Observable Subspaces for 3D Human Motion Recovery *

Andrea Fossati
CVLab
EPFL
andrea.fossati@epfl.ch

Mathieu Salzmann
EECS & ICSI
UC Berkeley
salzmann@icsi.berkeley.edu

Pascal Fua
CVLab
EPFL
pascal.fua@epfl.ch

Abstract

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

This has often been used to make motion tracking easier. However, learning the latent space in a problem-independent way makes it non trivial to initialize the tracking process by picking appropriate initial values for the latent variables, and thus for the pose. In this paper, we show that by directly using observable quantities as our latent variables, we eliminate this problem and achieve full automation given only modest amounts of training data.

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

1. Introduction

A common theme in many recent approaches to capturing human motion from video is to represent the set of likely poses as a low-dimensional manifold parameterized by a few latent variables. The mapping between the latent space and the pose space correlates the motions of individual body parts. This strongly constrains the fitting of a complete body model to image data, thus making the problem more tractable.

The mapping can be either linear [15, 11, 21] or non-linear [4, 16, 20] but is usually learned in a problem-independent way, which makes it difficult to recover motion without manually initializing the latent variables and the pose. Recent approaches have endeavored to address this issue by learning a shared low-dimensional latent space both for pose and image data [10]. Though this improves tracking performance, learning the manifold requires a complex training procedure that needs large amounts of data, and yields a latent space that has no intuitive meaning.

In this paper, we introduce a direct way to derive a mapping between easily observable image quantities, which will serve as our latent variables, and pose sequences. This yields full automation given only modest amounts of training data. The idea is very general and we demonstrate it in the cases of skating, skiing, and golfing, as shown in Figure 1.

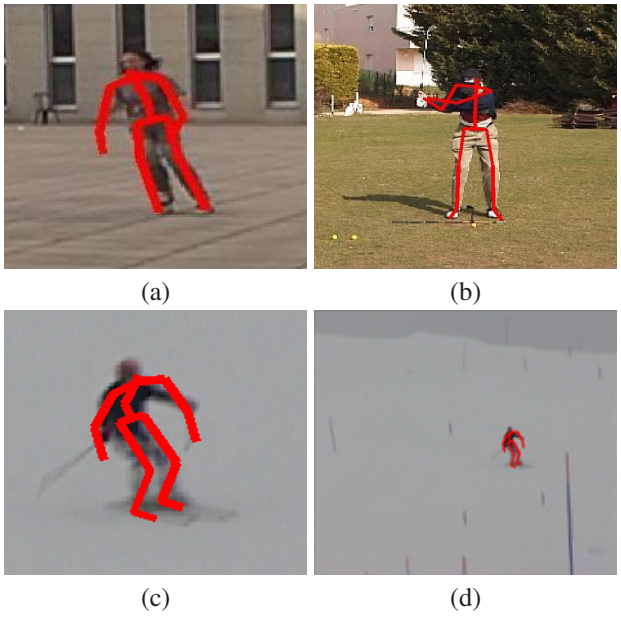


Figure 1. Our observable subspaces allow us to recover different 3D motions, such as roller skating (a), golfing (b) and skiing (c), even under adverse conditions. Note for example that (c) is a window of the 720×576 image shown in (d), which makes the subject quite small.

*This work has been funded in part by the Swiss National Science Foundation.

Specifically, the ground trajectory of a person's feet strongly constrains body pose for motions such as skating and skiing. Similarly, the hand trajectory in the swing plane provides pose constraints for the golf swing motion. These 2D trajectories are usually easy to compute from the images and can be parameterized as piecewise polynomials that are characterized by their curvatures. To capture the correlation between trajectory and motion, we learn a Gaussian Process mapping [12] from consecutive curvatures along the trajectories to 3D pose sequences that represent the motions. At run-time, we track people's feet or hands, fit splines to the resulting trajectories, and use this mapping to initialize the poses in a frame sequence. These poses can then be refined by minimizing an image-based criterion. In practice, because a sequence of 3D poses is too high-dimensional for direct fitting to image data, we first reduce its dimensionality by learning a linear subspace model [21]. A motion can then be expressed as an average motion plus a weighted sum of modes, and the mapping is learned from the trajectory curvatures to these weights. To demonstrate the effectiveness of this approach, we show that we can use our mapping to recover from single videos not only skating and golfing motions, which is what it was trained for, but also skiing motions, which are related to but different from skating ones.

2. Related Work

Even after many years of effort, recovering 3D human motion from image sequences reliably remains an open problem. Among the sources of difficulties are joint reflection ambiguities, occlusions, cluttered backgrounds, non-rigidity of tissue and clothing, complex and rapid motions, and poor image resolution. The problem is particularly pronounced when using a single video to recover the 3D motion. In this case, incorporating motion models into the algorithms has consistently been shown to be effective [9]. Such models can be physics-based [3, 23] or learned from training data [15, 11, 1, 21, 17, 13, 2].

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

To this end, the manifold is usually parameterized by a few latent variables and the mapping between them and the poses, or pose sequences, can be either linear [15, 11, 21] or not [4, 16, 20]. For example, in [20], a Gaussian Process Latent Variable Model (GPLVM) [8] was used to learn a differentiable manifold from modest amounts of training data, which allowed motion recovery by continuous optimization of an image-based objective function. There has been attempts at constraining the topology of the latent space to assume known configurations, such as circles, or to respect the distances in the high-dimensional space between neighboring examples [19]. However, such techniques still require learning a latent space, which remains a complex optimization problem, whereas we propose to directly make use of observable quantities as a latent space.

Another issue with this kind of approach is that the latent variables have no physical meaning and are hard to initialize from image data. In GPLVM approaches such as [20], the process is initiated by finding a training example that best fits the data and using the corresponding latent variable for initialization purposes. This implies a search that our technique avoids. This difficulty was addressed in [10, 14, 6] by learning a common low-dimensional latent space both for pose and image data. However, because learning the joint latent space is more involved than learning individual latent spaces, standard techniques require more training data than is normally available. As a result, the authors of [10] had to develop a more sophisticated algorithm able to use not only examples for which the correspondence between pose and image data is known, but also examples for which it is not. Unfortunately, learning a GPLVM is computationally expensive and sensitive to initialization of the latent variables. Furthermore, it yields a complex objective function with many local minima, which is not always ideal for inference purposes.

Here, by contrast, we rely on ordinary Gaussian Processes (GP) [12] to establish a direct mapping between low-dimensional observable image data and the high dimensional pose space. Since fewer parameters need to be learned, training is more straightforward and requires far less data. Initialization is similarly easy since the image data directly give us a mapping to a pose sequence, which we then simply need to refine.

GPs were also used in [18] to map silhouettes directly to 3D poses. However, because both spaces are highly non-linear, this required using not a single GP but a mixture of them and a very sophisticated learning procedure, which was only demonstrated on clean silhouette data. In our case, the mapping is straightforward and we can rely on a standard implementation of the training procedure.

3. Framework

Our goal is to relate 3D motions to image trajectories of the hands or feet so that we can predict the former from the latter. Here, we propose to learn a Gaussian Process mapping [12] from the space of image trajectories to that of hu-

man motions represented as sequences of 3D poses, which can be done with a relatively small training database. Given this mapping, we can track the hands or feet of subjects in video sequences, infer plausible motions, and refine them to obtain accurate 3D pose estimates by minimizing an image-based objective function. In practice, however, the space of 3D pose sequences is too high-dimensional to be directly used for optimization purposes. Therefore, to reduce the dimensionality of our problem and the complexity of optimization, we use a linear subspace motion model [21, 15] to represent 3D pose sequences with a manageable number of parameters, and learn a mapping from trajectory curvatures to these parameters.

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

3.1. Motion Representation

We rely on a coarse body model in which individual limbs are modeled as cylinders. Let $\mathbf{y}^t = [\psi_t^T, \mathbf{g}_t^T]^T$ be the vector that defines its pose at time t, where ψ_t is a set of N_j joint angles and $\mathbf{g}_t$ a 6D vector that defines the position and orientation of a reference body joint in a global reference system.

A *motion* can be viewed as a time-varying pose. While pose varies continuously over time, we assume a discrete representation in which pose is sampled at N_t distinct time instants. In this way, a motion $\mathbf{y}$ is just a sequence of N_t discrete poses, and can be written as the $D = (N_j N_t + 6N_t)$-dimensional vector

$$\mathbf{y} = [\psi_1^T, \cdots, \psi_{N_t}^T, \mathbf{g}_1^T, \cdots, \mathbf{g}_{N_t}^T]^T \ . \tag{1}$$

Naturally, we assume that the temporal sampling rate is sufficiently high to interpolate the continuous pose signal. In our examples we split activities into short and temporally smooth motions. Therefore we simply consider poses as equally-spaced in time between the beginning and the end of a motion. This avoids the need to explicitly account for differences in speed between motions.

3.2. Gaussian Processes

Let $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_N]^T$ be the $N \times D$ matrix of N training motions from which the mean motion $\mathbf{y}_0$ was subtracted, and $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_N]^T$ the $N \times d$ matrix of corresponding d-dimensional image trajectories parameters. $\mathbf{Y}$ and $\mathbf{X}$ are said to be related through a Gaussian Process mapping [12] if

$$\mathbf{y}_i = f(\mathbf{x}_i) + \epsilon_i \ , \tag{2}$$

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

$$p(\mathbf{f}|\mathbf{X}) = \mathcal{N}(0, \mathbf{K}) \ , \tag{3}$$

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

$$p(\mathbf{Y} \mid \mathbf{X}, \Theta) \ = \ \frac{1}{\sqrt{(2\pi)^{ND}|\mathbf{K}|^D}} \exp\left(-\frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{Y}\mathbf{Y}^T\right)\right) \ , \tag{4}$$

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

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

$$\mu(\mathbf{x}') \ = \ \mathbf{y}_0 + \mathbf{Y}^T\mathbf{K}^{-1}\mathbf{k}(\mathbf{x}') \ , \tag{5}$$

$$\sigma^2(\mathbf{x}') \ = \ k(\mathbf{x}', \mathbf{x}') - \mathbf{k}(\mathbf{x}')^T\mathbf{K}^{-1}\mathbf{k}(\mathbf{x}') \ , \tag{6}$$

where $\mathbf{k}(\mathbf{x}')$ is the vector with elements $k(\mathbf{x}', \mathbf{x}_j)$ for latent positions $\mathbf{x}_j \in \mathbf{X}$. We can therefore simply use the mean prediction of the model $\mu(\mathbf{x}')$ to initialize the motion in the new sequence, and refine it via optimization of an image-based objective function, as will be explained in Section 3.4.

3.3. Linear Subspace Motion Model

Since, in practice, optimizing an image-based criterion with respect to the $N_j N_t + 6N_t$ parameters of a sequence of poses is intractable, we first reduce the dimensionality of this space. To this end, we perform Principal Component Analysis on the dataset $\mathbf{Y}$ to find a low-dimensional basis with which we can effectively model the motion. In particular, the model approximates motions in the training set with a linear combination of the mean motion $\mathbf{y}_0$ and a set of *eigen-motions* $\{\tilde{\mathbf{y}}_i\}_{1 \le i \le N_m}$ as

$$\mathbf{y} \ \approx \ \mathbf{y}_0 + \sum_{i=1}^{N_m} \alpha_i \tilde{\mathbf{y}}_i \ . \tag{7}$$

The scalar coefficients, $\{\alpha_i\}$, characterize the motion, and $N_m \le N_j N_t + 6N_t$ controls the fraction of the total variance of the training data that is captured by the subspace, measured by

$$Q(N_m) \ = \ \frac{\sum_{i=1}^{N_m} \lambda_i}{\sum_{i=1}^{N_j N_t + 6N_t} \lambda_i} \ , \tag{8}$$

where λ_i are the eigenvalues of the data covariance matrix, ordered such that $\lambda_i \ge \lambda_{i+1}$. In practice, we choose N_m

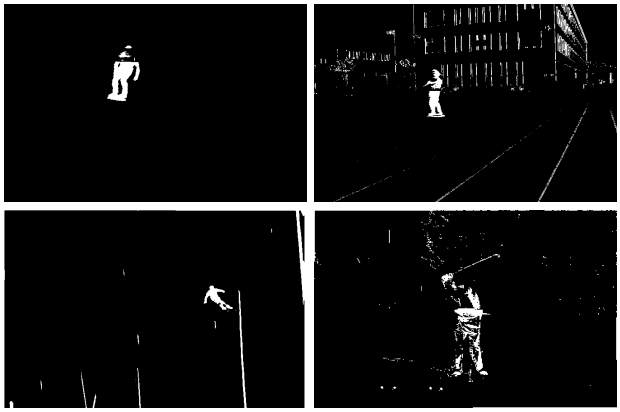


Figure 2. Input silhouettes used to compute our results. The silhouettes were extracted using a standard background subtraction technique on skating and golfing examples, while on skiing an intensity threshold was used.

such that $Q(N_m) > 0.9$. Finally, the GP mapping is learned from the trajectory curvatures to the parameters α_i of our training data rather than to the sequence of poses directly. Since we ensure that 90% of the training data is modeled by the linear subspace, this only yields a negligible loss of accuracy.

3.4. Fitting the Model to Image Data

Given an input video sequence, we can easily obtain the trajectory parameters $\mathbf{x}'$ as will be described in Section 4.2. From these, we compute the mean prediction $\mu(\mathbf{x}')$ of our GP model and use it to initialize the eigen-motion coefficients α'. We then refine α' by maximizing an image likelihood using a standard particle filtering technique [7]. To this end, we sample the linear subspace around our initial solution according to the eigenvalues λ_i. Note that, for this purpose, we could equivalently have used the variance $\sigma^2(\mathbf{x}')$ of Eq. 6. Since the motion in the images is of arbitrary length, we just warp the one we obtain with the eigen-motion coefficients to fit the correct number of frames through a simple spline interpolation of the N_j joint angles defining a pose.

The image likelihood is computed as a binary *AND* between the silhouette obtained by background subtraction in the input images and the reprojection of our cylinder-based body model in the computed poses. Our method is robust to very low-quality silhouettes, as shown in Figure 2. Because, in our examples, most global motion parameters $\mathbf{g}_1, \ldots, \mathbf{g}_{N_t}$ either can be computed from the feet trajectories or remain constant, the linear subspace decomposition is only performed on the joint angles. Only two global orientations need to be estimated from the images, which we do by considering them as unknowns in the first and last frames, and linearly interpolating them in between.

4. Experimental Results

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

In this section, we first describe our training data, next we explain how we obtain trajectory curvatures from sequences of images and finally we present our tracking results.

4.1. Obtaining Training Data

To obtain the training sequences of 3D poses, we used a commercial optical motion capture system that recovers the positions of reflective markers placed on the joints of a person using six infrared cameras [22].

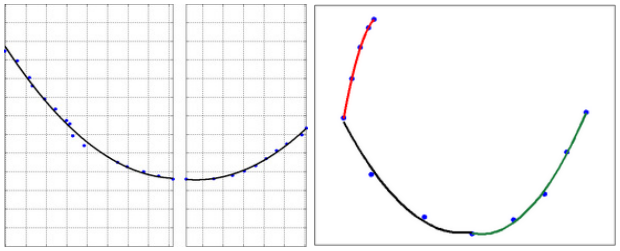
In the case of skating, we captured a subject performing turns with a varying radius. We then split the reconstructed sequences into small motions representing half a turn each and time-normalized these subsequences to build vectors of length $N_j N_t$ by concatenating N_t poses of N_j joint angles. For each one of these vectors, we computed the trajectory of the feet on the ground plane to which we fitted a second order polynomial. This yielded a two-dimensional latent representation $\mathbf{x}$ containing the curvature of the half-turns and a parameter discriminating between the two halves of a turn.

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

4.2. Retrieving Trajectory Parameters

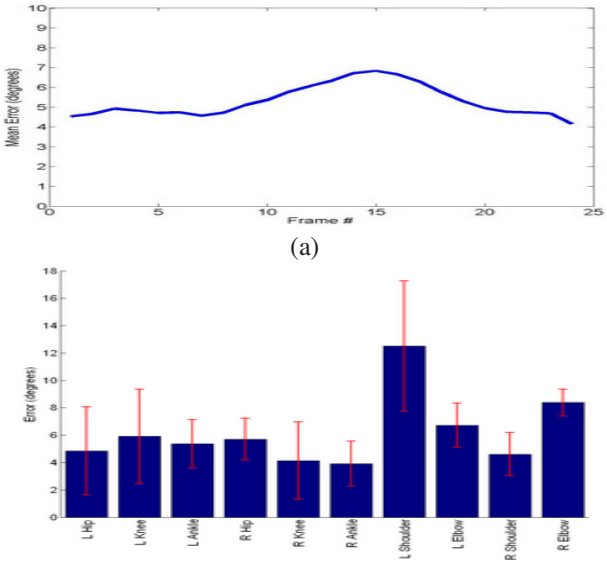
For new sequences in which we want to infer the poses, we first need to recover the trajectory parameters $\mathbf{x}'$. To this end, we track the feet or the hands of the subject in the video using a standard image correlation measure.

In the case of skating, this is made more robust by introducing the knowledge of where the ground plane is. This yields feet trajectories on a 2D rectified plane, which can be automatically split into half-turns. We then obtain the curvatures of the half-turns by fitting a polynomial to the



(a) (b)

Figure 3. (a) Two consecutive skating motions that correspond to the test sequence of Figure 5. The blue dots represent the tracked feet locations on the ground plane and the black line is a second order polynomial fitted to them. The underlying grid is composed of 20cm×20cm squares. The first latent parameter is the curvature of the polynomial, whose sign changes if the subject is turning left or right. The second one is a binary variable indicating if the subject is in the first or second half of the turn. (b) Hands' motion, corresponding to the second golfing sequence of Figure 8. The blue dots depict the tracked hand locations, while the 3 lines show the polynomials fitted to the different phases of the swing, whose 3 curvatures are the latent parameters.



(a) (b)

Figure 4. (a) Average frame-wise error for the sequence of Figure 5, in degrees. (b) Mean errors for different joint angles, in degrees, averaged throughout the sequence. The bars represent the standard deviations of the errors.

trajectories in the same way as for the training data, as depicted in Figure 3(a).

For golf swings, tracking the hands can be made robust by also tracking the golf club, as proposed in [5]. Since the trajectory parameters for the training sequences are estimated in the image plane using the same camera as in the test sequence, we can directly fit the piecewise polynomial to the hand trajectories to obtain $\mathbf{x}'$, as shown in Figure 3(b).

4.3. Motion Recovery

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

To obtain a quantitative evaluation of our results, we filmed some of the motion captured skating sequences. We then removed one sequence from the training data to adopt a leave-one-out validation scheme. We applied our algorithm to this sequence, and measured the reconstruction error as the average of the absolute error over the N_j joint angles that define a pose. This error is plotted frame-wise in Figure 4(a), and has an average value of 5.3 degrees over 24 frames, with a standard deviation of 0.8 degrees. This number of frames corresponds to the time during which the subject was within the capture volume of the Vicon system. In Figure 4(b) we plot the errors for different joint angles, averaged throughout the sequence. We achieve better accuracy on the lower part of the body than on the upper part, because it is much better constrained by the feet trajectory. We show the retrieved pose, both reprojected in the input image and seen from a different viewpoint, in Figure 5.

This ground-truth data also helped us in computing how much accuracy is brought by the refinement step: Without it, the above mean error would have been 6.4 degrees, with a standard deviation of 1.1 degree. Moreover we also made some experiments without using the observable variables to initialize the PCA motion model, in order to compare our approach to [21]. In such paper the PCA weights were all initialized to zero, and doing this on our skating sequence would lead to a mean error of 10.7 degrees with a standard deviation of 1.8 degrees.

To demonstrate that our approach also works in non studio-like environments, we filmed the outdoor sequence of Figure 6 in which the skater is not the one we captured to train the GP. The viewpoint is also very different to show that our approach, being fully 3D, is totally view-independent. Note that the reprojections of our skeleton model correspond well to the underlying images.

Finally, since the skiing motion is very similar to the skating one, we applied our GP trained for skating on the skiing sequence of Figure 7 in which a subject is slaloming between gates. Of course modeling the ground plane on a ski slope is not straightforward. We therefore selected a part of the slalom track that could be roughly approximated by a plane. We then used the GPS coordinates of the gates to warp the 2D trajectory to an orthogonally rectified one, in which we could compute the latent parameters. To this end it would have been enough to have a 3D reference on the ground plane. The results we have obtained are encouraging, but can only be evaluated qualitatively. Nevertheless,

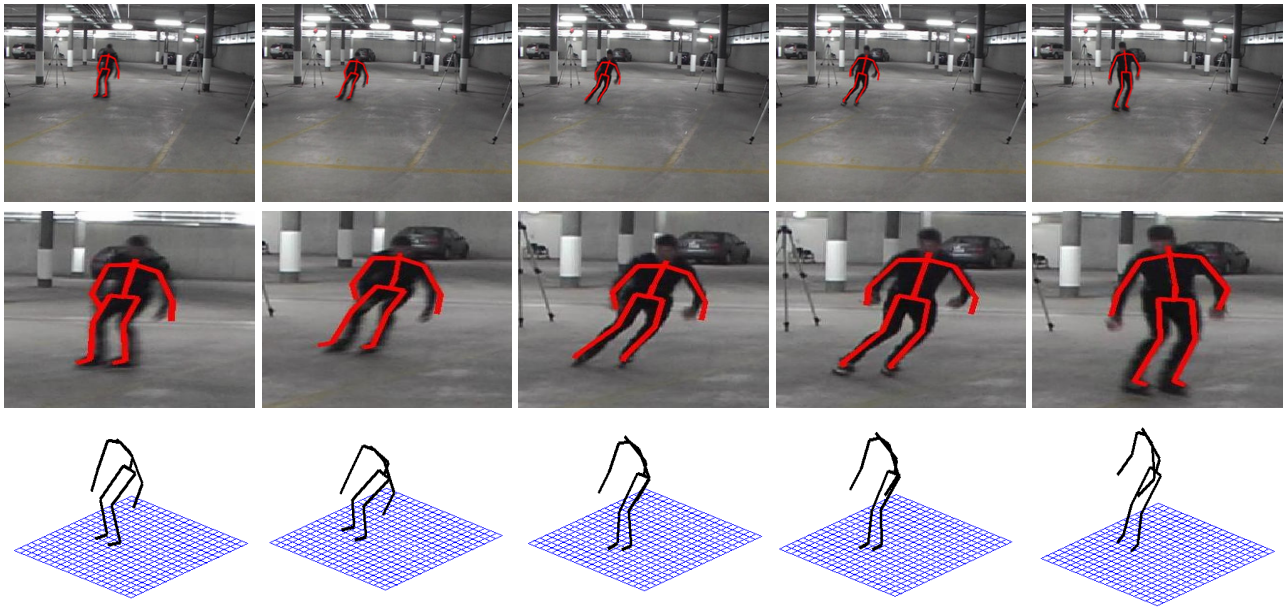


Figure 5. Roller skating in a studio setup. **First row:** We reprojected the recovered body poses in the input images. **Second row:** Zoomed version of the first row. **Third row:** To highlight the 3D nature of the results, we display the 3D skeleton seen from a different viewpoint.

they highlight our method's ability to generalize over the learned motion.

In the case of golf, we show the results obtained when tracking two different subjects, whose motion was *not* captured to build our database, performing a swing. These results are depicted by Figure 8, both overlaid on the input images and seen from a different viewpoint.

5. Conclusions

We have proposed a general technique that uses easily retrievable image measurements as latent variables from which we can recover 3D human body motion via a Gaussian Process mapping. By contrast with state-of-the-art approaches that consider the latent variables as unknowns, learning our mapping involves very few parameters and is therefore much easier to do. It allows us to recover 3D motion from monocular video sequences without having to manually intialize either the poses or the latent variables.

We have demonstrated our approach on challenging activities such as roller skating, skiing, and golfing. A potential extension of this approach would be to look into more complex activities for which some of the latent variables are indeed observable and others not. In such cases, such as when the person's individual style truly matters, we will look at hybrid approaches where we will establish a first mapping using the approach presented here and then learn a second mapping modeling deviations from what the first predicts. Because the first mapping will have captured much of the complexity, it is hoped that the second will be easy to learn, even in these difficult cases.

Acknowledgements We thank Prof. Andrew Zisserman for the insightful inspiration, Prof. Jan Skaloud and Adrian Wägli for providing us with the ski data and Frederic Meyer for helping us in building the skating database.

References

[1] A. Agarwal and B. Triggs. Tracking articulated motion with piecewise learned dynamical models. In *ECCV*, 2004.

[2] T. Brox, B. Rosenhahn, D. Cremers, and H. Seidel. Nonparametric density estimation with adaptive, anisotropic kernels for human motion tracking. In *HUMAN MOTION Understanding, Modeling, Capture and Animation*, 2007.

[3] M. Brubaker, D. Fleet, and A. Hertzmann. Physics-based person tracking using simplified lower-body dynamics. In *CVPR*, 2007.

[4] A. Elgammal and C. Lee. Inferring 3D Body Pose from Silhouettes using Activity Manifold Learning. In *CVPR*, 2004.

[5] N. Gehrig, V. Lepetit, and P. Fua. Golf club visual tracking for enhanced swing analysis tools. In *BMVC*, 2003.

[6] J. Ham, I. Ahn, and D. Lee. Learning a manifold-constrained map between image sets: applications to matching and pose estimation. *CVPR*, 2006.

[7] M. Isard and A. Blake. CONDENSATION – conditional density propagation for visual tracking. *IJCV*, 1:5–28, 1998.

[8] N. D. Lawrence. Gaussian Process Models for Visualisation of High Dimensional Data. In *NIPS*, 2004.

[9] T. B. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *CVIU*, 104(2):90–126, 2006.

[10] R. Navaratnam, A. Fitzgibbon, and R. Cipolla. The Joint Manifold Model for Semi-supervised Multi-valued Regression. In *ICCV*, 2007.

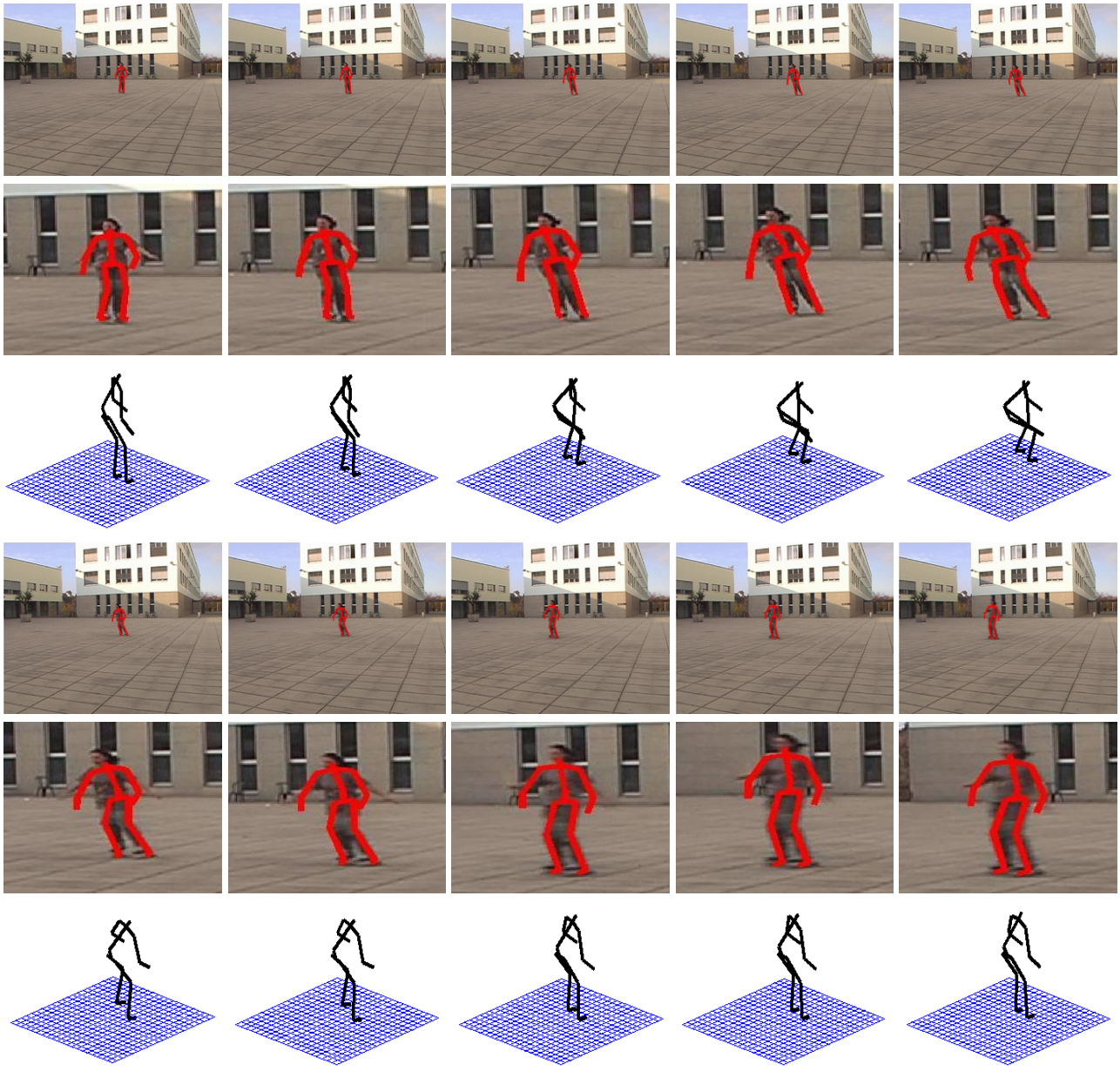


Figure 6. Roller skating. **First and fourth rows:** Recovered body pose reprojected in the input image. **Second and fifth rows:** Zoomed versions of the first and fourth rows, respectively. **Third and sixth rows:** 3D skeleton of the subject seen from a different viewpoint.

[11] D. Ormoneit, H. Sidenbladh, M. Black, and T. Hastie. Learning and tracking cyclic human motion. In *NIPS*, 2001.

[12] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006.

[13] B. Rosenhahn, T. Brox, and H. Seidel. Scaled motion dynamics for markerless motion capture. In *CVPR*, 2007.

[14] A. P. Shon, K. Grochow, A. Hertzmann, and R. P. N. Rao. Learning shared latent structure for image synthesis and robotic imitation. In *NIPS*, 2006.

[15] H. Sidenbladh, M. J. Black, and D. J. Fleet. Stochastic Tracking of 3D human Figures using 2D Image Motion. In *ECCV*, June 2000.

[16] C. Sminchisescu and A. Jepson. Generative Modeling for Continuous Non-Linearly Embedded Visual Inference. In *ICML*, 2004.

[17] L. Taycher, G. Shakhnarovich, D. Demirdjian, and T. Darrell. Conditional Random People: Tracking Humans with CRFs and Grid Filters. In *CVPR*, 2006.

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

[19] R. Urtasun, D. Fleet, A. Geiger, J. Popović, T. Darrell, and N. Lawrence. Topologically-constrained latent variable models. In *ICML*, 2008.

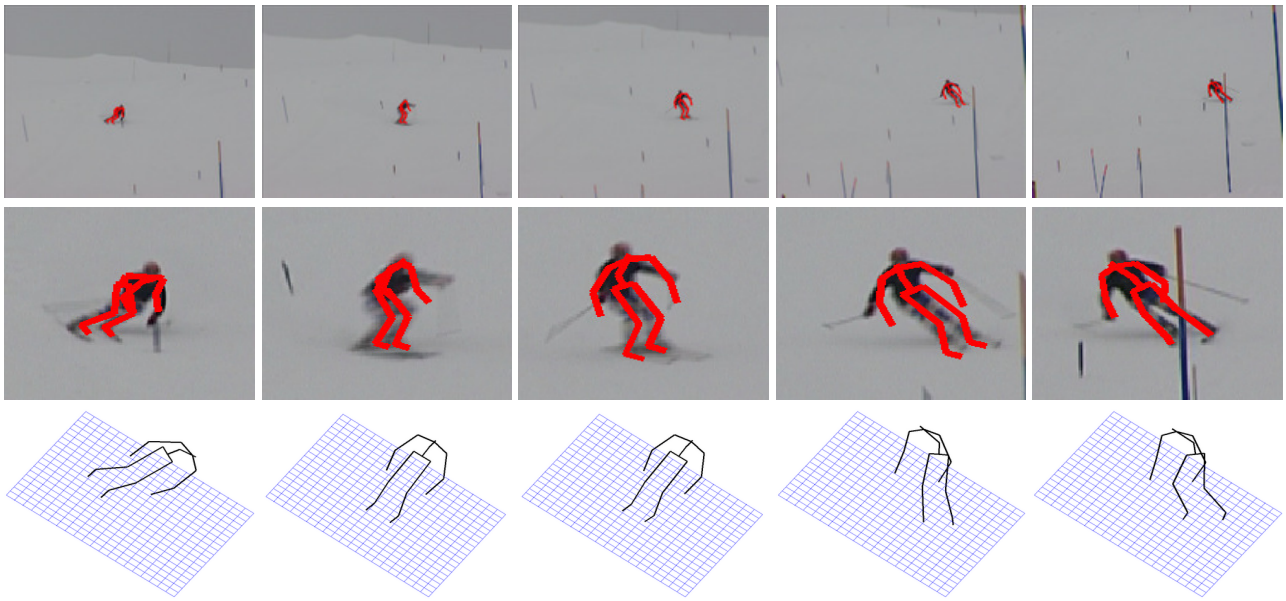


Figure 7. We used the model trained on skating motions to recover a skiing one. **First row:** Recovered body pose reprojected in the input image. **Second row:** Zoomed versions of the first row. **Third row:** 3D skeleton of the subject seen from a different viewpoint.

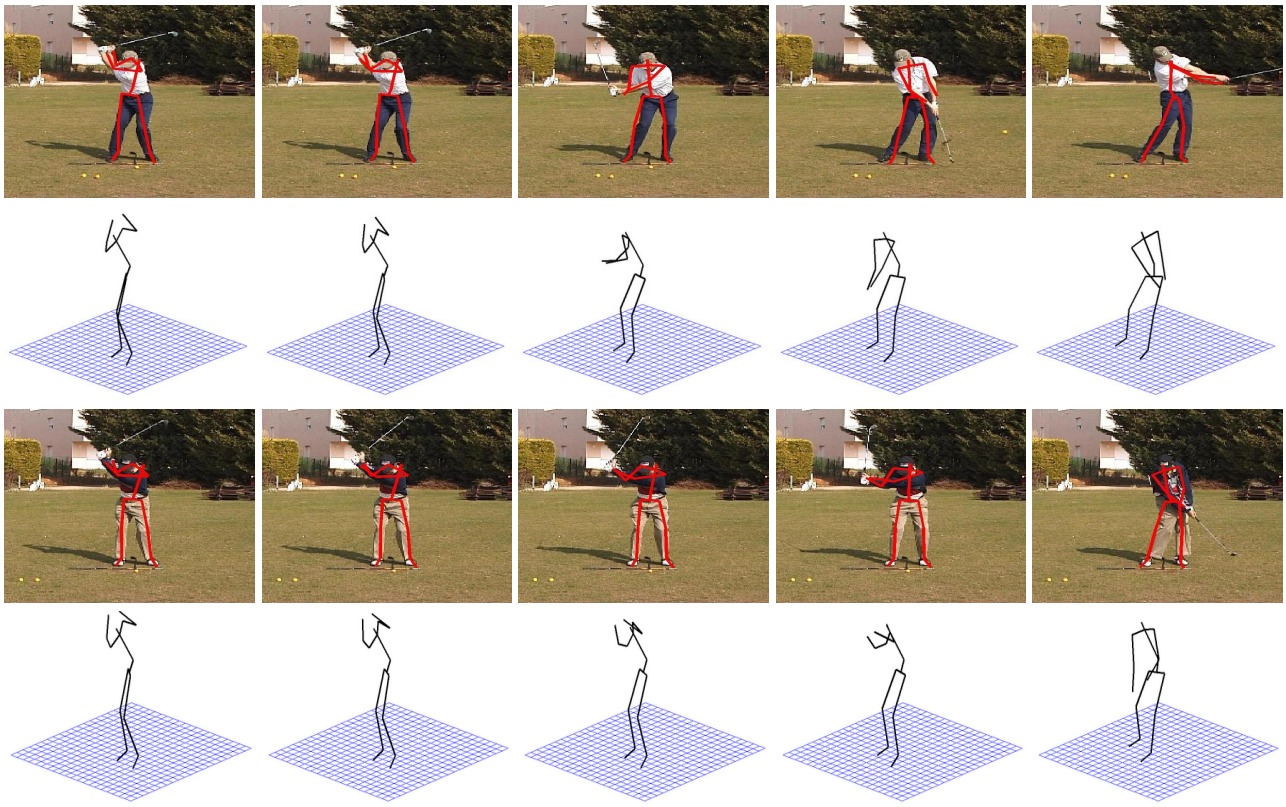


Figure 8. Golf swing tracking. **First and third rows:** Two different subjects performing a golf swing. The recovered body poses have been reprojected in the input images. **Second and fourth rows:** The 3D skeleton of the person is seen from a different viewpoint.

[20] R. Urtasun, D. Fleet, A. Hertzman, and P. Fua. Priors for people tracking from small training sets. In *ICCV*, 2005.

[21] R. Urtasun and P. Fua. 3d human body tracking using deterministic temporal motion models. In *ECCV*, 2004.

[22] Viconpeak. http://www.vicon.com/.

[23] M. Vondrak, L. Sigal, and O. Jenkins. Physical simulation for probabilistic motion tracking. In *CVPR*, 2008.