Mahalanobis Motion Generation

Raquel Urtasun, Pascal Fua *
Computer Vision Laboratory
Swiss Federal Institute of Technology
1015 Lausanne, Switzerland

Pascal Glardon, Daniel Thalmann
Virtual Reality Laboratory
Swiss Federal Institute of Technology
1015 Lausanne, Switzerland

Technical Report No: IC/2004/13

Abstract

Representing motions as linear sums of principal components has become a widely accepted animation technique. While powerful, the simplest version of this approach is not particularly well suited to modeling the specific style of an individual whose motion had not yet been recorded when building the database: It would take an expert to adjust the PCA weights to obtain a motion style that is indistinguishable from his. Consequently, when realism is required, current practice is to perform a full motion capture session each time a new person must be considered. In this paper, we extend the PCA approach so that this requirement can be drastically reduced: For whole classes of motion such as walking or running, it is enough to observe the newcomer moving only once at a particular speed using either an optical motion capture system or a simple pair of synchronized video cameras. This one observation is used to compute a set of principal component weights that best approximates the motion and to extrapolate in real-time realistic animations of the same person walking or running at different speeds.

1 Introduction

Representing motions as linear sums of principal components has become a widely accepted animation technique [5, 1, 22, 3]. These principal components are computed by motion capturing as many people as possible performing a specific activity, representing each motion as a temporally quantized vector of joints angles, and performing a Principal Component Analysis (PCA) on the resulting database of motion vectors. A linear combination of these vectors can then be considered as a valid motion and can be used to produce new animations.

While powerful, the simplest version of this approach is not particularly well suited to modeling the specific style of an individual whose motion had not yet been recorded when building the database: It

*This work was supported in part by the Swiss Federal Office for Education and Science

would take an expert to adjust the PCA weights to obtain a motion style that is indistinguishable from his. Consequently, when realism is required, current practice is to perform a full motion capture session each time a new person must be considered.

In this paper, we show that the PCA approach can be extended so that this requirement can be drastically reduced: For whole classes of motion such as walking or running, it is enough to observe the newcomer walking or running only *once* at a particular speed using either an optical motion capture system or a simple pair of synchronized video cameras. This one observation is used to compute a set of principal component weights that best approximates the motion and to extrapolate in real-time realistic animations of the same person moving at different speeds. This has an important advantage over approaches that simply rely on a linear combination of the captured data to create new styles [18, 17, 12]: Extrapolation allows us to reach a comparatively larger subspace of physically correct motions. Furthermore, these animations can be produced in real-time.

We first validated our approach by exclusively using reliable optical motion capture data: We built a walking database by capturing nine people walking at speeds ranging from 3 to 7 km/h and a running database by capturing five people running at speeds ranging from to 6 to 12 km/h. Given a captured motion that had not been used to perform the PCA decomposition, we project it into PCA space and compute Mahalanobis distances to database motions corresponding to the same speed. These can then be used to synthesize new motions at different speeds and we have verified that these synthesized motions and the actual ones that we have also recorded are both statistically and visually close.

We then replaced the optical motion capture data for the new person by stereo imagery acquired with a cheap and commercially available device [9]. To this end, we take advantage of a technique that we developed in previous work [23] and that lets us track the motion by minimizing a differentiable objective function whose state variables are the PCA weights. This step replaces the projection into PCA space discussed above and allows us again to speed-up or slow-down the motion while preserving the style.

Note that, even though the database we used for validation purposes is specific to two kinds of motions, the approach itself is completely generic. Transposed onto a production set, it has great labor saving potential: The actors' motion need only be captured once to generate a whole range of realistic and personalized animations, thus sparing the need for time-consuming motion capture sessions and expensive gear.

In the remainder of this paper, we will first discuss related work, introduce our motion models and show how they can be used to capture the motion from synchronized videos. We will then introduce our approach to computing PCA weights for observed motions that are not part of the initial motion database and using them extrapolate new ones. Finally, we will validate it using both optical motion capture and video data.

2 Related Work

The literature on walking and running animation is so rich that a full article would be necessary to discuss the advances for walking alone since the last major review of the field [15]. Three main classes of approach can nevertheless be distinguished.

Inverse kinematics. This involves specifying at each key time the corresponding key positions of some joints and obtaining the joint angles according to biomechanical data information. This can be done

efficiently [2] but there is no guarantee of physical realism and this often leads to overly mechanical movements.

Inverse Dynamics. These techniques look for the correct forces and torques to apply to joints to reach a given position. This produces smooth results but may involve postures that are not humanly feasible. It therefore becomes necessary to check and potentially correct these postures by applying appropriate constraints [14].

Motion Capture. New motions are typically created by interpolation. For example, Golam and Wong [10] interpolate walking and running motions in 3–D space where the axes correspond to significant parameters of a locomotion cycle. The method uses bilinear interpolation to synthesize a new motion given four motions having different values for each of the three dimensions. This approach, however, is limited to a small number of input motions and parameters. Multivariate interpolation can be used to solve this problem. Motions are classified manually according to characteristics, leading to a parameter vector for each motion. Similar motions are time-normalized using a time warping process that structurally aligns motions. A new one is generated by applying polynomial and Radial Basis Functions interpolation between the B-Spline coefficients defining the verbs, according to a given parameter vector [18, 17]. Kovar and Gleicher [12] introduce *registration curves* that contain all the information needed to perform blending operations on motions: Timing, coordinates of the root and constraints. The transitions are achieved by simple linear interpolation between two motions.

To synthesize a motion that closely resembles that of a specific person, as opposed to a generic virtual human, using motion capture data is clearly the favored approach because there is no easy way to set the parameters for either IK or ID to achieve the desired goal. However, the drawback of the methods discussed above is that they do not provide for *extrapolation*. As a result, to create a whole range of motions such as those of a specific athlete running at varying speeds, one must perform a full motion capture session of that athlete actually running at a number of different speeds. This can prove cumbersome and the technique we advocate here takes advantage of Principal Components Analysis (PCA) to alleviate this problem by giving our system an extrapolation capability.

PCA [5, 22, 3] has recently been extensively used in motion synthesis. It is also used to compress the data [1] or to emphasize similarities within instances of objects such as heads [4] in order to deform them to change their apparent age or gender. Unfortunately for walking and running motions, the PCA weights have no obvious direct interpretation. Furthermore, it is difficult to adjust them manually to reproduce the style of a specific individual whose motions have not been used to compute the principal components. Our approach therefore goes one step further than existing techniques by automatically extrapolating walking and running animations that are visually and statistically undistinguishable from those measured using actual motion capture.

3 Models for Motion Synthesis and Analysis

In this section, we introduce the motion models we use both to synthesize walking and running animations and to capture such motion from video sequences.

To build walking models, we used a Vicon optical motion capture system [25] to capture nine people walking at speeds ranging from 3 to 7 km/h by increments of 0.5 km/h on a treadmill. Similarly, to build a running model, we captured five people running at speeds ranging from 6 to 12 km/h by increments of 1 km/h. The data was segmented into cycles and sampled at regular time intervals using quaternion

spherical interpolation [19] so that each example can be treated as $N = 100$ samples of a motion starting at normalized time 0 and ending at normalized time 1. An example is then represented by an *angular motion vector* Θ of dimension $N * NDofs$, where $NDofs = 78$ is the number of angular degrees of freedom in the body model. Θ is of the form

$$\Theta = [\theta_{\mu_1}, \ldots, \theta_{\mu_N}] \ , 0 \leq \mu_i < 1 \ , \tag{1}$$

where the θ_{μ_i} represent the joint angles at normalized time μ_i. The posture at a given time $0 \leq \mu_t \leq 1$ is estimated by interpolating the values of the θ_{μ_i} corresponding to postures immediately before and after μ_t.

This process produces $M = 324$ (9 subjects, 7 speeds, 4 cycles) angular motion vectors for walking and 140 (5 subjects, 7 speeds, 4 cycles) for running. We form their covariance matrix and compute its eigenvectors $\Theta_{1 \leq i \leq M}$ by Singular Value Decomposition. Assuming our set of examples to be representative, any other motion vector Θ can be approximated as a weighted sum of the mean motion Θ_0 and the Θ_i:

$$\Theta \approx \Theta_0 + \sum_{i=1}^{m} \alpha_i \Theta_i \tag{2}$$

where the α_i are scalar coefficients that characterize the motion and $m \leq M$. m controls the percentage of the database that can be represented in this manner. This percentage is defined as

$$\sigma = \frac{\sum_{i=1}^{m} \lambda_i}{\sum_{i=1}^{M} \lambda_i} \tag{3}$$

where the λ_i are the eigenvalues corresponding to the Θ_i eigenvectors. It is depicted by Figure 1 (a) as a function of m. It is taken to be 0.9 for the walking and 0.95 for the running databases, given $m \simeq 10$. The posture at time μ_t is computed by interpolating the components of the Θ vector of Eq. 2 as discussed above.

Figure 1 depicts the first three α_i components of the original running motion vectors when expressed in terms of the Θ_i eigenvectors. Note that the vectors corresponding to specific subjects tend to cluster. The walking database exhibit the same clustering behavior but in higher dimension. This is due to the fact that the inter-variability between subjects is smaller than for a running motion.

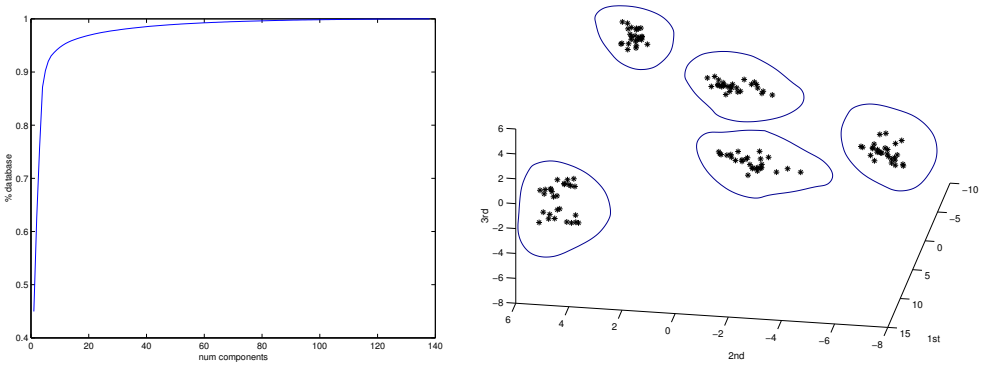
4 Motion Generation

In this section we show how to extrapolate from a motion that is captured after the motion database of Section 3 has been built. This is a two-step process: First, we project the new motion into the PCA space and measure its Mahalanobis distance to each recorded motion of the same speed. The generated motion is then taken to be a weighted average of motions at the target speed with the weights being inversely proportional to those distances.

More precisely, let $\Theta^{p,s}$ be the motion vector of Eq. 1 corresponding to database person p moving at speed s. Each one of these vectors can be approximated by its projection in PCA space $\widehat{\Theta}^{p,s}$ computed as

$$\widehat{\Theta}^{p,s} = \Theta_0 + \sum_i \alpha_i^{p,s} \Theta_i \ , \tag{4}$$

$$\alpha_i^{p,s} = (\Theta^{p,s} - \Theta_0) \cdot \Theta_i \ , \tag{5}$$



(a) (b)

Figure 1: The motion database. (a) Percentage of the database that can be generated with a given number of eigenvalues. (b) Clustering behavior of the first 3 α_i coefficients of Eq. 2 for the 140 motion vectors measured for 5 subjects running at speeds ranging from 6 to 12 km/h. They form relatively compact clusters in 3–D space that can be used for recognition purposes.

where the Θ_i are the principal component vectors of Eq. 2, and the $\alpha_i^{p,s}$ are the scalar coefficients that characterize each motion.

Let Y^{x,s_1} be a motion vector corresponding to motion at speed s_1 by person x who has not been captured before, whose length is arbitrary and from which we wish to extrapolate motion Y^{x,s_2} by the same person moving at speed $s_2 \neq s_1$. As before, we break Y^{x,s_1} into cycles and perform quaternion spherical interpolation [19] to produce a set of Θ^{x,s_1} motion vector of the same dimension as the principal component vectors. By projecting one of these cycles into PCA space, we can compute a set of α_i^{x,s_1} weights analogous to those of Eq. 5.

Because the influence of each Θ_i principal component vector is proportional to the corresponding λ_i eigenvalue, we use the normalized Mahalanobis instead of the Euclidean distance to compare motion vectors. For each p in the database, we therefore take the distance between Θ^{x,s_1} and Θ^{p,s_1} to be

$$d_m(\Theta^{x,s_1}, \Theta^{p,s_1}) = \sqrt{\frac{\sum_i \lambda_i^2 (\alpha_i^{x,s_1} - \alpha_i^{p,s_1})^2}{\sum_i \lambda_i^2}} \quad . \tag{6}$$

Kovar [13] defined a more realistic distance function in terms of the distances between meshes. However the cost of using such distance is prohibitive in our context since we would need to evaluate $P * N$ distances between meshes, where P is the total number of subjects and N he number of frames. The distance proposed in this paper requires only $m * P$ norm evaluations, where m is the number of eigenvalues, allowing a real-time animation of multiple subjects. The only preprocessing needed is the generation of the pca database, which takes a few seconds.

The weights are then taken to be the normalized inverse of these distances

$$w^{x,p} = \frac{[d_m(\Theta^{x,s_1}, \Theta^{p,s_1})]^{-1}}{\sum_q [d_m(\Theta^{x,s_1}, \Theta^{q,s_1})]^{-1}} \tag{7}$$

This completes the interpolation step of our motion synthesis scheme and we are now ready to generate a new motion. If s_2 is one of the speeds recorded in the database, we can simply take the new motion

$\widetilde{\Theta}_{s_1}^{x,s_2}$ to be an extrapolation of Θ^{x,s_1}:

$$\begin{aligned} \widetilde{\Theta}_{s_1}^{x,s_2} &= \Theta_0 + \sum_i \widetilde{\alpha}_i^{x,s_2} \Theta_i \\ \widetilde{\alpha}_i^{x,s_2} &= \sum_p w^{x,p} \alpha_i^{p,s_2} \quad . \end{aligned} \tag{8}$$

Otherwise, to produce smooth transitions between speeds, we take the $\widetilde{\alpha}_i^{x,s_2}$ coefficients to be Cubic Spline Interpolations of those computed as described above.

5 Validation

In this section we use a *cross-validation* framework to validate statistically and visually our approach. For each database subject p in turn, we remove all the $\Theta^{p,s}$ motion vectors that correspond to him or her and perform a new PCA. For any two speeds s_1 and s_2, we can then use the procedure of Section 4 to synthesize $\widetilde{\Theta}_{s_1}^{p,s_2}$ from Θ^{p,s_1} and compare it to $\widehat{\Theta}^{p,s_2}$, the actual projection of the recorded motion. Ideally, the Mahalanobis distances of these two motions should be zero for all p, s_1 and s_2.

In practice, this can of course never be exactly true. As discussed in Section 3, recall that the database contains, for each subject, several motion cycles at the same speed and that they are never exactly similar to one another.

5.1 Animation Results

We now show running and walking animation results obtained by synthetically varying the speed of a motion captured at *one* single speed. These animations are visualized using a time warping technique that produces smooth transitions, as described in [16]. Figure 2 depicts running at speeds increasing from 6 to 12 km/h. For comparison purposes, we superpose the result with an animation obtained by interpolating the actual motion capture data at *all* the relevant speeds. Note that the two synthetic characters are superposed almost perfectly. To highlight the quality of the result, in the bottom of Figure 2, we superpose two animation corresponding to two *different* women. There the differences are obvious. The same phenomenon can be seen in Figure 3 where we plot the alpha coefficients as a function of speed.

Figure 4 depicts a similar behavior for walking at speeds ranging from 3 to 7 km/h. Again, as can be seen in the top row, the motions generated from a single example and those interpolated using a whole set of examples match very well, except for small discrepancies of the arm motion. As will be discussed below, this can be ascribed to the fact that people do not perform the motion twice in exactly the same fashion.

5.2 Statistical Validation

We now introduce the statistical cross-validation framework we use to validate the results shown above. To this end, we define the following quantitative measures:

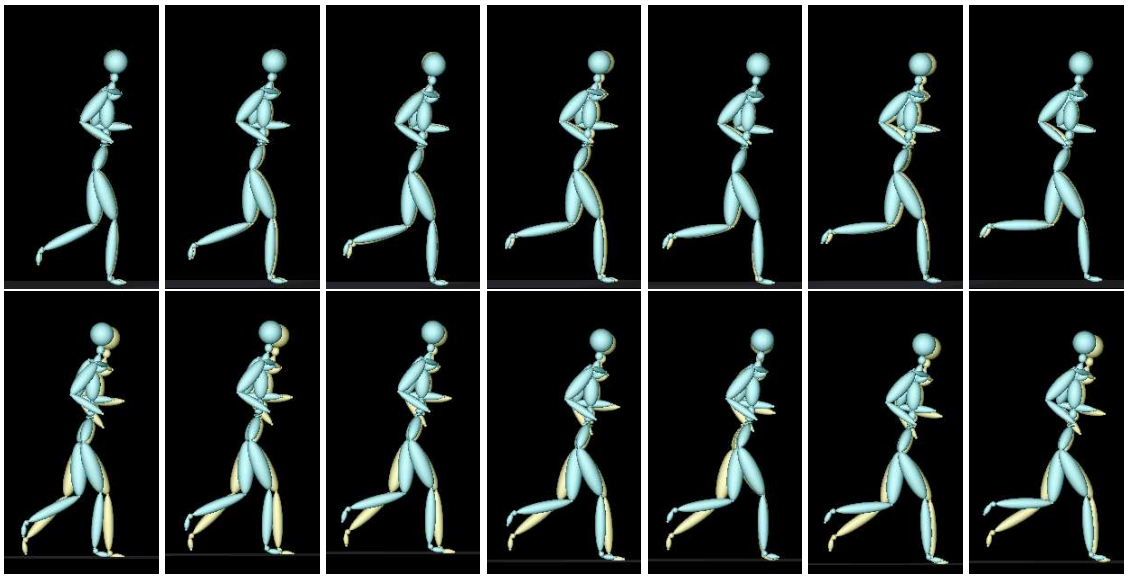


Figure 2: Running at speeds increasing from 6 to 12km/h. **Top row:** Superposition of the synthesized motion generated from a *single* optical motion capture at 6km/h, in yellow, to an animation obtained by interpolating the actual motion capture data at *all* the relevant speeds, in blue. Note that the two synthesized characters are superposed almost perfectly. **Bottom row:** Superposition of the animations corresponding to *two* different women. Note the big differences compare to the results in the top row.

- **Interpolation Error:** The average over all subjects and pairs of speeds of the normalized Mahalanobis distance between the recorded and synthesized motion vectors discussed above.
- **Intra-variability:** The dispersion between different realization by the same subject of the same motion at the same speed. It is taken to be the mean over all subjects and all speeds of the normalized Mahalanobis distance of the $\widehat{\Theta}^{p,s}$ motion vectors corresponding to different cycles.
- **Inter-variability:** The dispersion between different clusters belonging to different subjects. It is calculated as the mean over all subjects of the distance between motion vectors corresponding to different cycles and speeds.

In Figure 5, we give the Interpolation Errors, Intra-variability and Inter-variability values for our walking and running databases. Because we perform cross-validation, we take each person out of the databases in turn and therefore list as many values as there are subjects in each one. Note that the interpolation error is consistently larger than the intra-variability but much smaller than the inter-variability. In other words, our motion generation scheme, while not perfect, nevertheless yields motions that are close enough to those of their rightful owner to be associated with him or her rather to anybody else.

6 From Video to Animation

In this section we show that we can replace the optical motion capture data we have used so far by synchronized video-sequences acquired using an inexpensive commercial product [9]. In the remainder of this section, we first outline briefly the Computer-Vision algorithm [23] we use to extract the PCA coefficients of Eq. 2 from the images. We then show that they are accurate enough to produce valid and realistic motions by used as input to our motion generation scheme.

6.1 Inferring the PCA coefficients from Video

Most recent tracking techniques presented in the Computer Vision literature rely on probabilistic methods to increase robustness [11, 8, 7, 6, 20]. While effective, such approaches require large amount of computation. In previous work [23], we developed an approach that relies on the motion models of Section 3 to formulate the tracking problem as the one of minimizing differentiable objective functions. The body-model is composed of implicit surfaces attached to an articulated skeleton. Each primitive defines a field function and the skin is taken to be a level set of the sum of these fields. Defining surfaces in this manner lets us define a distance function of data points to the model that is easy to evaluate and differentiable. The structure of these objective functions is rich enough to take advantage of standard deterministic optimization methods and, thus, reduce the computational costs.

Given a T frames video sequence in which the motion of a subject remains relatively steady such as those of Figures 8 and 7, the entire motion can be completely described by the angular motion vector of Eq. 2 and, for each frame, a six-dimensional vector G_t that defines the position and orientation of the root body model node with respect to an absolute referential. We therefore take the state vector ϕ to be

$$\phi = [G_1, \ldots, G_T, \mu_1, \ldots, \mu_T, \alpha_1, \ldots, \alpha_m] \ , \tag{9}$$

where the μ_t are the normalized times assigned to each frame and which must also be treated as optimization variables.

In the sequences of Figures 8 and 7, the images we show were acquired using one of the three synchronized cameras used to compute clouds of 3–D points via correlation-based stereo. The ϕ state vector, and thus the motion, were recovered by minimizing in the least-squares sense the distance of the body model to those clouds in *all* frames simultaneously [24]. Note the good performance of the tracking even though the images are of low resolution. Figure 6 shows the first four components recovered by tracking the motion of Figure 8 in the walking database. Note that the recovered coefficients fall squarely in the cluster corresponding to the subject.

6.2 Synthesized animation

The set of coefficients recovered by the computer vision algorithm is used in the same manner as in Section 4 to generate animations of the subjects shown in Figure 8 and 7 walking at different speeds.

Figure 9 depicts a side view of the optical motions captured from 3 to 7km/h (yellow) superposed on the synthesized motion extrapolated from the video-sequence of Figure 8 where the person was walking at a speed of 5km/h. For the legs the correspondance is almost perfect. The small differences in the arms stem from three different reasons: the intra-variability of the walking motion, low resolution of the images in which the arms contain less than 10 pixels, and the fact that the optically captured motions were performed using a treadmill which is not the case for the video-sequence. Note that the motion capture data has not been corrected, which is why the left leg is penetrating into the floor.

Finally, we show how our synthetization framework can generate movements from a large space by tracking a subject that is not part of the database depicted in Figure 7 at 3km/h and by synthesizing his movement from 3 to 7km/h. The motion remains natural and physically possible even though it is *different* from all the recorded motions. In Figure 10, we highlight this difference by superposing the generated motion to the closest one in the database, which is clearly dissimilar. This was to be expected because, as shown in Figure 6, the corresponding alpha coefficients do not match any of the clusters that correspond to specific individuals.

7 Conclusion

We have presented a real-time motion generation technique that allows us to generate the motion of a particular individual walking or running at different speeds given one observation at a specific speed. This observation can be made using either a sophisticated optical motion capture system or a much simpler set of synchronized cameras. We have shown that the resulting animations preserve the person's style both visually and in a statistical sense.

While only validated our approach in the case of two specific kinds of motion, we believe the proposed technique to be fully generic and applicable to a whole range of activities, including those that unlike walking and running are not cyclical. In future work, we will therefore focus on expanding our motion repertoire. This could result in non-linearities, which may require us to replace PCA by more sophisticated statistical tools such as the Isomap [21] that allows the same kind of treatment for non-linear models.

References

[1] M. Alexa and W. Mueller. Representing animations by principal components. In *Eurographics*, volume 19, 2000.

[2] P. Baerlocher and R. Boulic. An Inverse Kinematics Architecture for Enforcing an Arbitrary Number of Strict Priority Levels. *The Visual Computer*, 2004.

[3] V. Blanz, C. Basso, T. Poggio, and T. Vetter. Reanimating Faces in Images and Video. In *Eurographics*, Granada, Spain, September 2003.

[4] V. Blanz and T. Vetter. A Morphable Model for The Synthesis of 3–D Faces. In *Computer Graphics, SIGGRAPH Proceedings*, Los Angeles, CA, August 1999.

[5] M.E. Brand and A. Hertzmann. Style Machines. *Computer Graphics, SIGGRAPH Proceedings*, pages 183–192, July 2000.

[6] K. Choo and D.J. Fleet. People tracking using hybrid monte carlo filtering. In *International Conference on Computer Vision*, Vancouver, Canada, July 2001.

[7] A. J. Davison, J. Deutscher, and I. D. Reid. Markerless motion capture of complex full-body movement for character animation. In *Eurographics Workshop on Computer Animation and Simulation*. Springer-Verlag LNCS, 2001.

[8] J. Deutscher, A. Blake, and I. Reid. Articulated Body Motion Capture by Annealed Particle Filtering. In *CVPR*, Hilton Head Island, SC, 2000.

[9] Digiclops Stereo Vision camera system. Point Grey Research Inc., Vancouver, Canada. http://www.ptgrey.com/products/digiclops/.

[10] A. Golam and K. C. Wong. Dynamic time warp based framespace interpolation for motion editing. *Graphics Interface*, pages 45–52, 2000.

[11] M. Isard. and A. Blake. CONDENSATION - conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, August 1998.

[12] L. Kovar and M. Gleicher. Flexible automatic motion blending with registration curves. In *ACM Symposium on Computer Animation*, pages 214–224, July 2003.

[13] M. Gleicher L. Kovar and F. Pighin. Motion Graphs. In *Computer Graphics, SIGGRAPH Proceedings*, pages 473 – 482, July 2002.

[14] C.K. Liu and Z. Popovic. Synthesis of Complex Dynamic Character Motion from Simple Animations. In *Computer Graphics, SIGGRAPH Proceedings*, 2002.

[15] F. Multon, L. France, M.P. Cani-Gascuel, and G. Debunne. Computer animation of human walking: a survey. *Journal of Visualization and Computer Animation*, 10(1):39–54, 1999.

[16] R. Boulic P. Glardon and D. Thalmann. A Locomotion Engine Based on Hierarchical PCA. In *Submitted to Computer Graphics Forum, Eurographics*, 2004.

[17] S.I. Park, H. J. Shin, and S.Y. Shin. On-line locomotion generation based on motion blending. In *ACM Symposium on Computer Animation*, 2002.

[18] C. Rose, M.F. Cohen, and B. Bodenheimer. Verbs and adverbs: Multidimensional motion interpolation. *Computer Graphics and Applications*, 18(5):32–41, 1998.

[19] K. Shoemake. Animating Rotation with Quaternion Curves. *Computer Graphics, SIGGRAPH Proceedings*, 19:245–254, 1985.

[20] C. Sminchisescu and B. Triggs. Kinematic Jump Processes for Monocular 3D Human Tracking. In *Conference on Computer Vision and Pattern Recognition*, Madison, WI, June 2003.

[21] J.B. Tenenbaum, V. de Silva, and J.C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[22] N. Troje. Decomposing biological motion: A framework for analysis and synthesis of human gait patterns. *Journal of Vision*, 2(5):371–387, 2002.

[23] R. Urtasun and P. Fua. 3D Human Body Tracking using Deterministic Temporal Motion Models. In *European Conference on Computer Vision*, Prague, Czech Republic, May 2004.

[24] R. Urtasun and P. Fua. Human Motion Models for Characterization and Recognition. In *Automated Face and Gesture Recognition*, Seoul, Korea, May 2004.

[25] Vicon Optical Motion Catpure system. Vicon Motion System Ltd., Oxford, UK. http://www.vicon.com/.

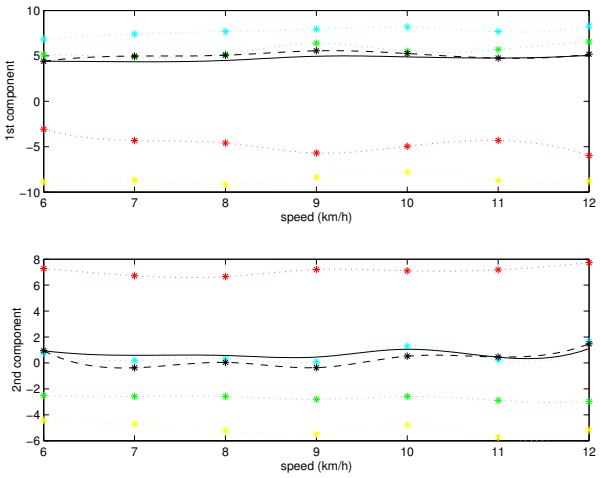


Figure 3: Cubic Spline interpolations of the first two components as a function of the speed for the running database. The synthesized motion depicted in Figure 2, is generated from a single optical motion at 8km/h and is shown in solid black. The original captured motion is depicted in dashed black while the database subjects are shown in different dotted colors.

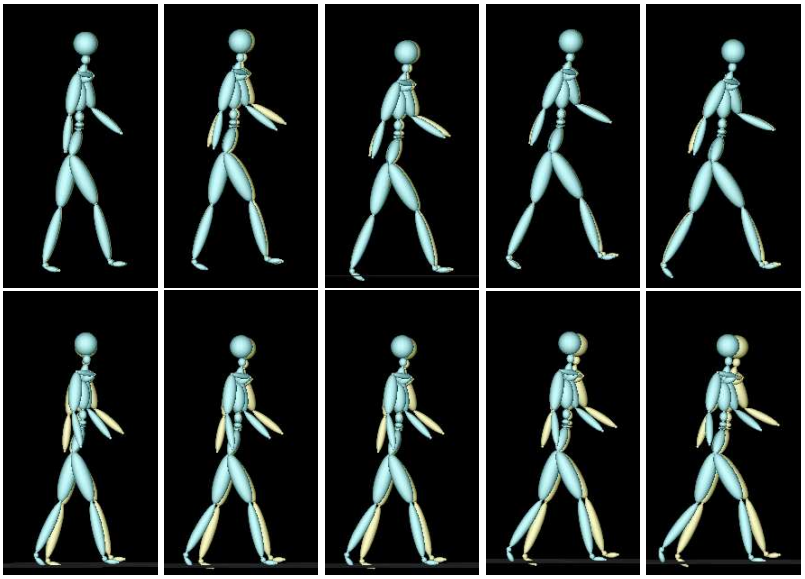


Figure 4: Walking at speeds ranging from 3 to 7 km/h.**Top row:** Superposition of the motions generated from a *single* example, in yellow, to those interpolated using a whole set of examples, in blue. They matched very well except from small discrepancies of the arm motion, that can be ascribed to the fact that people do not perform the motion twice exactly in the same fashion. **Bottom row:** Superposition of the animation corresponding to two different subjects. Note the big differences compare to the results in the top row. Note that the motion captured has not been corrected. This is the reason why the left leg is penetrating into the floor.

	sub_1	sub_2	sub_3	sub_4	sub_5
Intra Variability	0.22	0.21	0.22	0.19	0.20
Inter Variability	3.17	2.45	3.19	3.05	3.97
Interp Error	0.38	0.78	0.92	0.34	0.51

	sub_1	sub_2	sub_3	sub_4	sub_5	sub_6	sub_7	sub_8	sub_9
Intra Variability	0.24	0.21	0.21	0.22	0.23	0.23	0.24	0.23	0.20
Inter Variability	1.17	1.16	1.10	1.22	1.13	1.24	1.20	1.19	1.13
Interp Error	0.35	0.62	0.76	0.47	0.35	0.63	0.49	0.55	0.60

Figure 5: Interpolation Errors, Intra-variability and Inter-variability values. **Top row:** Running. **Bottom row:** Walking. Note that the interpolation error is consistently larger than the intra-variability but much smaller than the inter-variability in both databases.

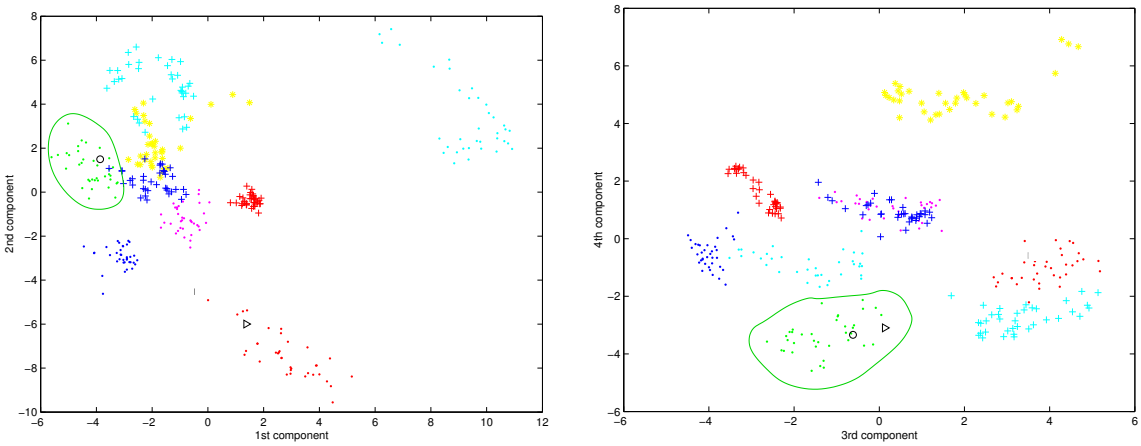


Figure 6: Clustering behavior of the four first pca components for the walking database. Coefficients corresponding to different subjects are depicted of different colors and symbols. The black circle represents the tracking results for the woman of Figure 8, while the man of Figure 7 is shown as a black triangle. Note that the female subject coefficients are inside her cluster, while the male ones do not belong to any given cluster since his motion is not part of the database.

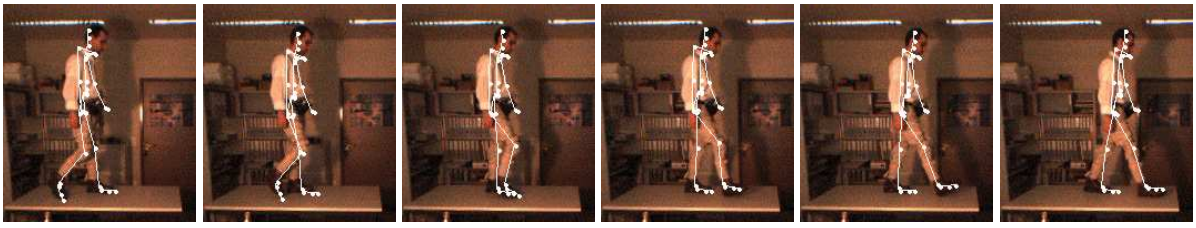


Figure 7: Tracking the walking motion of a man whose motion was *not* recorded in the database. Animation results for this subject are shown in Figure 10.

Figure 8: 3D motion recovered from a video-sequence of a woman whose motion was also recorded in the database. It is displayed as a stick figure projected into the original low resolution video-sequence corres. Corresponding animation results are shown in Figure 9.

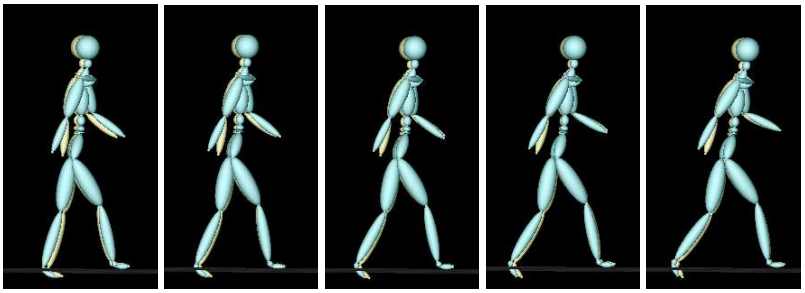


Figure 9: Walking at speeds from 3 to 7 km/h. Superposition of the video of Figure 8, in blue, on those interpolated from a whole set of optical motion capture examples, in yellow. The leg motion matches almost perfectly, which results in the lower body of the two figures being almost perfectly superposed. Small discrepancies in the arms are due to the intra-variability of the motion, the low resolution of the video, and the fact that the optical motions were performed using a treadmill. Note that the motion capture data has not been cleaned-up, which is why the left leg is penetrating the floor.

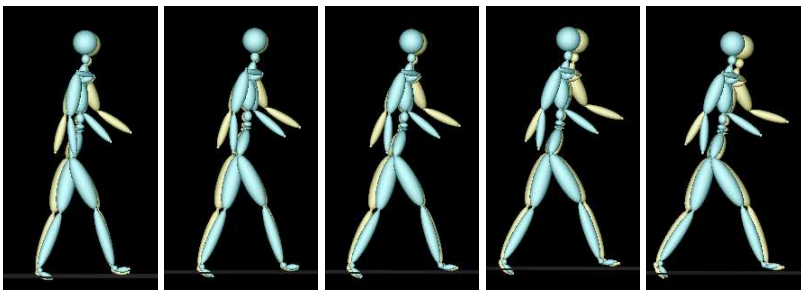


Figure 10: Side view of the original and synthesized motion from 3 to 7 km/h for the subject of the the video sequence of Figure 7. The subject is compared to its closest neighbor in the database according to the Mahalanobis distance. The original motion capture data is represented in blue, and the synthesized one in yellow. They are quite different, which goes to show that our approach can generate a wide range of realistic styles.