Learning to Track 3D Human Motion from Silhouettes

Ankur Agarwal, Bill Triggs

Learning to Track 3D Human Motion from Silhouettes

Ankur Agarwal ANKUR.AGARWAL@INRIALPES.FR
Bill Triggs BILL.TRIGGS@INRIALPES.FR
GRAVIR-INRIA-CNRS, 655 Avenue de l'Europe, Montbonnot 38330, France

Abstract

We describe a sparse Bayesian regression method for recovering 3D human body motion directly from silhouettes extracted from monocular video sequences. No detailed body shape model is needed, and realism is ensured by training on real human motion capture data. The tracker estimates 3D body pose by using Relevance Vector Machine regression to combine a learned autoregressive dynamical model with robust shape descriptors extracted automatically from image silhouettes. We studied several different combination methods, the most effective being to learn a nonlinear observation-update correction based on joint regression with respect to the predicted state and the observations. We demonstrate the method on a 54-parameter full body pose model, both quantitatively using motion capture based test sequences, and qualitatively on a test video sequence.

1. Introduction

We consider the problem of estimating and tracking the 3D configurations of complex articulated objects from monocular images, *e.g.* for applications requiring 3D human body pose or hand gesture analysis. There are two main schools of thought on this. *Model-based approaches* presuppose an explicitly known parametric body model, and estimate the pose by either: *(i)* directly inverting the kinematics, which requires known image positions for each body part (Taylor, 2000); or *(ii)* numerically optimizing some form of model-image correspondence metric over the pose variables, using a forward rendering model to predict the images, which is expensive and requires a good initialization, and the problem always has many local minima (Sminchisescu & Triggs, 2003). An important sub-case is *model-based tracking*, which focuses on tracking the pose estimate from one time step to the next starting from a known initialization, based on an approximate dynamical model (Bregler & Malik, 1998, Sidenbladh et al., 2002). In contrast, *learning based approaches* try to avoid the need for accurate 3D modelling and rendering, and to capitalize on the fact that the set of *typical* human poses is far smaller than the set of kinematically possible ones, by estimating (learning) a model that directly recovers pose estimates from observable image quantities (Grauman et al., 2003). In particular, *example based methods* explicitly store a set of training examples whose 3D poses are known, and estimate pose by searching for training image(s) similar to the given input image, and interpolating from their poses (Athitsos & Sclaroff, 2003, Stenger et al., 2003, Mori & Malik, 2002, Shakhnarovich et al., 2003).

In this paper we take a learning based approach, but instead of explicitly storing and searching for similar training examples, we use sparse Bayesian nonlinear regression to distill a large training database into a single compact model that generalizes well to unseen examples. We regress the current pose (body joint angles) against both image descriptors (silhouette shape) and a pose estimate computed from previous poses using a learned dynamical model. High dimensionality and the intrinsic ambiguity in recovering pose from monocular observations makes the regression nontrivial. Our algorithm can be related to probabilistic tracking, but we eliminate the need for: *(i)* an exact body model that must be projected to predict an image; and *(ii)* a predefined error model to evaluate the likelihood of the observed image signal given this projection. Instead, pose is estimated directly, by regressing it against a dynamics-based prediction and an observed shape descriptor vector. Regressing on shape descriptors allows appearance variations to be learned automatically, enabling us to work with a simple generic articular skeleton model; while including an estimate of the pose in the regression allows the method to overcome the inherent many-to-one projection ambiguities present in monocular image observations.

Our strategy makes good use of the sparsity and generalization properties of our nonlinear regressor, which is a variant of the *Relevance Vector Machine (RVM)* (Tipping, 2000).

Appearing in *Proceedings of the 21^{st} International Conference on Machine Learning*, Banff, Canada, 2004.

RVM's have been used, *e.g.*, to build kernel regressors for 2D displacement updates in correlation-based patch tracking (Williams et al., 2003). Human pose recovery is significantly harder — more ill-conditioned and nonlinear, and much higher dimensional — but by selecting a sufficiently rich set of image descriptors, it turns out that we can still obtain enough information for successful regression (Agarwal & Triggs, 2004a).

Our motion capture based training data models each joint as a spherical one, so formally, we represent 3D body pose by 55-D vectors $\mathbf{x}$ including 3 joint angles for each of the 18 major body joints. The input images are reduced to 100-D observation vectors $\mathbf{z}$ that robustly encode the shape of a human image silhouette. Given a temporal sequence of observations $\mathbf{z}_t$, the goal is to estimate the corresponding sequence of pose vectors $\mathbf{x}_t$. We work as follows: At each time step, we obtain an approximate preliminary pose estimate $\check{\mathbf{x}}_t$ from the previous two pose vectors, using a dynamical model learned by linear least squares regression. We then update this to take account of the observations $\mathbf{z}_t$ using a joint RVM regression over $\check{\mathbf{x}}_t$ and $\mathbf{z}_t$ — $\mathbf{x} = \mathbf{r}(\check{\mathbf{x}}, \mathbf{z})$ — learned from a set of labelled training examples $\{(\mathbf{z}_i, \mathbf{x}_i) \,|\, i = 1 \ldots n\}$. The regressor is a linear combination $\mathbf{r}(\mathbf{x}, \mathbf{z}) \equiv \sum_k \mathbf{a}_k \, \phi_k(\mathbf{x}, \mathbf{z})$ of prespecified scalar basis functions $\{\phi_k(\mathbf{x}, \mathbf{z}) \,|\, k = 1 \ldots p\}$ (here, instantiated Gaussian kernels). The learned regressor is regular in the sense that the weight vectors $\mathbf{a}_k$ are well-damped to control over-fitting, and sparse in the sense that many of them are zero. Sparsity occurs because the RVM actively selects only the 'most relevant' basis functions — the ones that really need to have nonzero coefficients to complete the regression successfully.

Previous work: There is a good deal of prior work on human pose analysis, but relatively little on directly learning 3D pose from image measurements. (Brand, 1999) models a dynamical manifold of human body configurations with a Hidden Markov Model and learns using entropy minimization. (Athitsos & Sclaroff, 2000) learn a perceptron mapping between the appearance and parameter spaces. Human pose is hard to ground truth, so most papers in this area use only heuristic visual inspection to judge their results. However, the interpolated-k-nearest-neighbor learning method of (Shakhnarovich et al., 2003) used a human model rendering package (POSER from Curious Labs) to synthesize ground-truthed training and test images of 13 degree of freedom upper body poses with a limited ($\pm 40°$) set of random torso movements and view points, obtaining RMS estimation errors of about $20°$ per d.o.f. In comparison, our regression algorithm estimates full 54 d.o.f. body pose and orientation — a problem whose high dimensionality would really stretch the capacity of an example based method such as (Shakhnarovich et al., 2003) — with mean errors of only about $4°$. We also used POSER to synthesize a large set of training and test images from different viewpoints, but rather than using random synthetic poses, we used poses taken from real human motion capture sequences. Our results thus relate to real poses and we also capture the dynamics of typical human motions for temporal consistency. The motion capture data was taken from the public website *www.ict.usc.edu/graphics/animWeb/humanoid* .

Figure 1. Different 3D poses can have very similar image observations, causing the regression from image silhouettes to 3D pose to be inherently multi-valued.

(Howe et al., 1999) developed a Bayesian learning framework to recover 3D pose from known image locations of body joint centres, based on a training set of pose-centre pairs obtained from resynthesized motion capture data. (Mori & Malik, 2002) estimate the centres using shape context image matching against a set of training images with pre-labelled centres, then reconstruct 3D pose using the algorithm of (Taylor, 2000). Rather than working indirectly via joint centres, we chose to estimate pose directly from the underlying image descriptors, as we feel that this is likely to prove both more accurate and more robust, providing a generic framework for estimating and tracking any prespecified set of parameters from image observations.

(Pavlovic et al., 2000, Ormoneit et al., 2000) learn dynamical models for specific human motions. Particle filters and MCMC methods have widely been used in probabilistic tracking frameworks *e.g.* (Sidenbladh et al., 2002). Most of the previous learning based methods for human tracking take a generative, model based approach, whereas our approach is essentially discriminative.

2. Observations as Shape Descriptors

To improve resistance to segmentation errors and occlusions, we use a robust representation for our image observations. Of the many different image descriptors that could be used for human pose estimation, and in line with (Brand, 1999, Athitsos & Sclaroff, 2000), we have chosen to base our system on image silhouettes. There are two

main problems with silhouettes: (*i*) Artifacts such as shadow attachment and poor background segmentation tend to distort their local form. This often causes problems when global descriptors such as shape moments are used, as in (Brand, 1999, Athitsos & Sclaroff, 2000), because each local error pollutes every component of the descriptor. To be robust, shape descriptors must have good spatial locality. (*ii*) Silhouettes make several discrete and continuous degrees of freedom invisible or poorly visible. It is difficult to tell frontal views from back ones, whether a person seen from the side is stepping with the left leg or the right one, and what are the exact poses of arms or hands that fall within (are 'occluded' by) the torso's silhouette (see fig. 1). These factors limit the performance attainable from silhouette-based methods.

Histograms of edge information are a good way to encode local shape robustly (Lowe, 1999). Here, we use shape contexts (histograms of local edge pixels into log-polar bins) (Belongie et al., 2002) to encode silhouette shape quasi-locally over a range of scales, making use of their locality properties and capability to encode approximate spatial position on the silhouette — see (Agarwal & Triggs, 2004a). Unlike Belognie *et al*, we use quite small image regions (roughly the size of a limb) to compute our shape contexts, and for increased locality, we normalize each shape context histogram only by the number of points in its region. This is essential for robustness against occlusions, shadows, *etc*. The shape context distributions of all edge points on a silhouette are reduced to 100-D histograms by vector quantizing the 60-D shape context space using Gaussian weights to vote softly into the few histogram centres nearest to the contexts. This softening allows us to compare histograms using simple Euclidean distance rather than, say, Earth Movers Distance (Rubner et al., 1998). Each image observation (silhouette) is thus finally reduced to a 100-D quantized-distribution-of-shape-context vector, giving reasonably good robustness to occlusions and to local silhouette segmentation failures.

3. Tracking and Regression

The 3D pose can only be observed indirectly via ambiguous and noisy image measurements, so it is appropriate to start by considering the Bayesian tracking framework in which our knowledge about the state (pose) $\mathbf{x}_t$ given the observations up to time t is represented by a probability distribution, the posterior state density $p(\mathbf{x}_t|\mathbf{z}_t, \mathbf{z}_{t-1}, \ldots, \mathbf{z}_0)$.

Given an image observation $\mathbf{z}_t$ and a prior $p(\mathbf{x}_t)$ on the corresponding pose $\mathbf{x}_t$, the posterior likelihood for $\mathbf{x}_t$ is usually evaluated using Bayes' rule, $p(\mathbf{x}_t|\mathbf{z}_t) \propto p(\mathbf{z}_t|\mathbf{x}_t)\, p(\mathbf{x}_t)$, where $p(\mathbf{z}_t|\mathbf{x}_t)$ is a precise 'generative' observation model that predicts $\mathbf{z}_t$ and its uncertainty given $\mathbf{x}_t$. Unfortunately, when tracking objects as complicated as the human body, the observations depend on a great many factors that are difficult to control, ranging from lighting and background to body shape and clothing style and texture, so any hand-built observation model is necessarily a gross oversimplification.

One way around this would be to learn the generative model $p(\mathbf{z}|\mathbf{x})$ from examples, then to work backwards via its Jacobian to get a linearized state update, as in the extended Kalman filter. However, this approach is somewhat indirect, and it may waste a considerable amount of effort modelling appearance details that are irrelevant for predicting pose. Instead, we prefer to learn a 'discriminative' (diagnostic or anti-causal) model $p(\mathbf{x}|\mathbf{z})$ for the pose $\mathbf{x}$ given the observations $\mathbf{z}$ — *c.f.* the difference between generative and discriminative classification, and the regression based trackers of (Jurie & Dhome, 2002, Williams et al., 2003). Similarly, in the context of maximum likelihood pose estimation, we would prefer to learn a 'diagnostic' regressor $\mathbf{x} = \mathbf{x}(\mathbf{z})$, *i.e.* a point estimator for the most likely state $\mathbf{x}$ given the observations $\mathbf{z}$, not a generative predictor $\mathbf{z} = \mathbf{z}(\mathbf{x})$.

Unfortunately, this brings up a second problem. In monocular human pose reconstruction, image projection suppresses most of the depth (camera-object distance) information, so the state-to-observation mapping is always many-to-one. In fact, even when the labelled image positions of the projected joint centers are known exactly, there may still be some hundreds or thousands of kinematically possible 3D poses, linked by 'kinematic flipping' ambiguities (*c.f.* *e.g.* (Sminchisescu & Triggs, 2003)). Using silhouettes as image observations allows relatively robust feature extraction, but induces further ambiguities owing to the lack of limb labelling: it can be hard to tell back views from front ones, and which leg or arm is which in side views. These ambiguities make learning to regress $\mathbf{x}$ from $\mathbf{z}$ difficult because the true mapping is actually multi-valued. A single-valued least squares regressor will tend to either zig-zag erratically between different training poses, or (if highly damped) to reproduce their arithmetic mean (Bishop, 1995), neither of which is desirable. Introducing a robustified cost function might help the regressor to focus on just one branch of the solution space so that different regressors could be learned for different branches, but applying this in a heavily branched 54-D target space is not likely to be straightforward.

To reduce the ambiguity, we can take advantage of the fact that we are tracking and work incrementally from the previous state $\mathbf{x}_{t-1}$[1] (*e.g.* (D'Souza et al., 2001)). The basic assumption of discriminative tracking is that state information from the current observation is independent of state in-

[1] As an alternative we tried regressing the pose $\mathbf{x}_t$ against a sequence of the last few silhouettes $(\mathbf{z}_t, \mathbf{z}_{t-1}, \ldots)$, but the ambiguities are found to persist for several frames.

formation from previous states (dynamics):

$$p(\mathbf{x}_t \,|\, \mathbf{z}_t, \mathbf{x}_{t-1}, \ldots) \;\propto\; p(\mathbf{x}_t \,|\, \mathbf{z}_t)\, p(\mathbf{x}_t \,|\, \mathbf{x}_{t-1}, \ldots) \qquad (1)$$

The pose reconstruction ambiguity is reflected in the fact that the likelihood $p(\mathbf{x}_t \,|\, \mathbf{z}_t)$ is typically multimodal (*e.g.* it is obtained by using Bayes' rule to invert the many-to-one generative model $p(\mathbf{z}|\mathbf{x})$). Probabilistically this is fine, but to handle it in the context of point estimation / maximum likelihood tracking, we would in principle need to learn a *multi-valued* regressor for $\mathbf{x}_t(\mathbf{z}_t)$ and then fuse each of the resulting pose estimates with the estimate from the dynamics-based regressor $\mathbf{x}_t(\mathbf{x}_{t-1}, \ldots)$. Instead, we adopt the working hypothesis that given the dynamics based estimate — or any other rough initial estimate $\check{\mathbf{x}}_t$ for $\mathbf{x}_t$ — it will usually be the case that only one of the observation-based estimates is at all likely a posteriori. Thus, we can use the $\check{\mathbf{x}}_t$ value to "select the correct solution" for the observation-based reconstruction $\mathbf{x}_t(\mathbf{z}_t)$. Formally this gives a regressor $\mathbf{x}_t = \mathbf{x}_t(\mathbf{z}_t, \check{\mathbf{x}}_t)$, where $\check{\mathbf{x}}_t$ serves mainly as a key to select which branch of the pose-from-observation space to use, not as a useful prediction of $\mathbf{x}_t$ in its own right. (To work like this, this regressor must be nonlinear and well-localized in $\check{\mathbf{x}}_t$). Taking this one step further, if $\check{\mathbf{x}}_t$ is actually a useful estimate of $\mathbf{x}_t$ (*e.g.* from a dynamical model), we can use a single regressor of the same form, $\mathbf{x}_t = \mathbf{x}_t(\mathbf{z}_t, \check{\mathbf{x}}_t)$, but now with a stronger dependence on $\check{\mathbf{x}}_t$, to capture the net effect of implicitly reconstructing an observation-estimate $\mathbf{x}_t(\mathbf{z}_t)$ and then fusing it with $\check{\mathbf{x}}_t$ to get a better estimate of $\mathbf{x}_t$.

4. Learning the Regression Models

In this section we detail the regression methods that we use for recovering 3D human body pose. Poses are represented as real vectors $\mathbf{x} \in \mathbb{R}^m$. For a full body model, these are 55-dimensional, including 3 joint angles for each of the 18 major body joints[2]. This is not a minimal representation of the true human pose degrees of freedom, but it corresponds to our motion capture based training data, and our regression methods handle such redundant output representations without problems.

4.1. Dynamical (Prediction) Model

Human body dynamics can be modelled fairly accurately with a second order linear autoregressive process, $\mathbf{x}_t = \check{\mathbf{x}}_t + \boldsymbol{\epsilon}$, where $\check{\mathbf{x}}_t \equiv \tilde{\mathbf{A}}\,\mathbf{x}_{t-1} + \tilde{\mathbf{B}}\,\mathbf{x}_{t-2}$ is the second order dynamical estimate of $\mathbf{x}_t$ and $\boldsymbol{\epsilon}$ is a residual error vector (*c.f. e.g.* (Agarwal & Triggs, 2004b)). To ensure dynamical stability and avoid over-fitting, we actually learn the autoregression for $\check{\mathbf{x}}_t$ in the following form:

$$\check{\mathbf{x}}_t \;\equiv\; (\mathbf{I} + \mathbf{A})(2\mathbf{x}_{t-1} - \mathbf{x}_{t-2}) + \mathbf{B}\,\mathbf{x}_{t-1} \qquad (2)$$

where $\mathbf{I}$ is the $m \times m$ identity matrix. We estimate $\mathbf{A}$ and $\mathbf{B}$ by regularized least squares regression against $\mathbf{x}_t$, minimizing $\|\boldsymbol{\epsilon}\|_2^2 + \lambda(\|\mathbf{A}\|_{\text{Frob}}^2 + \|\mathbf{B}\|_{\text{Frob}}^2)$ over the training set, with the regularization parameter λ set by cross-validation to give a well-damped solution with good generalization.

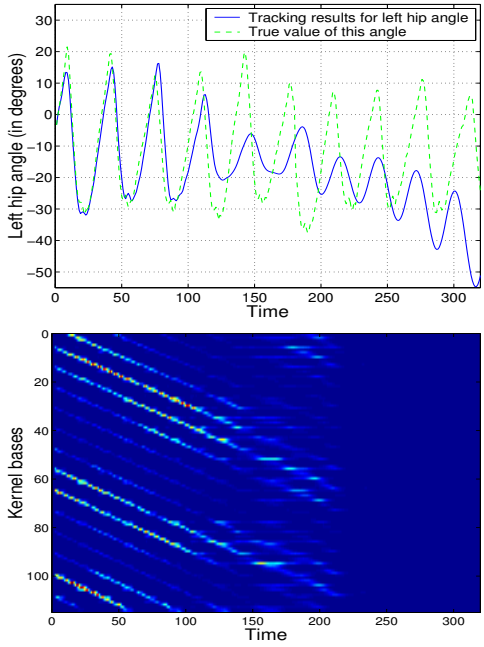

Figure 2. An example of mistracking caused by an over-narrow pose kernel K_x. The kernel width is set to 1/10 of the optimal value, causing the tracker to lose track from about t=120, after which the state estimate drifts away from the training region and all kernels stop firing by about t=200. *Top:* the variation of one parameter (left hip angle) for a test sequence of a person walking in a spiral. *Bottom:* The temporal activity of the 120 kernels (training examples) during this track. The banded pattern occurs because the kernels are samples taken from along a similar 2.5 cycle spiral walking sequence, each circuit involving about 8 steps. The similarity between adjacent steps and between different circuits is clearly visible, showing that the regressor can locally still generalize well.

4.2. Likelihood (Correction) Model

Now consider the observation model. As discussed above, the underlying density $p(\mathbf{x}_t \,|\, \mathbf{z}_t)$ is highly multimodal owing to the pervasive ambiguities in reconstructing 3D pose from monocular images, so no single-valued regression function $\mathbf{x}_t = \mathbf{x}_t(\mathbf{z}_t)$ can give acceptable point estimates for $\mathbf{x}_t$. This is confirmed in practice: although we have managed to learn moderately successful pose regressors $\mathbf{x} = \mathbf{x}(\mathbf{z})$, they tend to systematically underestimate pose angles (owing to effective averaging over several possible

[2]The subject's overall azimuth (compass heading angle) θ can wrap around through 360°. We maintain continuity by regressing $(a, b) = (\cos\theta, \sin\theta)$ rather than θ, using $\text{atan2}(b, a)$ to recover θ from the not-necessarily-normalized vector returned by regression. We thus have $3{\times}18{+}1 = 55$ parameters to estimate.

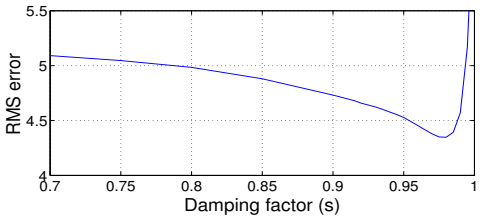

Figure 3. The variation of the RMS test-set tracking error with damping factor *s*. See the text for discussion.

solutions) and to be subject to occasional glitches where the wrong solution is selected (Agarwal & Triggs, 2004a). Although such regressors can be combined with dynamics-based predictors, this only smooths the results: it cannot remove the underlying underestimation and 'glitchiness'.

In default of a reliable method for multi-valued regression, we include a non-linear dependence on $\check{\mathbf{x}}_t$ with $\mathbf{z}_t$ in the observation-based regressor. Our full regression model also includes an explicit $\check{\mathbf{x}}_t$ term to represent the direct contribution of the dynamics to the overall state estimate, so the final model becomes $\mathbf{x}_t \equiv \hat{\mathbf{x}}_t + \boldsymbol{\epsilon}'$ where $\boldsymbol{\epsilon}'$ is a residual error to be minimized, and:

$$\hat{\mathbf{x}}_t \;\equiv\; \mathbf{C}\,\check{\mathbf{x}}_t + \sum_{k=1}^{p} \mathbf{d}_k\,\phi_k(\check{\mathbf{x}}_t, \mathbf{z}_t) \;=\; \begin{pmatrix}\mathbf{C} & \mathbf{D}\end{pmatrix} \begin{pmatrix}\check{\mathbf{x}}_t \\ \mathbf{f}(\check{\mathbf{x}}_t, \mathbf{z}_t)\end{pmatrix} \tag{3}$$

Here, $\{\phi_k(\mathbf{x}, \mathbf{z}) \,|\, k = 1 \ldots p\}$ is a set of scalar-valued basis functions for the regression, and $\mathbf{d}_k$ are the corresponding $\mathbb{R}^m$-valued weight vectors. For compactness, we gather these into an $\mathbb{R}^p$-valued feature vector $\mathbf{f}(\mathbf{x}, \mathbf{z}) \equiv (\phi_1(\mathbf{x}, \mathbf{z}), \ldots, \phi_p(\mathbf{x}, \mathbf{z}))^\top$ and an $m \times p$ weight matrix $\mathbf{D} \equiv (\mathbf{d}_1, \ldots, \mathbf{d}_p)$. In the experiments reported here, we used instantiated-kernel bases of the form:

$$\phi_k(\mathbf{x}, \mathbf{z}) = K_x(\mathbf{x}, \mathbf{x}_k) \cdot K_z(\mathbf{z}, \mathbf{z}_k) \tag{4}$$

where $(\mathbf{x}_k, \mathbf{z}_k)$ is a training example and K_x, K_z are (here, independent Gaussian) kernels on $\mathbf{x}$-space and $\mathbf{z}$-space, $K_x(\mathbf{x}, \mathbf{x}_k) = e^{-\beta_x \|\mathbf{x}-\mathbf{x}_k\|^2}$ and $K_z(\mathbf{z}, \mathbf{z}_k) = e^{-\beta_z \|\mathbf{z}-\mathbf{z}_k\|^2}$.

Building the basis from Gaussians based at training examples in joint $(\mathbf{x}, \mathbf{z})$ space forces examples to become relevant only if they have similar estimated poses *and* similar image silhouettes. It is essential to choose the relative widths of the kernels appropriately. In particular, if the $\mathbf{x}$-kernel is chosen too wide, the method tends to average over (or zig-zag between) several alternative pose-from-observation solutions, which defeats the purpose of including $\check{\mathbf{x}}$ in the observation regression. On the other hand, by locality, the observation-based state corrections are effectively 'switched off' whenever the state happens to wander too far from the observed training examples $\mathbf{x}_k$. So if the $\mathbf{x}$-kernel is set too narrow, observation information is only incorporated sporadically and mistracking can easily occur.

RVM Training Algorithm

0. Initialize $\mathbf{A}$ with ridge regression. Initialize the running scale estimates $a_{\text{scale}} = \|\mathbf{a}\|$ for the components or vectors $\mathbf{a}$.

1. Approximate the $\nu \log \|\mathbf{a}\|$ penalty terms with "quadratic bridges" $\nu\,(\mathbf{a}/a_{\text{scale}})^2 + \text{const}$ (the gradients match at a_{scale});

2. Solve the resulting linear least squares problem in $\mathbf{A}$;

3. Remove any components $\mathbf{a}$ that have become zero, update the scale estimates $a_{\text{scale}} = \|\mathbf{a}\|$, and continue from 1 until convergence.

Figure 4. Our RVM training algorithm.

Fig. 2 illustrates this effect, for an $\mathbf{x}$-kernel a factor of 10 narrower than the optimum. The method initially seemed to be sensitive to the kernel width parameters, but after selecting optimal parameters by cross-validation on an independent motion sequence we observed accurate performance over a sufficiently wide range of both the kernel widths: a tolerance factor of $\sim$2 on β_x and $\sim$4 on β_z.

The coefficient matrix $\mathbf{C}$ in (3) plays an interesting role. Setting $\mathbf{C} \equiv \mathbf{I}$ forces the correction model to act as a differential update on $\check{\mathbf{x}}_t$. On the other extreme, $\mathbf{C} \equiv \mathbf{0}$ gives largely observation-based state estimates with only a latent dependence on the dynamics. An intermediate setting, however, turns out to give best overall results. Damping the dynamics slightly ensures stability and controls drift — in particular, preventing the observations from disastrously 'switching off' because the state has drifted too far from the training examples — while still allowing a reasonable amount of dynamical smoothing. Usually we estimate the full (regularized) matrix $\mathbf{C}$ from the training data, but to get an idea of the trade-offs involved, we also studied the effect of explicitly setting $\mathbf{C} = s\mathbf{I}$ for $s \in [0, 1]$. We find that a small amount of damping, $s_{opt} \approx .98$ gives the best results overall, maintaining a good lock on the observations without losing too much dynamical smoothing (see fig. 3.) This simple heuristic setting gives very similar results to the full model obtained by learning an unconstrained $\mathbf{C}$.

4.3. Relevance Vector Regression

The regressor is learned using a Relevance Vector Machine (Tipping, 2001). This sparse Bayesian approach gives similar results to methods such as damped least squares / ridge regression, but selects a much more economical set of active training examples for the kernel basis. We have also tested a number of other training methods (including ridge regression) and bases (including the linear basis). These are not reported here, but the results turn out to be relatively insensitive to the training method used, with the kernel bases having a slight edge.

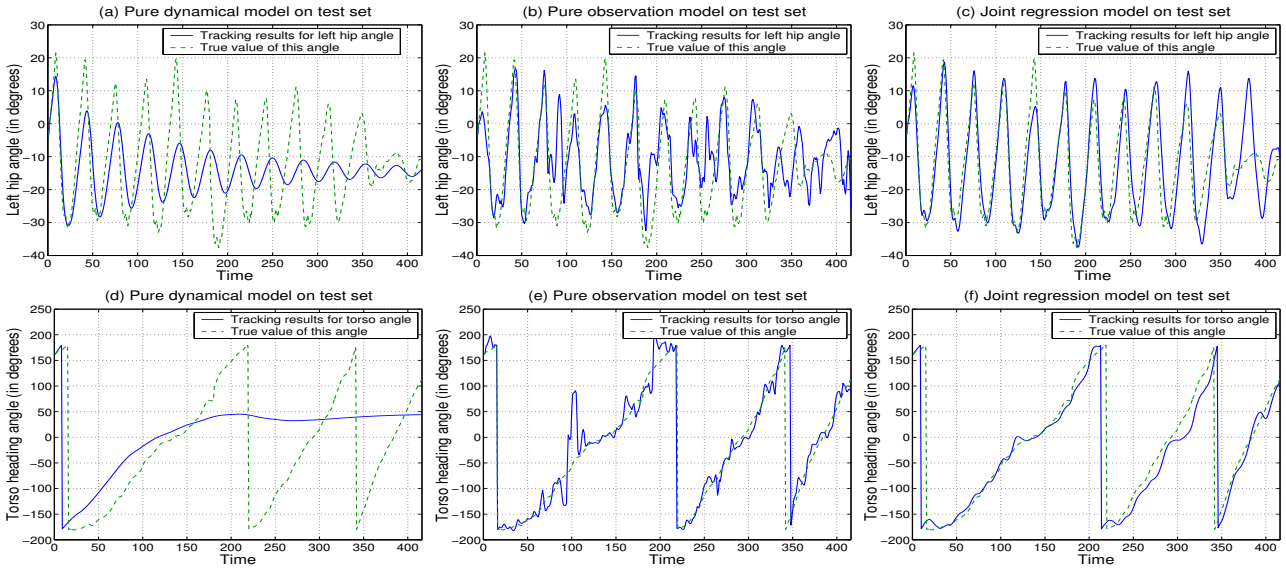

Figure 5. Tracking results on a spiral walking test sequence. (a) Variation of a joint-angle parameter, as predicted by a pure dynamical model initialized at $t = \{0, 1\}$, (b) Estimated values of this angle from regression on observations alone (*i.e.* no initialization or temporal information), (c) Results from our novel joint regressor, obtained by combining dynamical and state+observation based regression models. (d,e,f) Similar plots for the overall body rotation angle. Note that this angle wraps around 360°, i.e. $\theta \cong \theta \pm 360^\circ$.

When regressing $\mathbf{y}$ on $\mathbf{x}$ (using generic notation), we use Euclidean norm to measure $\mathbf{y}$-space prediction errors, so the estimation problem takes the form:

$$\mathbf{A} \;:=\; \arg\min_{\mathbf{A}} \left\{ \sum_{i=1}^{n} \|\mathbf{A}\,\mathbf{f}(\mathbf{x}_i) - \mathbf{y}_i\|^2 \;+\; R(\mathbf{A}) \right\} \quad (5)$$

where $R(-)$ is a regularizer on $\mathbf{A}$. RVM's take either individual parameters or groups of parameters $\mathbf{a}$ (in our case, columns of $\mathbf{A}$), and impose $\nu \log \|\mathbf{a}\|$ regularizers or priors on each group. Rather than using the (Tipping, 2000) algorithm for training, we use a continuation method based on successively approximating the $\nu \log \|\mathbf{a}\|$ regularizers with quadratic "bridges" $\nu\,(\|\mathbf{a}\|/a_{\text{scale}})^2$ chosen to match the prior gradient at a_{scale}, a running scale estimate for $\mathbf{a}$. The bridging functions allow parameters to pass through zero if they need to, without too much risk of premature trapping at zero. The algorithm is sketched in fig. 4. Regularizing over whole columns (rather than individual components) of $\mathbf{A}$ ensures a sparse expansion, as it swaps entire basis functions in or out.

5. Experimental Results & Analysis

We conducted experiments using a database of motion capture data for an $m = 54$ d.o.f. body model (3 angles for each of 18 joints, including body orientation w.r.t. the camera). We report mean (over all angles) RMS (over time) absolute difference errors between the true and estimated joint angle vectors, in degrees:

$$D(\mathbf{x}, \mathbf{x}') = \frac{1}{m} \sum_{i=1}^{m} |(x_i - x'_i) \bmod \pm 180^\circ| \quad (6)$$

The training silhouettes were created by using Curious Labs' POSER to re-render poses obtained from real human motion capture data, and reduced to 100-D shape descriptor vectors as in §2. We used 8 different sequences totalling ∼2000 instantaneous poses for training, and another two sequences of ∼400 points each as validation and test sets.

The dynamical model is learned from the training data exactly as described in §4.1, but when training the observation model, we find that its coverage and capture radius can be increased by including a wider selection of $\check{\mathbf{x}}_t$ values than those produced by the dynamical predictions. Hence, we train the model $\mathbf{x} = \mathbf{x}_t(\check{\mathbf{x}}, \mathbf{z})$ using a combination of 'observed' samples $(\check{\mathbf{x}}_t, \mathbf{z}_t)$ (with $\check{\mathbf{x}}_t$ computed from (2)) and artificial samples generated by Gaussian sampling $\mathcal{N}(\mathbf{x}_t, \Sigma)$ around the training state $\mathbf{x}_t$. The observation $\mathbf{z}_t$ corresponding to $\mathbf{x}_t$ is still used, forcing the observation based part of the regressor to rely mainly on the observations, *i.e.* on recovering $\mathbf{x}_t$ (or at least an update to $\check{\mathbf{x}}_t$) from $\mathbf{z}_t$, using $\check{\mathbf{x}}_t$ mainly as a hint about the inverse solution to choose. The covariance matrix Σ is chosen to reflect the local scatter of the training examples, with a larger variance along the tangent to the trajectory at each point to ensure that phase lag between the state estimate and the true state is reliably detected and corrected.

Fig. 5 illustrates the relative contributions of the different terms in our model by plotting tracking results for a motion capture test sequence in which the subject walks in a

decreasing spiral. (This sequence was not included in the training set, although similar ones were). The purely dynamical model (2) provides good estimates for a few time steps, but gradually damps and drifts out of phase. (Such damped oscillations are characteristic of second order linear autoregressive dynamics, trained with enough regularization to ensure model stability). At the other extreme, using observations alone without any temporal information (*i.e.* $\mathbf{C} = \mathbf{0}$ and $K_x = 1$) provides noisy reconstructions with occasional 'glitches' due to incorrect reconstructions. Panels (c),(f) show that joint regression on both dynamics and observations gives smoother and stabler tracking. There is still some residual misestimation of the hip angle in (c) at around t=140 and t=380. Here, the subject is walking directly towards the camera (heading angle $\theta \sim 0^\circ$), so the only cue for hip angle is the position of the corresponding foot, which is sometimes occluded by the opposite leg. Even humans have difficulty estimating this angle from the silhouette at these points.

Fig. 6 shows some silhouettes and corresponding maximum likelihood pose reconstructions, for the same test sequence. The 3D poses for the first two time steps were set by hand to initialize the dynamical predictions. The average RMS estimation error over all joints using the RVM regressor in this test is 4.1°. Well-regularized least squares regression over the same basis gives similar errors, but has much higher storage requirements. The Gaussian RVM gives a sparse regressor for (3) involving only 348 of the 1927 training examples, thus allowing a significant reduction in the amount of training data that needs to be stored. Reconstruction results on a test video sequence are shown in fig. 7. The reconstruction quality demonstrates the generalized dynamical behavior captured by the model as well as the method's robustness to imperfect visual features, as a naive background subtraction method was used to extract somewhat imperfect silhouettes from the images.

In terms of computational time, the final RVM regressor already runs in real time in Matlab. Silhouette extraction and shape-context descriptor computations are currently done offline, but would be doable online in real time. The (offline) learning process takes about 26 min for the RVM with $\sim$2000 data points, and about the same again for (Matlab) Shape Context extraction and clustering.

The method is reasonably robust to initialization errors. The results shown in figs. 5 and 6 were obtained by initializing from ground truth, but we also tested the effects of automatic (and hence potentially incorrect) initialization. In an experiment in which the tracker was automatically initialized at each time step in turn using the pure observation model, then tracked forwards and backwards using the dynamical tracker, the initialization lead to successful tracking in 84% of the cases. The failures occur at the 'glitches',

Figure 6. Some sample pose reconstructions for a spiral walking sequence not included in the training data, corresponding to figures 5(c) & (f). The reconstructions were computed with a Gaussian kernel RVM, using only 348 of the 1927 training examples. The average RMS estimation error per d.o.f. over the whole sequence is 4.1°.

where the observation model gave completely incorrect initializations.

6. Discussion & Conclusions

We have presented a method that recovers 3D human body pose from sequences of monocular silhouettes by direct nonlinear regression of joint-angles against histogram-of-shape-context silhouette shape descriptors and dynamics based pose estimates. No 3D body model or labelling of image positions of body parts is required. Regressing the pose jointly on image observations and previous poses allows the intrinsic ambiguity of the pose-from-monocular-observations problem to be overcome, thus producing stable, temporally consistent tracking. We use a kernel-based Relevance Vector Machine for the regression, thus selecting a sparse set of relevant training examples as exemplars. The method shows promising results on tracking unseen video sequences, giving an average RMS error of 4.1° per body-joint-angle on real motion capture data.

Future work: We plan to investigate the extension of our regression based system to a complete discriminative Bayesian tracking framework, including multiple hypotheses and robust error models. We would also like to include richer features, such as internal edges in addition to silhouette boundaries to reduce susceptibility to poor image segmentation.

Acknowledgments

This work was supported by the European Union projects VIBES and LAVA, and the research network PASCAL.

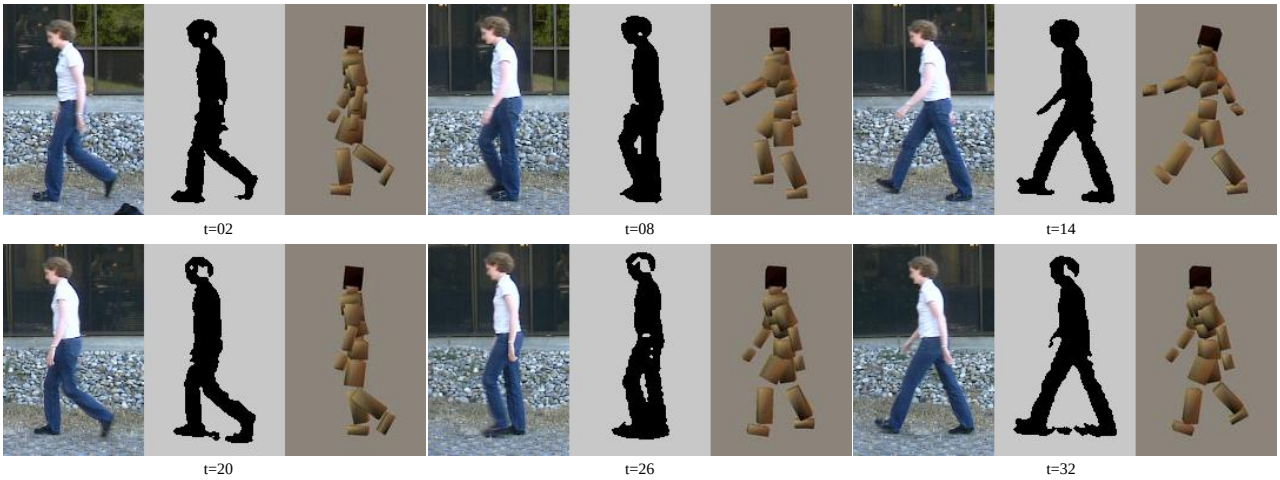

Figure 7. 3D poses reconstructed from a test video sequence (obtained from www.nada.kth.se/∼hedvig/data.html). The presence of shadows and holes in the extracted silhouettes demonstrates the robustness of our shape descriptors — however, a weak or noisy observation signal sometimes causes failure to track accurately. *E.g.* at $t = 8, 14$, the pose estimates are dominated by the dynamical predictions, which do ensure smooth and natural motion but may cause slight mistracking of some parameters.

References

Agarwal, A., & Triggs, B. (2004a). 3D Human Pose from Silhouettes by Relevance Vector Regression. *Int. Conf. Computer Vision & Pattern Recognition*.

Agarwal, A., & Triggs, B. (2004b). Tracking Articulated Motion with Piecewise Learned Dynamical Models. *European Conf. Computer Vision*.

Athitsos, V., & Sclaroff, S. (2000). Inferring Body Pose without Tracking Body Parts. *Int. Conf. Computer Vision & Pattern Recognition*.

Athitsos, V., & Sclaroff, S. (2003). Estimating 3D Hand Pose From a Cluttered Image. *Int. Conf. Computer Vision*.

Belongie, S., Malik, J., & Puzicha, J. (2002). Shape Matching and Object Recognition using Shape Contexts. *IEEE Trans. Pattern Analysis & Machine Intelligence*, *24*, 509–522.

Bishop, C. (1995). *Neural Networks for Pattern Recognition*, chapter 6. Oxford University Press.

Brand, M. (1999). Shadow Puppetry. *Int. Conf. Computer Vision* (pp. 1237–1244).

Bregler, C., & Malik, J. (1998). Tracking People with Twists and Exponential Maps. *Int. Conf. Computer Vision & Pattern Recognition* (pp. 8–15).

D'Souza, A., Vijayakumar, S., & Schaal, S. (2001). Learning Inverse Kinematics. *Int. Conf. on Intelligent Robots and Systems*.

Grauman, K., Shakhnarovich, G., & Darrell, T. (2003). Inferring 3D Structure with a Statistical Image-Based Shape Model. *Int. Conf. Computer Vision* (pp. 641–648).

Howe, N., Leventon, M., & Freeman, W. (1999). Bayesian Reconstruction of 3D Human Motion from Single-Camera Video. *Neural Information Processing Systems*.

Jurie, F., & Dhome, M. (2002). Hyperplane Approximation for Template Matching. *IEEE Trans. Pattern Analysis & Machine Intelligence*, *24*, 996–1000.

Lowe, D. (1999). Object Recognition from Local Scale-invariant Features. *Int. Conf. Computer Vision* (pp. 1150–1157).

Mori, G., & Malik, J. (2002). Estimating Human Body Configurations Using Shape Context Matching. *European Conf. Computer Vision* (pp. 666–680).

Ormoneit, D., Sidenbladh, H., Black, M., & Hastie, T. (2000). Learning and Tracking Cyclic Human Motion. *Neural Information Processing Systems* (pp. 894–900).

Pavlovic, V., Rehg, J., & MacCormick, J. (2000). Learning Switching Linear Models of Human Motion. *Neural Information Processing Systems* (pp. 981–987).

Rubner, Y., Tomasi, C., & Guibas, L. (1998). A Metric for Distributions with Applications to Image Databases. *Int. Conf. Computer Vision*. Bombay.

Shakhnarovich, G., Viola, P., & Darrell, T. (2003). Fast Pose Estimation with Parameter Sensitive Hashing. *Int. Conf. Computer Vision*.

Sidenbladh, H., Black, M., & Sigal, L. (2002). Implicit Probabilistic Models of Human Motion for Synthesis and Tracking. *European Conf. Computer Vision*.

Sminchisescu, C., & Triggs, B. (2003). Kinematic Jump Processes For Monocular 3D Human Tracking. *Int. Conf. Computer Vision & Pattern Recognition*.

Stenger, B., Thayananthan, A., Torr, P., & Cipolla, R. (2003). Filtering Using a Tree-Based Estimator. *Int. Conf. Computer Vision*.

Taylor, C. (2000). Reconstruction of Articulated Objects from Point Correspondances in a Single Uncalibrated Image. *Int. Conf. Computer Vision & Pattern Recognition*.

Tipping, M. (2000). The Relevance Vector Machine. *Neural Information Processing Systems*.

Tipping, M. (2001). Sparse Bayesian Learning and the Relevance Vector Machine. *J. Machine Learning Research*, *1*, 211–244.

Williams, O., Blake, A., & Cipolla, R. (2003). A Sparse Probabilistic Learning Algorithm for Real-Time Tracking. *Int. Conf. Computer Vision*.