

GPLVMによる動作合成と 動作生成への応用に関する研究

山腰 諒一[†] 青木 工太^{††} 長橋 宏^{††}

[†] 東京工業大学 大学院総合理工学研究科 知能システム科学専攻

^{††} 東京工業大学 大学院理工学研究科 像情報工学研究施設

あらまし 本稿では、CGアニメーションなどに用いられるモーションキャプチャのデータからユーザが意図するような新たな動作を生成するシステムの提案について述べる。提案手法では、確率モデルを基本とした、Gaussian Process Latent Variable Models (GPLVM) を用いて、高次元空間の複数の動作データを同時に低次元化する。さらにユーザが低次元空間で高次元の姿勢を直感的に理解するために、基本姿勢との差や姿勢の位置などを低次元空間で視覚化する。それらを用いてユーザが意図する情報を対話的に与えることにより、所望の動作を生成する。

A Study for Motion Blending using GPLVM and Its Application for Motion Generation

Ryoichi YAMAKOSHI[†] Kohta AOKI^{††} Hiroshi NAGAHASHI^{††}

[†] Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology

^{††} Image Science and Engineering Laboratory, Tokyo Institute of Technology

Abstract In this paper, we propose an interactive editing-based motion synthesis system that makes 3D human motion data. Our proposed method uses a probabilistic model called Gaussian Process Latent Variable Models (GPLVM) to map the 3D motion data on a 2D latent space. We also generate several Pose Maps on the 2D latent space to represent the difference from a standard pose and its position. Based on the Pose Maps, users can interactively make a 3D motion sequence by only giving some target poses on the map.

1 はじめに

アニメーションの研究では、キャラクターの間らしい動きの再現するためにモーションキャプチャのデータが用いられ、動作編集、動作合成などのアプリケーションが提案されている。しかし、モーションキャプチャデータは高次元データであるため、データ処理に要するコストが少なくない。近年では、データの次元圧縮を行うことにより、高次元データをより扱いやすくする手法などが提案されている。データの低次元化によって、データの視覚化、動作の統合、補間が容易になる。しかし、低次元空間のデータから、実際の高次元特徴の姿勢などを直感的に理解することは難しい。

動作編集や合成に関しては Kovar らがモーショングラフの手法を用い、動作生成のアプリケーションなどを提案している [4]。また、Hyun らは高次元データを複数の圧縮手法により低次元化し、それぞれの圧縮手法の利点を活かしながら動作生成システムの提案を行っている [7]。

次元圧縮手法に関しては様々な提案がなされ、モーションキャプチャデータについても視覚化、分類、データの低次元化などを扱う研究が行われている。特に、従来から、PCA を使った低次元手法が挙げられる。しかし、この手

法は線形圧縮手法であり、3D の非線形データに用いることはあまり適切でないと考えられる。一方、非線形の圧縮手法として LLE [6]、Isomap [1]、multi-dimensional scaling (MDS) などが提案されている。しかし、これらのモデルではデータの過学習や適切なパラメータの設定が必要となるため、低次元空間のデータと高次元のデータを写像する適切な関数を生成することは難しい。そこで GPLVM [5] の発展手法であり、Grochow らが提案した非線形の圧縮手法 Scaled GPLVM (SGPLVM) が注目されている [3]。このモデルは少ないパラメータと少ないトレーニングデータで学習できる利点をもち、動作のトラッキング [11]、ロボットの模倣学習 [8] に応用されている。また、SGPLVM の発展手法として、Gaussian Process Dynamical Model (GPDM) が Jack M. らにより提案され、トラッキングに用いられている [12, 13]。

本論文では、モーションキャプチャデータを低次元化し、その空間で高次元の姿勢を直感的に理解するための、“姿勢マップ”提案するとともに、その姿勢マップを用いた動作生成手法について述べる。そして、その“姿勢マップ”を利用した対話的なモーション生成システムについて述べる。まず、Scaled Gaussian Process Latent Variable Models (SGPLVM [5, 3]) により、モーション

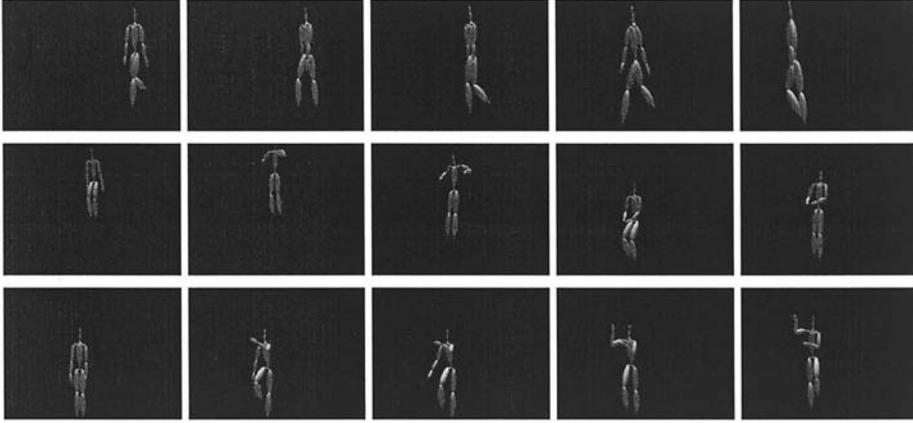
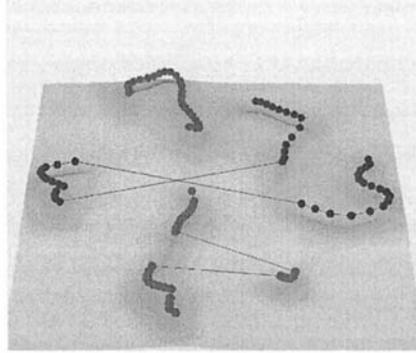


図 1 SGPLVM 空間:歩く, 飛び降りる, 梯子に登る動作から得られた空間. 高さは $-\frac{D}{2} \ln \sigma^2(x) - \frac{1}{2} \|x_i\|$, 明るいエリアが高さが低い領域, 暗いエリアは高さが高い領域である. プロット点:上段点列は歩く動作に対応する学習データ, 同様に中段点列は飛び降りる, 下段点列は梯子に登るである. 画像:プロット点の頂点から動作を復元した動作で, 上から順に歩く, 飛び降りる, 梯子に登るである. 全 114 フレームの学習データである.

キャプチャデータを低次元化する. その低次元空間において高次元の姿勢を直観的に理解するために, 複数の“姿勢マップ”を生成する. 姿勢マップはユーザがあらかじめ指定しておいた基本姿勢と姿勢位置から, 低次元化されたデータにより算出する.

2 SGPLVM

GPLVM[5] と SGPLVM[3] について簡単に述べる. GPLVM は Gaussian Process(GP)[14] と Probabilistic PCA[10] を基本とし, 学習データを非線形の写像関数を用いて低次元空間へ圧縮する手法である.

2.1 SGPLVM の定義

D次元の高次トレーニングデータ $\{y_n\}_{n=1}^N$ に対して, それぞれに低次元の潜在変数 $\{x_n\}_{n=1}^N$ が存在するとき, Probabilistic PCA [10] や Kernel PPCA[9] のモデルを

用いると,

$$p(y_n | W, \beta) = \int p(y_n | x_n, W, \beta) p(x_n) dx_n, \quad (1)$$

と表すことができる. ただし, $p(x_n) = N(x_n | 0, I)$, $p(y_n | x_n, W, \beta) = N(y_n | Wx_n, \beta^{-1}I)$ である. ここで, $p(W) = \prod_{i=1}^D N(w_i | 0, \alpha^{-1}I)$ と仮定することにより, Y の周辺化確率は以下の式のように得られる.

$$p(Y | X, \beta) = \frac{1}{(2\pi)^{\frac{DN}{2}} |K|^{\frac{D}{2}}} \exp\left(-\frac{1}{2} \text{tr}(K^{-1}YY^T)\right) \quad (2)$$

ただし, K は RBF カーネルである. K の要素は

$$k_{n,m} = \alpha \exp\left(-\frac{\gamma}{2}(x_n - x_m)^T(x_n - x_m)\right) + \delta_{nm}\beta^{-1}, \quad (3)$$

で表される. 式 (2) の対数尤度は

$$L = -\frac{DN}{2} \ln(2\pi) - \frac{D}{2} \ln |K| - \frac{1}{2} \text{tr}(K^{-1}YY^T) \quad (4)$$

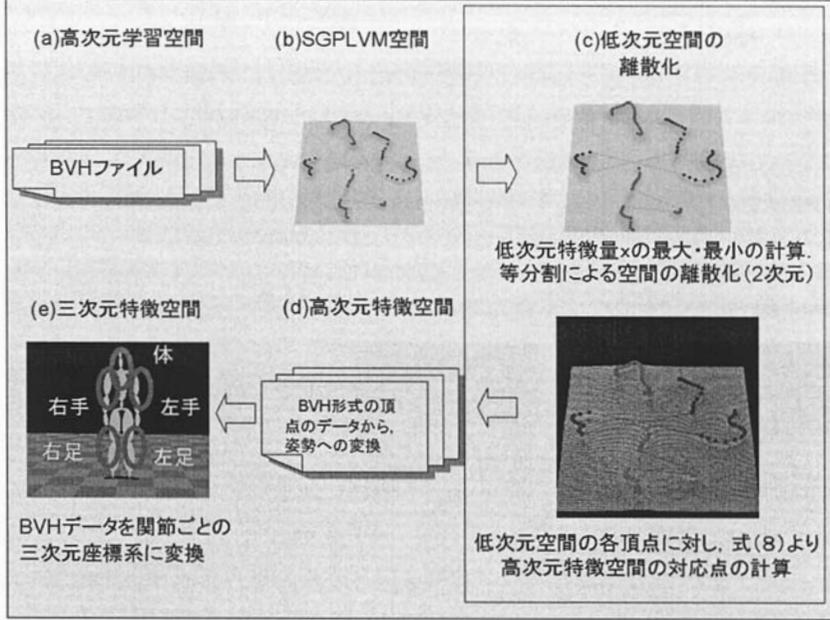


図 2 処理の流れ

と表され、この対数尤度を最適化するために、scaled conjugate gradients(SCG) を用いてパラメータ α, β, γ と潜在変数 x_i を決定する。ただし i は $1 \leq i \leq n$ の学習データ数である。

対数尤度から決定された潜在変数 x_i より、 Y の確率分布は

$$p(Y_I) = \frac{1}{(2\pi)^{\frac{D}{2}} |K_{I,I}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} Y_I^T K_{I,I}^{-1} Y_I\right). \quad (5)$$

と表せる。一方、周辺化された非学習データ y_j に対して、ガウス正規分布により

$$p(y_j | x_j, \alpha, \beta, \gamma) = N(y_j | f_j, \sigma_j^2 I), \quad (6)$$

が得られる。ただし、 j は i と異なり、学習データに対応しないインデックスとする。また $f_j = Y^T K_{I,I}^{-1} k_{I,j}$ 、 $\sigma_j^2 = k(x_j, x_j) - k_{I,j}^T K_{I,I}^{-1} k_{I,j}$ である。ここで、未知の低次元の値 x_j が与えられれば式 (6) の対数尤度を取り、以下の式のように変数 y で微分すると、

$$\frac{\partial L'}{\partial y} = \frac{\partial f'}{\partial x_j} (y - f_j) / \sigma^2(x_j) = 0, \quad (7)$$

$$y_j = f_j, \quad (8)$$

となる。そして未知の x_j に対応する y_j が得られる。ただし $f_j = f_j(x_j)$ である。

この GPLVM の発展手法が Scaled GPLVM [3] であり、異なった変数や次元データに対してもスケールリングにより圧縮が可能になる。

3 姿勢マップ

高次元動作データを扱いやすくするために、SGPLVM による低次元化を行う。そして、対話的な動作生成システムとして、ユーザが低次元空間上で高次元のデータを直感的に理解できる“姿勢マップ”を用いたシステムを提案する。

3.1 SGPLVM による低次元化

ここで、学習データとして BVH 形式を用い、高次 57 次元の特徴量 y_i を SGPLVM により 2 次元の特徴量 x_i に低次元化した。この BVH フォーマットは 18 関節のオイラー角 54 次元とルート位置 3 次元である。

図 1 は、SGPLVM による尤度関数 (写像関数) を 3 動作に対して学習したものである。複数の動作をひとつの低次元空間に写像することにより、同じ特徴空間で議論ができ、動作同士の補間や動作生成が容易になる。図 1 において 3 動作は、歩く、飛び降りる、梯子に登る動作である。X-Y 軸は低次元化した 2 次元の座標値である。Z 軸は $-\frac{D}{2} \ln \sigma^2(x) - \frac{1}{2} \|x_i\|^2$ を表し、学習に使ったデータの尤度が高いほど高くなる。しかし、図 1 ではトレーニングデータに対応する動作姿勢の点しか理解できず、これら以外の点に対する高次元のデータ、すなわちその動作姿勢を直感的に理解することができない。そこで、本論文では直感的に低次元空間の任意の点で人の姿勢を理解するために、姿勢マップを提案する。

3.2 姿勢マップの定義

姿勢マップは、SGPLVM によって低次元化した空間上の各点に対して、体の 5 つの部位ごとに基本姿勢 (図 2(e))

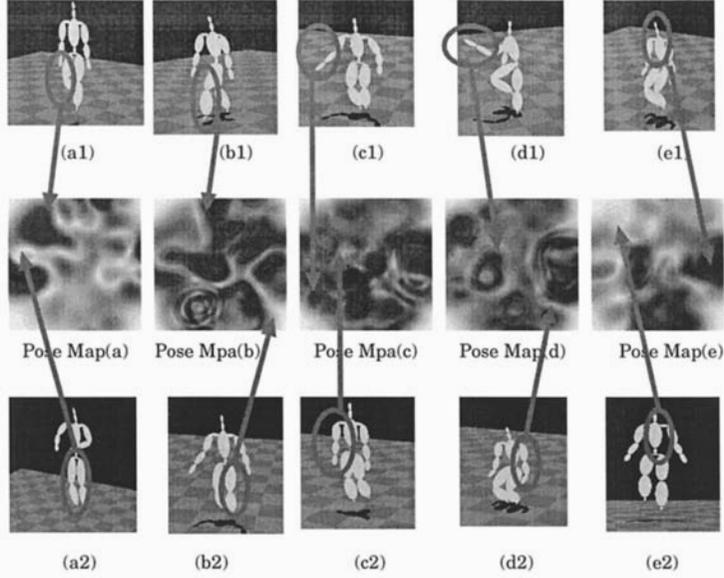


図3 中段列 (a)-(e):姿勢マップ:(a) 右足, (b) 左足, (c) 右腕, (d) 左腕, (e) 体. 暗い領域は基本姿勢と異なる姿勢である. 上段列 (a1)-(e1):それぞれの姿勢は暗い領域の姿勢である. 下段列 (a2)-(e2):それぞれの姿勢は明るい領域の姿勢である.

との差を画像として表したものである. 姿勢マップの生成の流れを図2に示す. まず図2(a), (b)でSGPLVM学習し2(c)のように, SGPLVM空間の x の最大・最小値を求める. さらに, 求めた最大・最小値から2次元空間を 100×100 の格子配列になるように等分割する. そして, 格子配列の各点に対応する高次元空間上の点, すなわちBVH形式のデータを式(8)から求める. 続いて, BVH形式から関節ごとにXYZ座標値 $P'_k(x)$ に変換する. ただし, k は関節を表すインデックスであり, x は格子点座標 (m, n) を表す. 同様に, 基本姿勢図2(e)に対しても P_k を求める. ここで図2(e)のように, 5つの部位 j に分類される関節集合を K_j とし, 基本姿勢の P_{K_j} と算出された姿勢 $P'_{K_j}(x)$ とのユークリッド距離 $d_j(x)$ を部位ごとに以下の式より求める.

$$d_j(x) = \sum_{k \in K_j} \|P_k - P'_k(x)\|^2. \quad (9)$$

ただし, $P_k, P'_k(x)$ のルート位置は原点に固定する. そして, $d_j(x)$ を低次元空間の各座標ごとに計算し, 式(10)より図3のように視覚化する.

$$f_j(x) = \frac{d_j(x) - \min(d_j(x))}{\max(d_j(x) - \min(d_j(x)))} \times 255. \quad (10)$$

図3の中段列では, 圧縮した低次元空間を式(10)による姿勢差を視覚化した結果である. 図3の上段列は暗い領域, つまり基本姿勢と異なる領域の姿勢を表示している. 同様に下段列は明るい領域, つまり基本姿勢に近い姿勢を表示している. 以下にアルゴリズムをまとめる.

1. 図2(e)のような基本姿勢を与える.
2. 基本姿勢を部位(右手, 左手, 体, 右足, 左足)ごとに分割する.
3. SGPLVMより低次元空間の点を高次元に復元し関節ごとに三次元座標を計算する.
4. 復元した座標と基本姿勢との差 $d_j(x)$ を計算する.
5. $d_j(x)$ を視覚化する.

以上のように, 提案する姿勢マップを用いることによって, 直感的に低次元空間で姿勢の理解が可能となる. さらに図3の中段列の画像を5つのレベルで領域分割を行う. 5つのレベルで量子化されたデータを次のように定義した. ラベル”L1”は”かなり異なる”とし一番暗い領域を示す. ”L2”は”異なる”を表し二番目に暗い領域を示す. 同様に”L3”, ”L4”そして”L5”は, それぞれ”少し異なる”, ”類似している”, ”かなり類似している”とする. また”L0”は”どの領域でもよい”である.

3.3 姿勢マップの利用

ユーザが欲しい姿勢をターゲット姿勢とし, 姿勢マップにより求める. ターゲット姿勢は以下のような $\phi = [D1, Bp, x, h]$ により求める. ここで要素D1は5つの量子化した領域(ラベル)を示す. 例えば, ユーザが $D1 = \{L1, L1, L1, L2, L2\}$ のラベルを与えると, 図3から,(a),(b),(c)-mapは”かなり類似している”, (d),(e)-mapは”類似している”の領域を選ぶ. $Bp = 5$ は5段階に領域を分割することとし, x はSGPLVM空間のターゲットとする格子座標である. また, h は学習に用いたデータの姿勢を必要とする時の閾値である. 例えば図1

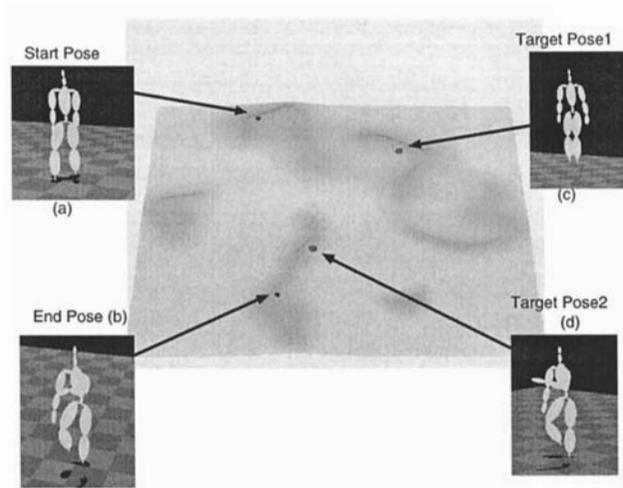


図 4 2D SGPLVM 空間 : (a,b) 最初, 最後の点. (c), は $Dp = \{L1, L1, L1, L2, L2\}$, (d) は $Dp = \{L4, L2, L2, L5, L3\}$ のターゲット点

の尤度が高い領域(暗い領域)のターゲット姿勢が欲しいとき, つまり図 1 の Z 軸の高さを閾値とすることで, ユーザが望むターゲット姿勢を得られる. 低次元空間と姿勢マップのラベルよりターゲット姿勢を以下の目的関数により求める.

$$I(\phi) = \sum_j \delta_{j, D_{L_j}} f_j(x), \quad (11)$$

$$\delta_{j, D_{L_j}} = \begin{cases} 1 & \frac{D_{L_j} \times 255}{B_p} < f_j(x) < \frac{(D_{L_j} + 1) \times 255}{B_p} \\ 0 & \text{指定なし} \end{cases}, \quad (12)$$

$f(x)$ は式 (10) により視覚化された画像の座標である. 例えば, 姿勢マップ図 3(a) は, 右足の基本姿勢からの姿勢差を表しているが, これだけではユーザが欲しいターゲット姿勢を得ることが難しい. そこで複数の部位に分割した姿勢マップを用いることにより, 欲しい姿勢を得られるようにする.

4 実験

図 1 に示すような複数の動作データを用いて所望の動作を生成する. はじめに, ユーザは動作の最初と最後の姿勢を低次元空間上に与える(図 4(a),(b)). これらの最初と最後の点はユーザが手動で与えたり, Inverse GP[8] により SGPLVM 空間上に与えることができる. ターゲット姿勢 x を図 4(c),(d) に示す. ただし, (c) は $DL = \{L1, L1, L1, L2, L2\}$, (d) は $DL = \{L4, L2, L2, L5, L3\}$, $B_p = 5$ である.

これら 4 点を用いて動作を生成するために, 図 4 から単純に直線で補間することで, 低次元空間上の座標系列を求めた. 続いてこれらの点に対する高次元特徴空間上の対応点を求め, この特徴量にもとずいて, 姿勢を決定し動作生成を行った. その結果を図 5 に示す. この動作の表示は MotView[2] を用いた.

5 結論・考察

本論文では, 姿勢マップを用いた対話的な動作生成システムを提案した. 提案システムでは, ターゲットとする姿勢を 2 次元空間上で与え, それらの間を補間・内挿することで動作データを生成することが可能である. しかし, このシステムは姿勢の位置について考慮されていない. そこで, 位置の情報を用いることにより, 動作のテンポなどを考慮したいと考えている. また, ひとつの SGPLVM の低次元空間を用いるのではなく, 複数の低次元空間を用いることも可能である.

姿勢マップと復元関数を保存しておくことで様々な応用が可能と考えられる. SGPLVM 関数だけでは低次元空間での位置や姿勢を理解することは難しいが, 複数の付加された情報を利用することで, 様々な処理が可能になると考えられる. 今後は, 姿勢マップと関数を用いたシステムについて検討する.

参考文献

- [1] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS*, pp. 585–591, 2001.
- [2] A. Gardner and S. Dutcher. Motview v0.8 <http://www.cs.wisc.edu/graphics/courses/cs-838-2000/students/gardner/motview/>, 2000.
- [3] K. Grochow, S. L. Martin, A. Hertzmann, and Z. Popović. Style-based inverse kinematics. *ACM Trans. Graph.*, 23(3):522–531, 2004.
- [4] L. Kovar, M. Gleicher, and F. H. Pighin. Motion graphs. *ACM Trans. Graph.*, 21(3):473–482, 2002.
- [5] N. D. Lawrence. Gaussian process latent variable models for visualisation of high dimensional

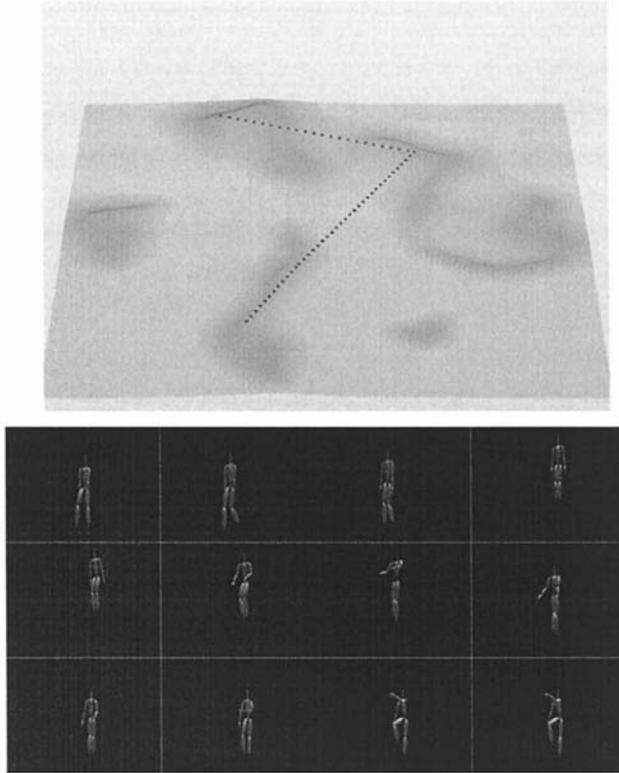


図 5 2D SGPLVM 空間：直線で動作を補間した図である画像：順に左から右にプロットした点の動作である。

- data. In S. Thrun, L. Saul, and B. Schölkopf eds., *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA, 2004.
- [6] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, December 2000.
- [7] H. J. Shin and J. Lee. Motion synthesis and editing in low-dimensional spaces: Research articles. *Comput. Animat. Virtual Worlds*, 17(3‐4):219–227, 2006.
- [8] A. Shon, K. Grochow, A. Hertzmann, and R. Rao. Learning shared latent structure for image synthesis and robotic imitation. In Y. Weiss, B. Schölkopf, and J. Platt eds., *Advances in Neural Information Processing Systems 18*, pp. 1233–1240. MIT Press, Cambridge, MA, 2006.
- [9] M. E. Tipping. Sparse kernel principal component analysis. In *NIPS*, pp. 633–639, 2000.
- [10] M. E. Tipping and C. M. Bishop. Mixtures of probabilistic principal component analysers. *Neural Computation*, 11(2):443–482, 1999.
- [11] R. Urtasun, D. J. Fleet, A. Hertzmann, and P. Fua. Priors for people tracking from small training sets. In *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, pp. 403–410, Washington, DC, USA, 2005. IEEE Computer Society.
- [12] J. M. Wang. Gaussian process dynamical models for human motion. Master's thesis, University of Toronto, 2005.
- [13] J. Wang, D. Fleet, and A. Hertzmann. Gaussian process dynamical models. In Y. Weiss, B. Schölkopf, and J. Platt eds., *Advances in Neural Information Processing Systems 18*, pp. 1441–1448. MIT Press, Cambridge, MA, 2006.
- [14] C. K. I. Williams and C. E. Rasmussen. Gaussian processes for regression. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo. eds., *Advances in Neural Information Processing Systems 8*. MIT Press, 1996.