Supervised Spectral Latent Variable Models

Liefeng Bo[1]
Toyota Technological Institute at Chicago
Chicago, IL 60637
blf0218@tti-c.org

Cristian Sminchisescu[2]
[2]University of Bonn
Bonn, 53115, Germany
sminchisescu.ins.uni-bonn.de

Abstract

We present a probabilistic structured prediction method for learning input-output dependencies where correlations between outputs are modeled as low-dimensional manifolds constrained by both geometric, distance preserving output relations, and predictive power of inputs. Technically this reduces to learning a probabilistic, input conditional model, over latent (manifold) and output variables using an alternation scheme. In one round, we optimize the parameters of an input-driven manifold predictor using latent targets given by preimages (conditional expectations) of the current manifold-to-output model. In the next round, we use the distribution given by the manifold predictor in order to maximize the probability of the outputs with an additional, implicit geometry preserving constraint on the manifold. The resulting *Supervised Spectral Latent Variable Model (SSLVM)* combines the properties of probabilistic geometric manifold learning (accommodates geometric constraints corresponding to any spectral embedding method including PCA, ISOMAP or Laplacian Eigenmaps), with the additional supervisory information to further constrain it for predictive tasks. We demonstrate the superiority of the method over baseline PPCA + regression frameworks and show its potential in difficult real-world computer vision benchmarks designed for the reconstruction of three-dimensional human poses from monocular image sequences.

Appearing in Proceedings of the 12^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2009, Clearwater Beach, Florida, USA. Volume 5 of JMLR: W&CP 5.

1 Introduction

We study structured prediction problems between multivariate inputs and outputs, as arising in computer vision and machine learning problems. In computer vision, the input is an image descriptor and the output is a scene representation, an object shape or a three-dimensional human pose. Both inputs and outputs are high-dimensional and internally correlated. Image features are spatially coherent (nearby pixels often have similar color or edge orientation), whereas outputs are structured due to physical constraints in the world. This can imply a manifold structure, underlying high-dimensional, perceptual representations. While the existence of manifolds has been long since conjectured and effective methods have been derived to detect and recover them from high-dimensional image data (Tenenbaum et al., 2000; Roweis & Saul, 2000; Belkin & Niyogi, 2002), it remains unclear how such representations can be used for visual inference.

One possibility is to endow manifolds with probabilistic formulations that allow mapping between data and intrinsic spaces, or computing probabilities for new datapoints (Sminchisescu & Jepson, 2004; Lawrence, 2005). Additionally, graph-based geometric constraints inspired by spectral non-linear embeddings have also been integrated in a latent variable model in an *unsupervised* setting initially by (Sminchisescu & Jepson, 2004), more recently by (Perpinan & Lu, 2007; Kanaujia et al., 2007; Lu et al., 2007) and subsequently, in a GPLVM formula (Urtasun et al., 2008). Latent variable models of this type have become popular in vision (Lawrence, 2005; Sminchisescu & Jepson, 2004; Urtasun et al., 2005; Urtasun et al., 2008; Wang et al., 2008; Lu et al., 2007; Kanaujia et al., 2007) predominantly as unsupervised intermediate representations, separately linked with images and used for visual inference in conjunction with particle filters (Sminchisescu & Jepson, 2004; Urtasun et al., 2005; Lu et al., 2007) or image-based manifold predictors (Kanaujia et al., 2007). This turned out to be effective but is potentially suboptimal: the manifold discovered using unsupervised learning is not necessarily ideal for prediction or inference. For in-

stance, the variance of the two distributions is by no means calibrated: the noise model of the image-to-manifold predictor could be ways different than the input variance of the manifold-to-output model, negatively impacting the output estimate.

A different approach towards functional manifold models, followed in (Shon et al., 2006; Navaratnam et al., 2007), is based on generative latent variable models with shared latent spaces that link inputs and outputs. These models are consistent but tend to be computationally expensive: training requires estimates of both GP mappings and latent variable coordinates. In addition and perhaps most importantly, the models have very different structure and properties than the ones proposed here: we work in a supervised probabilistic setting with models trained consistently using Maximum Likelihood (hence our ML training cost formally targets a data density model, not only a regularized map to data as in GPLVM-related constructs), we build input conditional models, not joint input-output models as in (Shon et al., 2006; Navaratnam et al., 2007), and we constrain the latent spaces using geometric preserving output relations, none present in models like (Shon et al., 2006; Navaratnam et al., 2007). Our latent space constraints are implicit and expressed in terms of conditional expectations over data points. Hence training complexity is independent of the latent space dimension and allows us to work with large datasets and latent spaces of significantly higher dimensionality than any of the previous methods. This makes it possible to run extensive experiments and reach perhaps unexpected conclusions: in the small sample simpler topology regime local or global geometry preserving constraints equivalent to those of non-linear embeddings tend to be more effective for low-dimensional models that for moderate dimensional ones. The situation is reversed for latent spaces with complex topology and moderate dimensionality where all-distance-preserving constraints, implicit in methods like MDS/PCA, dominate.

Another relevant class of (semi-)supervised techniques is based on the assumption that inputs (rather than outputs, as in our case) have manifold structure: partial least squares and its extensions (sliced inverse regression) (Cook, 1988) recover a linear subspace that is informative for prediction, whereas manifold regression (Nilsson et al., 2007) extends the framework to the non-linear case, using cross-covariance operators. Kernel Dependency Estimation (Weston et al., 2002), in turn, uses kernel PCA to model both inputs and outputs and fits a regression model in the transformed representation. Manifold regularization (Belkin et al., 2005) assumes that inputs have an intrinsic low-dimensional, non-linear geometry, and uses this putative property to diffuse constraints on outputs preferentially based on proximity relations in that geometry. The methods, however, do not consider the application of manifold ideas to the supervised case of structured outputs. Powerful structured prediction methods do exist (Taskar et al., 2004; Tsochantaridis et al., 2004) but these rely on different principles, not on manifold decorrelation methods, as pursued here.

Summarizing, we consider the problem of probabilistic structured prediction using a latent manifold representation, in a supervised setting. The method boils down to constructing an input conditional, probabilistic latent variable model over output and latent (manifold) variables, with a constraint that the geometry of the output distribution (modeled as distance functions between output training data) is preserved in latent space. Structure in the input is modeled by means of sparsity constraints. Overall, this is similar in spirit with (unsupervised) spectral non-linear embedding methods, and any of their underlying implicit geometric constrains, either local or global, can be accommodated. Additionally w.r.t. to spectral latent variable models, the latent manifold is constrained by supervised data via a predictive input to latent map. We are not aware of any model with this structure and properties in the literature. Learning combined, complex models via consistent end-to-end training is notoriously non-trivial (LeCun et al., 1998). Of essence is a structured training method that combines geometry preserving output constraints, input predictive constraints and sparsity in a model that is probabilistically consistent. In this respect our work substantially advances on earlier methods, where components were trained separately (Kanaujia et al., 2007). The result is a flexible probabilistic conditional model, which in our experiments, outperforms separately trained models, as well its linear and non-linear counterparts, or the equivalent unstructured high-dimensional predictors in a difficult real-world computer vision benchmark: the reconstruction of 3d human motions like boxing, gestures, throw and catch, jogging, walking, running—based on latent activity models trained on all motions simultaneously—from video.

2 Supervised Conditional Spectral Latent Variable Model

We work with a (supervised) training set $(\mathbf{r}_i, \mathbf{y}_i), i = 1 \dots N$ with inputs $\mathbf{r}$ and outputs $\mathbf{y}$, both multivariate. We construct a latent variable model with intermediate (hidden) representation $\mathbf{x}$ that preserves geometric constraints among outputs $\mathbf{y}$.

2.1 Conditional Latent Variable Model

The joint distribution over latent and output variables, conditioned on inputs is:

$$p(\mathbf{y}, \mathbf{x}|\mathbf{r}, \boldsymbol{\theta}, \boldsymbol{\delta}) = p(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta}) p(\mathbf{x}|\mathbf{r}, \boldsymbol{\delta}) \quad (1)$$

with $(\boldsymbol{\theta}, \boldsymbol{\delta})$ parameters of the two distributions (in the sequel dropped whenever not essential for readability). The

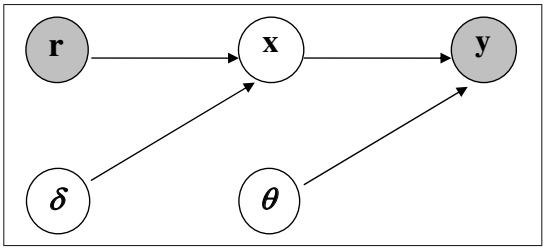

Figure 1: Graphical Model of SSLVM. Shaded nodes indicate observed random variables ($\mathbf{y}$ being observed only in training). We jointly learn two conditional distributions $p(\mathbf{x}|\mathbf{r}) = p(\mathbf{x}|\mathbf{r}, \boldsymbol{\delta})$ and $p(\mathbf{y}|\mathbf{x}) = p(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta})$ with a constraint that the geometry of the outputs, as encoded in distances between datapoints $d(\mathbf{y}_i, \mathbf{y}_j)$ is implicitly preserved among their corresponding latent pre-image expectations $d(\mathrm{E}(\mathbf{x}|\mathbf{y}_i), \mathrm{E}(\mathbf{x}|\mathbf{y}_j))$.

conditional response is calculated by integrating the latent space:

$$p(\mathbf{y}|\mathbf{r}) = \int p(\mathbf{y}|\mathbf{x})p(\mathbf{x}|\mathbf{r})\mathbf{dx} \approx \frac{1}{S}\sum_{s=1}^{S} p(\mathbf{y}|\mathbf{x}^{(s)}) \quad (2)$$

Since we work with non-linear conditional models $p(\mathbf{x}|\mathbf{r})$ and $p(\mathbf{y}|\mathbf{x})$ the integral in (2) cannot be computed analytically. Hence, we approximate using a Monte Carlo estimate based on S samples drawn from the conditional $p(\mathbf{x}|\mathbf{r})$ (Sminchisescu & Welling, 2007).[1] This is tractable and efficient because the latent conditional is usually low-dimensional and has, for our choice of models, a convenient parametric form—either Gaussian for regression or Gaussian mixture in the case of conditional mixtures of experts models.[2] Specifically, we use:

$$p(\mathbf{y}|\mathbf{x}) = p(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta} = (\mathbf{W}, \boldsymbol{\Sigma})) = N(\mathbf{W}\boldsymbol{\Phi}(\mathbf{x}), \boldsymbol{\Sigma}) \quad (3)$$

and

$$p(\mathbf{x}|\mathbf{r}) = p(\mathbf{x}|\mathbf{r}, \boldsymbol{\delta} = (\mathbf{V}_i, \boldsymbol{\rho}_i, \boldsymbol{\Omega}_i)) = \quad (4)$$

$$= \sum_{i=1}^{E} \frac{\exp(\boldsymbol{\rho}_i^\top \mathbf{r})}{\sum_{j=1}^{E}\exp(\boldsymbol{\rho}_j^\top \mathbf{r})} N(\mathbf{V}_i\boldsymbol{\Psi}(\mathbf{r}), \boldsymbol{\Omega}_i) \quad (5)$$

[1] Sampled configurations have parenthesized superscripts; subscripts index training datapoints.

[2] Bayesian mixture of experts adequately account for ambiguity when modeling relations between images and perceptual 3D world representations, *e.g.* the shape of an object or the pose of an articulated figure, either human or animal. Here, $3d \rightarrow 2d$ projection can be satisfactorily modeled as a non-linear mapping, but its inverse, the $3d \leftarrow 2d$ relation is usually not. Similar image features often correspond to very different 3d percepts, hence we need multivalued models that can offer plausible alternative interpretations, rather than average them, as would do *e.g.* a regressor or any function approximator. We wish ambiguous inputs to be resolved by multiple competitive experts, and unambiguous ones resolved by singletons.

with N Gaussian functions, $\boldsymbol{\Phi}, \boldsymbol{\Psi}$ kernels, and softmax functions for the gates of the experts. Eq. (4) covers the case of single-valued regression models (with 1 expert). Models in (3) and (4) are made computationally efficient and more robust to overfitting by using hierarchical priors on parameters $\mathbf{W}, \mathbf{V}, \boldsymbol{\rho}$. These are controlled by a second level of Gamma distributions and are trained using ML type II with forward selection, in order to select a sparse input subset for prediction—see (Tipping, 2001; Mackay, 1998; Bo et al., 2008) for details.

The latent space conditional is obtained using Bayes' rule:

$$p(\mathbf{x}|\mathbf{y}, \mathbf{r}) = \frac{p(\mathbf{y}|\mathbf{x})p(\mathbf{x}|\mathbf{r})}{p(\mathbf{y}|\mathbf{r})} = \quad (6)$$

$$= \frac{p(\mathbf{y}|\mathbf{x})p(\mathbf{x}|\mathbf{r})}{\frac{1}{S}\sum_{s=1}^{S} p(\mathbf{y}|\mathbf{x}^{(s)})} \quad (7)$$

For pairs of training data i and MC latent samples s, we abbreviate $p_{(s,i)} = p(\mathbf{x}^{(s)}|\mathbf{y}_i, \mathbf{r}_i)$. Notice how the choice of latent conditional $p(\mathbf{x}|\mathbf{r})$ influences the membership probabilities in (6). We can compute either the conditional mean or the mode (better for multimodal distributions) in latent space, using the same MC integration method used for (2):

$$\mathrm{E}\{\mathbf{x}|\mathbf{y}_n, \mathbf{r}_n\} = \int p(\mathbf{x}|\mathbf{y}_n, \mathbf{r}_n)\mathbf{x}\mathbf{dx} \quad (8)$$

$$= \sum_{s=1}^{S} p_{(s,n)}\mathbf{x}_s \quad (9)$$

$$s_{max} = \arg\max_s p_{(s,n)} \quad (10)$$

The model has the components for consistent calculations in both the latent and output spaces: (4) computes the latent space distribution, (2) the output marginal, (3) provides the conditional (or mapping) from latent to output, and (8) and (10) give the mean or mode of the mapping from output to latent space. More accurate but also more expensive mode-finding approximations can be obtained by direct gradient ascent on (6) (we will not use these, for now). Latent conditionals given partially observed $\mathbf{y}$ vectors are easy to compute, using (6). The distribution on $\mathbf{y}$ is Gaussian and unobserved components can be integrated analytically – this effectively removes them from the mean and the corresponding lines and columns of the covariance. Computations like these are useful as often outputs can have missing entries, *e.g.* marker drop-outs in a motion capture system during training, 'pattern completion' or restoration of an image under the latent model during testing, *etc.*

3 Implicit Latent Geometric Constraints

Assume that distances between outputs $\mathbf{y}$ are stored in a vector $\mathbf{D}$ of size N^2, with entries $d(\mathbf{y}_i, \mathbf{y}_j)$ with d an arbitrary similarity function that can be the Euclidean distance, a Gaussian centered at the first argument, or a geodesic

distance in the data graph (these will be used to model constraints like the ones encountered in PCA/MDS, Laplacian Eigenmaps or ISOMAP, respectively). Consider a similar vector $\mathbf{L}$ of corresponding latent space distances $d(\mathrm{E}(\mathbf{x}|\mathbf{y}_i), \mathrm{E}(\mathbf{x}|\mathbf{y}_j))$, where $\mathrm{E}(\mathbf{x}|\mathbf{y})$ is the conditional expectation of latent variable $\mathbf{x}$ given $\mathbf{y}$, *c.f.* (8). We use vectors of pairwise distances among outputs and their corresponding latent conditional expectations in order to construct an implicit geometric constraint (or penalty) in latent space:

$$C = (\mathbf{D} - \mathbf{L})(\mathbf{D} - \mathbf{L})^\top \tag{11}$$

which is 0 if output distances are preserved in latent space and large otherwise. Notice that $d(\mathbf{x}_i, \mathbf{x}_j)$ gives the distance between the i-th and j-th point in data or latent space, rather than the relative spatial positions of points in data and latent space. Clearly $d(\mathbf{x}, \mathbf{x}) = 0$. However, our method does not explicitly require distance properties. We can work, in principle, with any similarity measure such as the inner product. Notice also the dependence of the penalty on $\mathrm{E}(\mathbf{x}|\mathbf{y}) = \mathrm{E}(\mathbf{x}|\mathbf{y}, \boldsymbol{\theta})$, which is a function of the current parameters $\boldsymbol{\theta}$ of the latent to output model $p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x})$, *c.f.* (3), (6) and (8). The penalty ensures that under the latent space posterior, the implicit latent space distances $\mathrm{E}(\mathbf{x}|\mathbf{y}_i)$ among configurations $\mathbf{x}$ that correspond to datapoints $\mathbf{y}_i$ under the current model $\boldsymbol{\theta}$, are preserved (distances $d(\mathbf{y}_i, \mathbf{y}_j)$). Notice that no matter what spectral constraint is used, the resulting model is highly non-linear: latent variables depend non-linearly on inputs and and outputs depend non-linearly on latent the variables.

The *implicit geometric constraint regularizer* in (11) requires the calculation of conditional expectations for the latent variables given output data, which may at first appear more complex. Notice that the regulizer does not depend explicitly on latent variables, hence we do not optimize latent variables explicitly – this would be underconstrained, prone to overfitting, and computationally prohibitive for models with more than a few latent dimensions. Furthermore, the proposed expression can be approximated efficiently by considering the inner product as a form distance function:

$$d(\mathrm{E}(\mathbf{x}|\mathbf{y}_i)\, \mathrm{E}(\mathbf{x}|\mathbf{y}_j)) = \sum_{s=1}^{S}\sum_{t=1}^{S} p_{(s,i)} p_{(t,j)} \mathbf{x}_s^\top \mathbf{x}_t \tag{12}$$

Typically, most $p_{(s,i)}$ will be close to zero. Removing those does not reduce the evaluation of distance functions but offers large speedups when computing its derivative, since $\mathbf{x}_s^\top \mathbf{x}_t$ can be stored ahead of time.

3.1 Learning Algorithm

We learn the conditional model in (1) by optimizing a penalized likelihood criterion that consists of the marginal likelihood (2) averaged over a dataset and the geometric penalty on the latent space (11):

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\delta}) = \log \prod_{i=1}^{N} p(\mathbf{y}_i|\mathbf{r}_i) - \lambda C = \tag{13}$$

$$= \sum_{i=1}^{N} \log p(\mathbf{y}_i|\mathbf{r}_i) - \lambda C = \tag{14}$$

$$= \sum_{i=1}^{N} \log \sum_{s=1}^{S} p(\mathbf{y}_i|\mathbf{x}^{(s)}, \mathbf{r}_i) - \lambda C \tag{15}$$

and λ is the regularizing (trade-off) parameter. In practice we will optimize a cost $\mathcal{F}$ that is the penalized expectation of the complete data log-likelihood $\mathcal{L}_c$:

$$\mathcal{F} = < \mathcal{L}_c(\boldsymbol{\theta}, \boldsymbol{\delta}) > - \lambda C = \tag{16}$$

$$= \sum_{i=1}^{N}\sum_{s=1}^{S} p_{(s,i)} \log p(\mathbf{y}_i|\mathbf{x}^{(s)}) - \lambda C \tag{17}$$

We train the model by estimating $p(\mathbf{x}|\mathbf{r})$ and $p(\mathbf{y}|\mathbf{x})$ in alternation. We initialize the latent coordinates $\mathbf{x}_i$ corresponding to the given output data $\mathbf{y}_i$ using dimensionality reduction, based on the type of geometric constraint we wish to impose, *e.g.* PCA, ISOMAP or Laplacian Eigenmaps, and we use their corresponding distances between datapoints $d(\mathbf{y}_i, \mathbf{y}_j)$ in the penalty term C, see (11). Then, we first train $p(\mathbf{x}|\mathbf{r})$ based on $(\mathbf{r}_i, \mathbf{x}_i)$ data, and $p(\mathbf{y}, \mathbf{x})$ by sampling from the learned model $p(\mathbf{x}|\mathbf{r})$ with target data $\mathbf{y}_i$ and the constraint C. Then, we alternate between generating data $(\mathrm{E}(\mathbf{x}|\mathbf{y}_i), \mathbf{r}_i)$ for training the input model $p(\mathbf{x}|\mathbf{r})$, and training the output model $p(\mathbf{y}|\mathbf{x})$ using EM: in the *E-step* we estimate the membership probabilities *c.f.* (6), and in the *M-step* we solve a penalized weighted regression problem as in (16), with weights given by (6) and penalty given by C (11). Notice that C changes since $\mathrm{E}(\mathbf{x}|\mathbf{y})$ is a function of the current $\boldsymbol{\theta} = (\mathbf{W}, \boldsymbol{\Sigma})$ parameters of $p(\mathbf{y}|\mathbf{x})$, *c.f.* (8) and (3). The procedure is summarized as **Algorithm 1**.

4 Experiments

To illustrate the performance of our models, we analyze the HumanEva dataset (Sigal & Black, 2006)[3] which contains a number of sequences of walking, jogging, throw-catch, gestures, and boxing, capture from 3 subjects, for a total of 5942 training samples and 5832 test samples (the backgrounds are known and fairly uniform, hence silhouettes and their bounding boxes can be computed), acquired with a human motion capture system. The training set consists of pairs of human silhouette image bounding box descriptors and human pose information. 9-d histograms of gradient orientations 0–180^o are computed on a regular grid

[3] For complementary structured prediction models and experiments, see our companion papers (Bo et al., 2008; Bo & Sminchisescu, 2009a; Bo & Sminchisescu, 2009b).

Motion	RVM	PCA	SLVM PCA	SLVM ISOMAP	SLVM LE	PPCA	SSLVM PCA	SSLVM ISOMAP	SSLVM LE
Box	76.8	78.4	75.1	71.2	72.2	77.2	71.3	64.8	61.2
Jog	57.5	57.3	56.2	53.8	57.8	56.8	48.4	46.1	51.1
Walking	50.7	50.2	47.3	44.5	52.2	50.3	40.5	36.2	43.3

Table 1: Comparisons of prediction error (mm) for different models and different motions of a single subject (monocular video). All models use latent space predictors based on RVM. The dimensionality of latent space is 8. The model under RVM is the full-dimensional model. Models other than SSLVM (with spectral constraints PCA,ISOMAP, Laplacian Eigenmaps LE) are trained separately by combining a dimensionality reduction method with a latent (low-dimensional) predictor. PPCA-RVM learns a linear latent variable model based on an RVM latent space posterior—in this model, the latent variables can be integrated analytically.

Algorithm 1 . Supervised Spectral Latent Variable Model

Input: A training set $(\mathbf{r}_i, \mathbf{y}_i), i = 1 \dots N$ with multivariate input / output.

Output: Parameters $(\boldsymbol{\theta}, \boldsymbol{\delta})$ of a conditional latent variable model $p(\mathbf{y}, \mathbf{x}|\mathbf{r}, \boldsymbol{\theta}, \boldsymbol{\delta})$ with latent space $\mathbf{x}$ that preserves geometric constraints in the output distribution given by data $\mathbf{y}_i$.

1. **Initialize** latent variables $\mathbf{x}$ using a spectral embedding of $\mathbf{y}$ based on local/global distance functions $d(\mathbf{y}_i, \mathbf{y}_j)$ computed as in PCA, ISOMAP, Laplacian Eigenmaps, *etc*.
2. **Optimize by alternation.** *(Stage 1) Train the conditional* $p(\mathbf{x}|\mathbf{r}, \boldsymbol{\delta})$ using inputs $\mathbf{r}$ and the implicit latent variables $(\mathbf{r}_i, \mathbf{x}_i)$. The model is generally a Bayesian mixture of experts (BME) (4), but can 'degenerate' to one component, a sparse Bayesian regressor (RVM).
3. Sample latent variables $\mathbf{x}^s, s = 1 \dots S$, from conditional model $p(\mathbf{x}|\mathbf{r}, \boldsymbol{\delta})$, to obtain the Monte-Carlo estimate in (2), in order to train $p(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta})$.
4. *(Stage 2) Train the conditional* $p(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta})$, a Gaussian non-linear regression model (3), by maximizing the penalized complete log-likelihood (16), using Expectation-Maximization (EM): In the *E-step*, use the current $\boldsymbol{\theta}$ to evaluate the posterior probabilities $p(\mathbf{x}^{(s)}|\mathbf{y}_i, \mathbf{r}_i)$ of each datapoint i, *c.f*. (8). In the *M-step*, update parameters $\boldsymbol{\theta}$ by maximizing the complete penalized log-likelihood (16).
5. Update the latent variables $\mathbf{x}_i \leftarrow \mathrm{E}(\mathbf{x}|\mathbf{y}_i)$ according to (8) using $\boldsymbol{\theta}$ from Step 4.
6. If converged or maximum iteration reached, Stop; otherwise go to Step 2.

inside the silhouette bounding box and concatenated in a descriptor vector of size 270. Human poses are represented as 60d vectors of three-dimensional body joint positions. All poses are preprocessed by subtracting the root joint location from all the joint centers at every timeframe (This doesn't influence the error in our fairly extensively experiments. The root position is very well controlled by the silhouette, which is well localized, spatially, in the image, and by the other joints close to the root of the body, *e.g*. hip, torso). Hyperparameters such as λ (strength of regularizer) are estimated by five-fold cross validation (training set). The prediction error is the Euclidean distance between the estimated joint center and the true joint center averaged over all joints, per frame (Sigal & Black, 2006):

$$\mathrm{Err}_{seq} = \frac{1}{T}\sum_{i=1}^{T} D(\mathbf{y}_i, \overline{\mathbf{y}}_i) \tag{18}$$

where T is the length of sequence and

$$D(\mathbf{y}_i, \overline{\mathbf{y}}_i) = \frac{1}{M}\sum_{j=1}^{M} \|m_j(\overline{\mathbf{y}}_i) - m_j(\mathbf{y}_i)\| \tag{19}$$

where $m_j(\mathbf{y}_i) \in \mathbb{R}^3$ is a function which extracts the three dimensional coordinates of the jth joint center, M is the number of the joint centers for each pose and $\|\cdot\|$ is the Euclidean distance. The prediction for output is achieved, as usual, via integration, *c.f*. (2).

We compare our Supervised Spectral Latent Variable Model (SSLVM) with several competing methods: high-dimensional image-pose predictors (Relevance Vector Machine, RVM, Bayesian Mixture of Experts, BME), principle component analysis and its probabilistic versions (PPCA) (Tipping & Bishop, 1999) and Spectral Latent Variable Models (SLVM) (Kanaujia et al., 2007) (initialized with PCA, ISOMAP, or Lalacian Eigenmaps) where the latent variable model and the latent predictor are trained separately. We also test SSLVM with different implicit latent geometric constraints (distance functions equivalent to PCA, ISOMAP or Laplacian Eigenmaps constraints) and predictors both univalued and multivalued (SSLVM-RVM, SSLVM-BME). We train models both on different motions of a single subject and on combined motions from different subjects. We have also run experiments where no distance preserving penalty was used: all models trained in this way consistently gave 5–10% higher error.

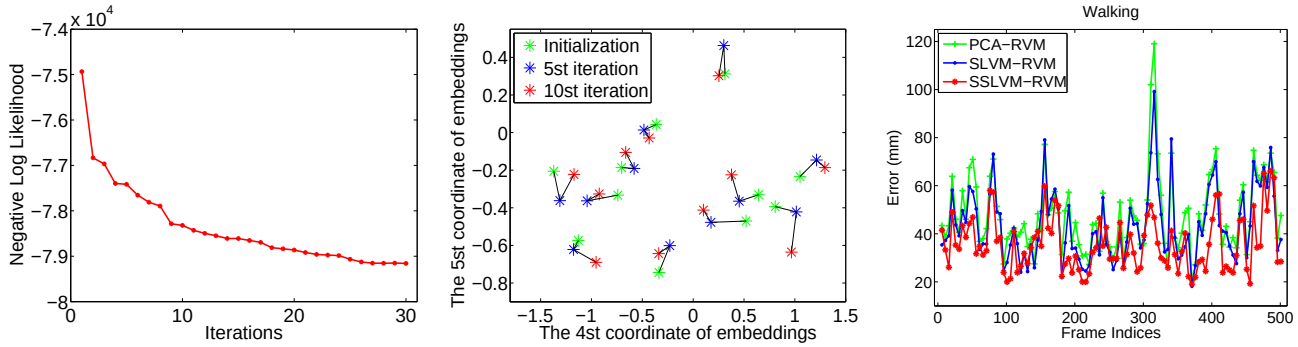

Figure 2: Analysis of walking (models with 8d latent spaces). *Left:* Model negative log likelihood function of iteration. *Middle:* The 4th and 5th latent variable of an SSLVM-RVM sampled at several iterations during training. *Right:* Comparative prediction error, per frame for several models.

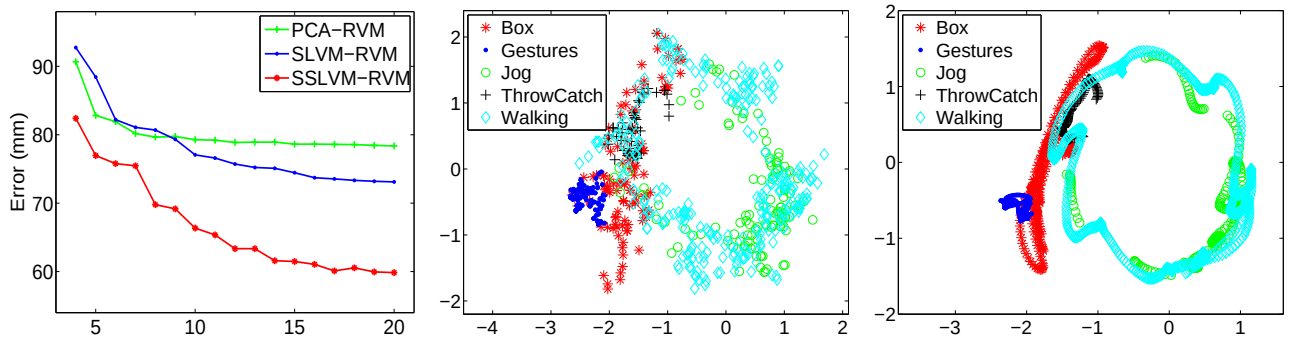

Figure 3: Analysis of data from Subject 1, models trained on multiple motions. *Left:* Average prediction error function of dimension shows that the gap between SSLVM and separately trained models increases with dimension. *Middle and right:* 2d projections of a 16d latent space of SSLVM-RVM and a PCA embedding, corresponding, respectively, to the same motion data *(right)*, show that the implicit latent coordinates of SSLVM have larger variance compared to (unsupervised) PCA. This compensates for the uncertainty in the image-based manifold predictor and substantially improves end-to-end output data model.

We investigate latent variable models based on PCA, ISOMAP and Laplacian Eigenmaps constraints. We use 8d and 16d latent spaces because these are the lowest latent dimensions that gives good performance for all models. The primary goal in selecting the model dimensionality was accurate prediction, not only the generation of 2d and 3d images for visualization. But see also our fig. 3, left, where multiple models corresponding to the full range of dimensions 2...20 are tested.

In tables 1 and 2, we report prediction error for several dimension reduction methods and a baseline high-dimensional RVM predictor. SSLVM-RVM consistently outperforms other dimension reduction methods and achieves about 10mm improvement over an RVM without dimension reduction. The higher errors observed for multiple activities (walking, jogging, throw-catch, gestures and boxing) are most likely caused by increased ambiguity in models that are learned jointly, using diverse data, as opposed to data from just one activity (to be expected). In fig. 2, left, we show the negative log-likelihood as function of the number of iterations in the alternation algorithm, where the implicit latent variables (conditional expectations of the output data) are sampled at several iterations. In fig. 2, middle, we show how the latent space expectations of datapoins change in order to account for the uncertainty of the image-based manifold predictor. The alternation algorithm (Algorithm 1) jointly callibrates the image-based manifold predictor and the manifold-to-output model. Fig. 3 shows prediction error for different models with dimensions 2...20 and gives insight into the latent spaces learned by SSLVMs trained on different motions. The embedding of SSLVM is very different from PCA. In both cases we show the conditional latent space expectations of a non-linear model with 'all distance preserving constraints' (type MDS/PCA) before and after SSLVM training (rightmost plot shows original expectations for models trained separately, latent variable model, then predictor to the latent variable; the middle plot shows the conditional latent space expectations after learning. This shows how the joint training calibrates the two subcompo-

nents of the model. In table 3, we report the prediction error of several dimension reduction methods based on more sophisticated mixture of experts (BME) image predictors (notice improvement over RVM). Models based on all pair-wise distance preserving constraints (type MDS/PCA) work better than those based on ISOMAP, LLE-style constraints in this case, as table 2 suggests. The best performing latent variable models SLVM-PCA and SSLVM-PCA are compared in table 3.

We have also obtained results using HumanEva's online evaluation system. Results are given in fig.2, left. The average joint error of SSLVM-RVM with ISOMAP-style global geometry preserving constraints is 48.4mm and 52.9mm for Walking and Jogging, respectively. This is significantly lower than PCA-RVM (58.5mm for Walking and 62.8mm for Jogging) and the separately trained SLVM-RVM with ISOMAP constraints (55.4mm for Walking and 59.4mm for Jogging).

We notice that models with latent spaces constrained by penalties built in terms of distances between all training datapoints (constraints similar to PCA/MDS) tend to be outperformed by more sophisticated distance functions, similar to the ones of ISOMAP or Laplacian Eigenmaps, for models trained on separate motions, see table 1. The situation is reversed for higher-dimensional models trained on different motions, see *e.g*. table 2. These findings do not warrant, in our view, a 'return to linear/PCA' type conclusion – no matter the spectral constraint used, the resulting SSLVM models are highly non-linear. However, supervised non-linear probabilistic models with latent spaces that preserve all distances between datapoints appear to be practically more suitable, at least in the small sample, complex topology, moderate dimension regime, which results when data from multiple motions is combined for training. Indeed, we do not yet appear to have enough data to observe coherent manifold structure (surfaces as opposed to occasionally intersecting curves) when models for multiple motions are learned jointly. Hence local distance preserving constraints (or global constraints computed in terms of local ones, similar to ISOMAP) may loose their effectiveness. For significantly larger datasets the situation may turn different. In any case, our experiments strongly suggest that distance preserving constraints and joint training of supervised latent variable predictors is very effective and consistently outperforms baseline competitors.

5 Conclusions

We have presented an input-output probabilistic structured prediction method that models correlations by means of latent manifolds constrained by both geometric relations among outputs (as in non-linear embedding methods) and by the predictive uncertainty of an input-to-latent space predictor. Structure in the input, in turn is modeled via sparsity constraints. The result is a structured, flexible probabilistic input conditional model–the Supervised Spectral Latent Variable Model—that combines unsupervised and supervised components consistently. The model outperforms a number of linear or unstructured competitors, and offers accurate predictions for difficult computer vision problems like the three-dimensional reconstruction of human motion from monocular video. Hierarchical, semi-supervised and temporal extensions of the model are currently considered.

Acknowledgements

This work was supported, in part, by the EC and the NSF, under awards MCEXT-025481 and IIS-0535140.

References

Belkin, M., & Niyogi, P. (2002). Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering. *Advances in Neural Information Processing Systems*.

Belkin, M., Niyogi, P., & Sindhwani, V. (2005). On manifold regularization. *Artificial Intelligence and Statistics*.

Bishop, C., Svensen, M., & Williams, C. K. I. (1998). GTM: The Generative Topographic Mapping. *Neural Computation*, 215–234.

Bo, L., & Sminchisescu, C. (2009a). Structured Output-Associative Regression. *IEEE International Conference on Computer Vision and Pattern Recognition*.

Bo, L., & Sminchisescu, C. (2009b). Twin Gaussian Processes for Structured Prediction. *International Journal of Computer Vision*. Special Issue on Evaluation of Articulated Human Motion and Pose Estimation.

Bo, L., Sminchisescu, C., Kanaujia, A., & Metaxas, D. (2008). Fast algorithms for large scale conditional 3d prediction. *IEEE International Conference on Computer Vision and Pattern Recognition*.

Cook, R. D. (1988). *Regression Graphics*. Wiley Inter-Science.

Kanaujia, A., Sminchisescu, C., & Metaxas, D. (2007). Spectral latent variable models for perceptual inference. *IEEE International Conference on Computer Vision*.

Lawrence, N. (2005). Probabilistic non-linear component analysis with gaussian process latent variable models. *Journal of Machine Learning Research*, 1783–1816.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proc. of IEEE*.

Lu, Z., Perpinan, M. C., & Sminchisescu, C. (2007). People Tracking with the Laplacian Eigenmaps Latent Variable Model. *Advances in Neural Information Processing Systems*.

Mackay, D. (1998). Comparison of Approximate Methods for Handling Hyperparameters. *Neural Computation*, *11*.

Memisevic, R. (2006). Kernel Information Embeddings. *International Conference on Machine Learning*.

Subject	RVM	PCA	SLVM PCA	SLVM ISOMAP	SLVM LE	PPCA	SSLVM PCA	SSLVM ISOMAP	SSLVM LE
Subject1	78.4	78.6	72.9	76.5	76.1	78.8	61.4	66.2	67.2
Subject2	86.6	87.1	84.8	85.8	82.5	86.3	74.0	76.5	72.4
Subject3	95.2	96.0	90.2	93.6	92.3	95.6	73.9	82.5	79.6

Table 2: Prediction error [mm] for different models learned for each subject, using data from multiple motions: box, gestures, jog, throw and catch and walking. The model labeled as RVM is the full-dimensional model where dimensions are predicted independently. Models other than SSLVM (and based on spectral constraints PCA, ISOMAP, Laplacian Eigenmaps LE) are trained separately by combining independently trained latent variable models and a latent (low-dimensional) predictors. The dimensionality of the latent space is 16.

Subject	BME	PCA	SLVM	PPCA	SSLVM
Subject 1	53.4	54.9	53.2	54.2	50.3
Subject 2	65.0	69.7	65.8	69.1	60.8
Subject 3	66.1	70.8	67.4	69.6	61.5

Table 3: Prediction error (mm) per 3d body joint, for models trained on different subjects and five motions: box, gestures, jog, throw and catch and walking. Dimensionality of the latent space is 16. BME is the high-dimensional model with no dimension reduction, but latent variable models use BME for the image to latent map. We compare SLVM models where the latent variable representation and latent predictor are trained separately and SSLVM models (with PCA constraints) where these are trained jointly. The model based on PPCA learns a latent variable model with linear mapping from latent to output space—in this model latent variables can be integrated analytically.

Navaratnam, R., Fitzgibbon, A. W., & Cipolla, R. (2007). The joint manifold model for semi-supervised multi-valued regression. *IEEE International Conference on Computer Vision* (pp. 1–8).

Nilsson, J., Sha, F., & Jordan, M. I. (2007). Regression on Manifolds Using Kernel Dimensionality Reduction. *International Conference on Machine Learning*.

Perpinan, M. C., & Lu, Z. (2007). The Laplacian Eigenmaps Latent Variable Model. *Artificial Intelligence and Statistics*.

Roweis, S., & Saul, L. (2000). Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*.

Shon, A. P., Grochow, K., Hertzmann, A., & Rao, R. (2006). Learning shared latent structure for image synthesis and robotic imitation. *Advances in Neural Information Processing Systems*.

Sigal, L., & Black, M. (2006). *HumanEva: Synchronized Video and Motion Capture Dataset for Evaluation of Articulated Human Motion* (Technical Report CS-06-08). Brown University.

Sminchisescu, C., & Jepson, A. (2004). Generative Modeling for Continuous Non-Linearly Embedded Visual Inference. *International Conference on Machine Learning* (pp. 759–766). Banff.

Sminchisescu, C., & Welling, M. (2007). Generalized Darting Monte-Carlo. *Artificial Intelligence and Statistics*.

Taskar, B., Guestrin, C., & Koller, D. (2004). Max-margin Markov networks. *Advances in Neural Information Processing Systems*.

Tenenbaum, J., Silva, V., & Langford, J. (2000). A Global Geometric Framewok for Nonlinear Dimensionality Reduction. *Science*.

Tipping, M. (2001). Sparse Bayesian learning and the Relevance Vector Machine. *Journal of Machine Learning Research*.

Tipping, M. E., & Bishop, C. M. (1999). Probabilistic principal component analysis. *Journal Of The Royal Statistical Society Series B*, *61*, 611–622.

Tsochantaridis, I., Hofmann, T., Joachims, T., & Altun, Y. (2004). Support vector machine learning for interdependent and structured output spaces. *International Conference on Machine Learning*.

Urtasun, R., Fleet, D., Geiger, A., Popovic, J., Darrell, T., & Lawrence, N. (2008). Topologically-constrained latent variable models. *International Conference on Machine Learning* (pp. 1080–1087).

Urtasun, R., Fleet, D., Hertzmann, A., & Fua, P. (2005). Priors for people tracking in small training sets. *IEEE International Conference on Computer Vision*.

Wang, J., Fleet, D. J., & Hertzmann, A. (2008). Gaussian Process Dynamical Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Weston, J., Chapelle, O., Elisseeff, A., Scholkopf, B., & Vapnik, V. (2002). Kernel dependency estimation. *Advances in Neural Information Processing Systems*.