008.

[WB95] A.W. Wilson and A.F. Bobick. Learning visual behavior for gesture analysis. In *ISCV*, pages 229–234, 1995.

[WB01] A. D. Wilson and A. F. Bobick. Hidden Markov models for modeling and recognising gesture under variation. *IJPRAI*, 15(1):123–169, 2001.

[WFH08] J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models for human motion. *PAMI*, 30(2):283–298, 2008.

[WHT03] L. Wang, W. Hu, and T. Tan. Recent developments in human motion analysis. *Pattern Recognition*, 36(3):585–601, 2003.

[WP98] C. R. Wren and A. P. Pentland. Dynamic models of human motion. In *F&G*, pages 22–27, 1998.

[YOI92] J. Yamato, J. Ohya, and K. Ishii. Recognizing human action in time-sequential images using hidden Markov model. In *CVPR*, pages 379–385, 1992.