Data-Driven Motion Reconstruction Using Local Regression Models

Christos Mousas, Paul Newbury, Christos-Nikolaos Anagnostopoulos

Data-Driven Motion Reconstruction Using Local Regression Models

Christos Mousas[1] Paul Newbury[1] Christos-Nikolaos Anagnostopoulos[2]

[1]Department of Informatics
University of Sussex
Brighton BN1 9QJ, UK
{c.mousas,p.newbury}@sussex.ac.uk

[2]Department of Cultural Technology and Communication
University of the Aegean
Mytilene 81100, Greece
canag@ct.aegean.gr

Abstract. Reconstructing human motion data using a few input signals or trajectories is always challenging problem. This is due to the difficulty of reconstructing natural human motion since the low-dimensional control parameters cannot be directly used to reconstruct the high-dimensional human motion. Because of this limitation, a novel methodology is introduced in this paper that takes benefit of local dimensionality reduction techniques to reconstruct accurate and natural-looking full-body motion sequences using fewer number of input. In the proposed methodology, a group of local dynamic regression models is formed from pre-captured motion data to support the prior learning process that reconstructs the full-body motion of the character. The evaluation that held out has shown that such a methodology can reconstruct more accurate motion sequences than possible with other statistical models.

Keywords: character animation, local regressions, motion reconstruction

1 Introduction

Full-body motion reconstruction is a process that is quite important in cases in which the ability to animate virtual characters while using a reduced number of sensors or user defined trajectories is necessary. Such techniques, especially those that are developed to reconstruct the motion of the character during the performance capture process can be quite beneficial in various areas that are related to virtual reality, such as rehabilitation and sports training, as well as in video games. Although various motion capture systems for capture of the user's performance, such as Vicon [1] and XSens [2], can provide desirable results, the basic limitation is the high cost for general family use. Recently, low-cost commercial products, such as those provided by Microsoft, Sony, and Nintendo, have developed next generation hardware devices to capture the online performances of individual players. However, a reduced number of input signals retrieved from those devices cause the motion reconstruction process to be challenging.

The reason is that human motion has many degrees of freedom (DOF) and, therefore, motion models are required that can reconstruct the natural and realistic motion of a character while using few parameters.

In the proposed methodology, the motion reconstruction process is formulated in a maximum a posteriori (MAP) framework that is responsible for producing a natural-looking motion sequence that best matches the user-defined inputs. Specifically, the proposed methodology learns a group of local regression models in order to constrain the prior learning of the pre-captured motion data. Then, by searching within the motion database, by using K nearest motion examples that are similar to the previously reconstructed poses $q_{t-1},...,q_{t-m}$, and the motion sequences $q_{t_{k-1}},...,q_{t_{k-m}}$ along with their subsequent poses q_{t_k} for $k = 1,...,K$ as the training data, it learns a predictive model for the reconstruction of the current character's posture q_t.

The proposed methodology can reconstruct a variety of motion sequences by using a reduced number of input trajectories, such as walking, running, jumping, and punching, as well as golf swings. Further, by evaluating the presented methodology with previous solutions for reconstructing the character's motion, the proposed approach can reconstruct motion sequences by reducing the reconstruction error. The remainder of this paper is organized as follows: Section 2 presents related work in data-driven character animation and motion reconstruction. Section 3 provides an overview of the proposed methodology. The proposed reconstruction methodology is explained in Section 4. Section 5, presents the results obtained from the evaluations of the proposed methodology versus those oft previously examined techniques. Finally, conclusions are drawn and the potential future work is discussed in Section 6.

2 Related Work

In data-driven motion synthesis techniques [3], the low-dimensional control signals that are obtained from the motion capture device are used to retrieve suitable motion sequences from a database that contains high-dimensional motion capture sequences. The reuse of pre-recorded human motion data requires efficient retrieval of similar motions from databases [4][5], as well as a good understanding of how motions must be parameterized in order to yield smooth transitions between several retrieved motion clips [6].

On the other hand, statistical motion models are often described as several mathematical functions that represent human motion by a set of parameters that are associated with probability distributions [7]. So far, using pre-captured motion data to learn statistical motion models have been used for full-body character control [8], in key frames interpolation [9], motion styles synthesis [10][11], facial animation [12] and speech-driven facial expressions [13][14], hands-over animation techniques [15][16], interactive creation of a character's pose [8][17] or control of human actions using vision-based tracking [18], real-time human motion control with inertial sensors [19] or accelerometer sensors [17], construction of physically-valid motion models for human motion synthesis [14] and so forth.

During the past years, a number of researchers have developed approaches that use sparse constraints provided by sensors to control high-dimensional human motions. A

single depth camera to track and reconstruct various human motions, their approaches acquired no makers attached on user's body, however, no less than 15 control points needed to be used to segment the human body [20][21]. By combining inverse kinematics algorithms and a few constraints from eight magnetic sensors to provide an analytic solution for human motion control [22]. By using six to nine retro-reflective markers as the control points for online human motion reconstruction [18], and by using five inertial sensors for real-time upper-body control [23]. Recently, full-body human motion control was achieved using the positional and orientational constraints from six inertial sensors [19]. Finally, another solution [17] utilizing a few constraints provided by four accelerometer sensors achieved full-body motion control of the character.

3 Overview

In the proposed methodology a reduced number of input trajectories retrieved from a reference motion sequence are used to reconstruct the full-body motion of the character. Those input trajectories automatically transform the control inputs into realistic human motion by building a group of online local regression models during the runtime. An overview of the methodology appears in Figure 1.

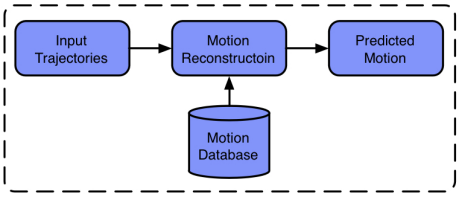

Fig. 1. An overview of the proposed system.

For the motion reconstruction process, it is assumed that the character's actions can be represented as an $m-order$ Markov chain, such as the current posture of the character at the $t-th$ frame q_t can be considered to depend only on previous m reconstructed postures $Q_{t,m} = [q_{t-1}, ..., q_{t-m}]$. Thus, the probabilistic model should be able to fulfill $p(q_t|q_{t-1}, ..., q_1) = p(q_t|q_{t-1}, ..., q_{t-m})$. In the proposed local regression modelling methodology, the spatiotemporal directions to the models to constrain the transformation of the character's postures in the configuration space are added. This approach predicts how the humans move in each region and constrains the reconstructed motion to remain in the natural-looking space. For that reason, it was considered to be an online learned model to generate the desired posture q_t from various forms of kinematic constraints c_t that are specified by the input trajectories. Finally, the motion reconstruction process is optimized in a MAP framework by estimating the posture q_t that is satisfied by the input trajectories c_t along with the previous reconstructed postured $Q_{t,m}$ such as:

$$\begin{aligned} &\arg\max_{q_t} p(q_t|c_t, Q_{t,m}) \\ &\propto \arg\max_{q_t} p(c_t|q_t)p(q_t|Q_{t,m}) \end{aligned} \tag{1}$$

In this case, by applying the negative logarithm to the posteriori distribution function $p(q_t|c_t, Q_{t,m})$, the constrained MAP problem becomes an energy minimization problem, which is now represented as:

$$arg\max_{q_t} \underbrace{-\ln p(c_t|q_t)}_{E_{likelihood}} + \underbrace{-\ln p(q_t|Q_{t,m})}_{E_{prior}} \tag{2}$$

where the likelihood term ($E_{likelihood}$) measures how well the reconstructed posture q_t fits the user defined input trajectories c_t, and the prior term (E_{prior}) measures the naturalness of the reconstructed posture. It should be noted that an optimal estimation of the reconstructed posture produced a natural motion that achieved the inputs that were specified by the user.

4 Motion Reconstruction

Using a reduced number of input trajectories to reconstruct the motion of the virtual character is quite challenging, since the control inputs cannot fully constrain the entire human motion to remain in the natural-looking space. Thus, in the proposed solution, a group of local regression models are used to solve this issue of motion reconstruction ambiguity. The methodology that used is presented in the following subsections.

4.1 Prior Motion Modelling

The prior motion modelling that was based on the local regression models is presented in this subsection. This model is responsible for adequately constraining the reconstructed posture of the character to remain in the natural-looking space. The novelty of the proposed model is that there is no need to find an appropriate structure for a global dynamic model, which would necessarily by high dimensional and non-linear. In the proposed methodology, a k-nearest neighbour ($k-NN$) searching algorithm is adopted to find the K motion examples that are contained in our motion capture database and are similar to the already reconstructed posture. The examples and their subsequent postures are used for our online learning process.

In order to estimate the current posture q_t at the $t-th$ frame, a searching of the database of the motion data is employed in the first steps. This searching process finds the motion segments that are most similar to the recently reconstructed motion segment $Q_{t,m} = [q_{t-1}, ..., q_{t-m}]$. Thus, the k nearest motion segments q_{t_k} for $k = 1, ..., K$ are chosen as training data to learn a predictive model by means of a statistical learning method of current posture q_t.

Now, assume a linear relationship between an input angle vector $x = [q_{t-1}, ..., q_{t-m}]$ and an output joint angle vector $y = q_t$. In this case, one should note that the predictive

function for each DOF in the output q_t is learned separately. Then, by subtracting the means from the input and output training data, one assumes that the mean values of x and y are zeros. Therefore, the function of the proposed model is represented by using linear regression as follows:

$$y = a^T x + \beta_y \tag{3}$$

where the input joint angle x is an $m \times D$-dimensional vector. D represents the dimension of DOF for a virtual character and y is the joint angle value for the output motion. Vectors a and β_y are regression coefficients that represent a homoscedastic noise variable, which is independent of vector x. Moreover, given the K motion examples $(x_k; y_k)$ for $k = 1, ..., K$ that are similar to the current reconstructed poses, and by minimizing the expected squared error $E = \min \sum_{k=1}^{K} \|y_k - a^T x_k\|^2$, the coefficient a is obtained by the least squares solution that follows:

$$a = (X^T X)^{-1} X^T y \tag{4}$$

where the row of the matrix X stores the input joint angle vectors x_k for $k = 1, ..., K$, and K output joint angle values are stacked in vector y.

The proposed methodology calculates the projections of the highest correlation between the input joint angle matrix X and the output vector y. These projections can be obtained by maximizing the squared relationship as follows:

$$correlation^2(X_{u_j}, y) = \frac{(u_j^T X^T y)}{(u_j^T X^T X_{u_j})} \tag{5}$$

where u_j denotes the one of the projection's directions. It should be noted that since each of the projections X_{u_j} is orthogonal to others and its length is unit, it is possible to get $u_j^T X^T X_{u_j} = 1$. Hence, u_j is one column of the matrix U that includes the eigenvectors of the covariance matrix $C = (X^T X)^{-1} X^T y y^T X$. In the proposed model, X is projected onto U for only considering the projections. Thus, by minimizing $\|y - XU\gamma\|^2$ with respect to the reduced coefficient γ, one obtains:

$$a = U\gamma = U(U^T X^T X U)^{-1} U^T X^T y \tag{6}$$

Since each DOF is predicted separately in output q_t, the model has only one projection direction u for each time. The weight for each data point x_k can now be calculated by the Gaussian function, using its relative distance from the previously reconstructed postures $Q_{t,m} = [q_{t-1}, ..., q_{t-m}]$ as:

$$\omega_k = \exp\Big(-\frac{1}{2}(x_k - Q_{t,m})^T W (x_k - Q_{t,m}) \Big) \tag{7}$$

where W denotes the diagonal matrix that contains the weights for each DOF. It should be noted that in the proposed implementation used an identity matrix to represent W. The eigenvectors are extracted from the matrix as:

$$C_\omega = (DX)^{-1} X^T D y y^T D X \tag{8}$$

where D denotes a diagonal matrix that constrains ω_k along its diagonal. Moreover, the weighted regression coefficients can be represented as:

$$a_\omega = U(U^T X^T DXU)^{-1} U^T X^T Dy \tag{9}$$

In this case, assuming that there is a Gaussian distributed noise variable β_y, its standard deviation σ can be estimated by $y_k - \beta x_k$ for $k = 1, ..., K$. In our experiments, a predictive function for each DOF of the reconstructed posture is constructed. Therefore, to predict the $d-th$ DOF of the character's posture, the local regression model is described as:

$$q_{t,d} = a_{d,\omega}^T Q_{t,m} + N(0, \sigma_d) \tag{10}$$

where $q_{t,d}$ and σ_d are scalars, $q_{t,d}$ represents the $d-th$ DOF of the $t-th$ frame posture, and σ_d is the standard deviation of the $d-th$ predictive function. $a_{d,\omega}^T$ and $Q_{t,m}$ are vectors, where $a_{d,\omega}^T$ are the weighted regression coefficients for the $d-th$ DOF, and $Q_{t,m}$ is the reconstructed motion segment of the previous m postures of the character. The complexity of such a model for reconstructing the character's posture is $O(Km^2D^2)$, where K, m, and D represents the number of training data, the previous m postures, and the dimension of DOF for the virtual character repsectivelly.

4.2 Likelihood Estimation

The likelihood term ($E_{likelihood}$) of the MAP framework measures how well the corresponding joint in the reconstructed character's postures fits the user-defined constraints. Therefore, the likelihood term is formulated as:

$$\begin{aligned} E_{likelihood} &= -lnp(c_t|q_t) \\ &\propto \|f(q_t;s) - c_t\|^2 \end{aligned} \tag{11}$$

where q_t, c_t, and s are vectors. q_t denotes the joint angles of the reconstructed posture at frame t, s denotes the character's skeletal size, which is modelled according to the Acclaim Skeletal File format (ASF) as provided by [24]. Finally, c_t is the user-specified input trajectories that are retrieved from reference motion data. Finally, f denotes a forward kinematics function that calculates the global coordinates value of the current posture q_t of the character.

5 Implementation and Results

The following subsections briefly present the implementation and the results obtained from the evaluation process of the proposed methodology.

5.1 Implementation

For the implementation of the proposed methodology a gradient-based optimization was used using the Levenberg-Marquardt method [25] for the objective function that is defined in Equation (2). This method uses the most similar motion examples that are

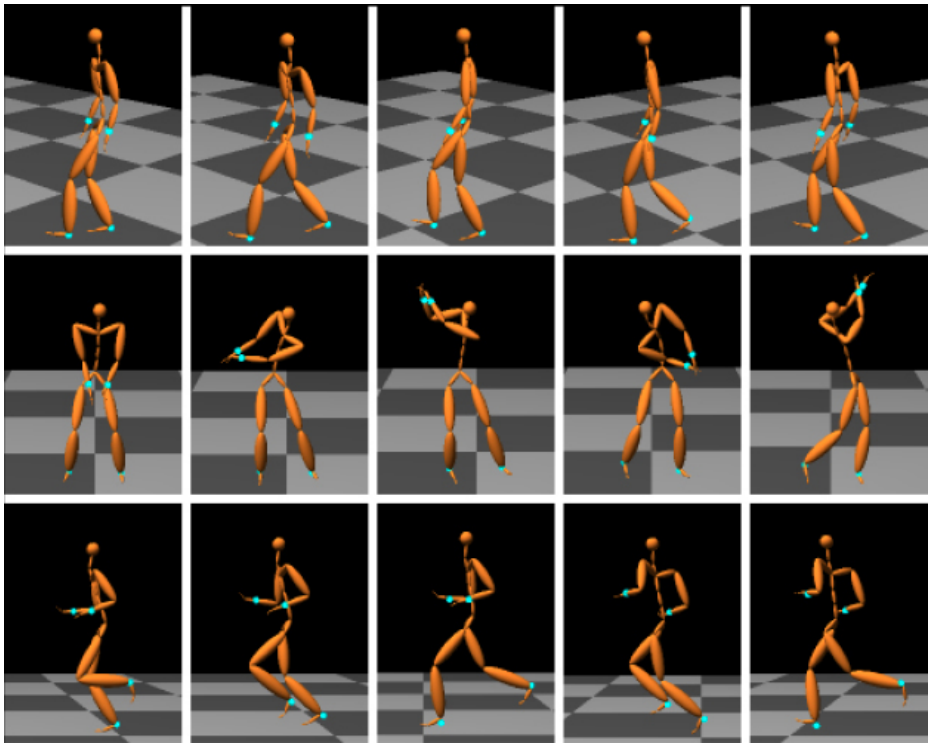

Fig. 2. Examples of postures synthesized with the proposed methodology.

already in the motion database to initialize the optimization. The computational efficiency of the proposed motion reconstruction process relies on the scope of the search in the motion database. Thus, the process to find the K nearest neighbour is accelerated using the neighbour graph approach as presented in [18]. Finally, examples of posture reconstructed with the proposed methodology are shown in Figure 2.

5.2 Results

To evaluate the proposed methodology two different datasets were used. The first dataset contains 85,097 postures that are separated into five different actions that the character can perform: walking, running, jumping, punching, and swinging a golf club. The second dataset contains a total of 1.1 M poses that were downloaded from [24]. All of the motion sequences were recorded by use of a Vicon motion capture system that has a framerate of 120 fps. In the proposed implementation, the motion data were downsampled to 60fps in order to achieve more natural-looking motion for visualization. The effectiveness of the proposed approach was verified on various behaviours and the reconstruction error was evaluated against the ground truth data.

Comparing to Other methods: The proposed methodology was evaluated against three popular approaches for reconstructing the character's motion. Specifically, the methodology was evaluated against the Gaussian Process Latent Variable Model [17], the local PCA model [18] and the local PCR model [19], while reconstructing different actions that the character can perform. The results of this evaluation process appear in Figure 3. Specifically, the mean error of five different actions for all of the aforementioned techniques is presented. In the evaluation, we also adopt six constraint points that were used in [19]. The results show that our method achieved smaller mean errors than the other three techniques. In another aspect, while using only four control points (two wrists and two ankles), the three previous methods cannot reconstruct natural-looking human data. The results indicate that the proposed method is better than those of the two other local methods.

Using Different Number of Control Points: We tested a different number of control points for four methods. We chose from two to six positional control points. (1) Left wrist and right ankle; (2) left wrist and two ankles; (3) two wrists and two ankles; (4) root, two wrists and two ankles; (5) head, root, two wrists and two ankles. After testing with different motions, we concluded that the reconstruction errors usually decrease as the number of constraints increases. In addition, we also found that, in comparison to the six constraint points (head, center of torso, two wrists and two ankles), which was used in [19] to obtain a natural-looking, reconstructed human motion, we can use as few constraint points as possible (4 constraint points: two wrists and two ankles) to achieve a comparable result with the motion capture data. Table 1 is the average reconstruction error comparison for various numbers of control points. Despite the use of fewer constraint points, our model is more powerful for accurate motion reconstruction than the three previous methods.

Table 1. Reconstruction errors based on different numbers of control points for different algorithms.

	2	3	4	5	6
GPLVM	56.76	41.88	16.13	9.45	5.92
LPCA	43.27	29.25	10.56	6.39	3.81
LPCR	38.71	23.61	7.33	4.73	3.22
Proposed Method	18.63	7.86	2.67	2.25	1.90

Using Different Datasets: Table 2 presents the average reconstruction errors of five different actions from the three aforementioned techniques, while using the different training database. The reconstruction errors were calculated using 3D positional constraints from six control points. We found that, for the GPLVM method, the reconstruction error is large when using a large and heterogeneous database. When using a small database, the reconstruction error is also larger than that of local modeling approaches. For the other local modeling methods, the reconstruction error decreased when the size of training database increased. In addition, our proposed model can achieve a smaller

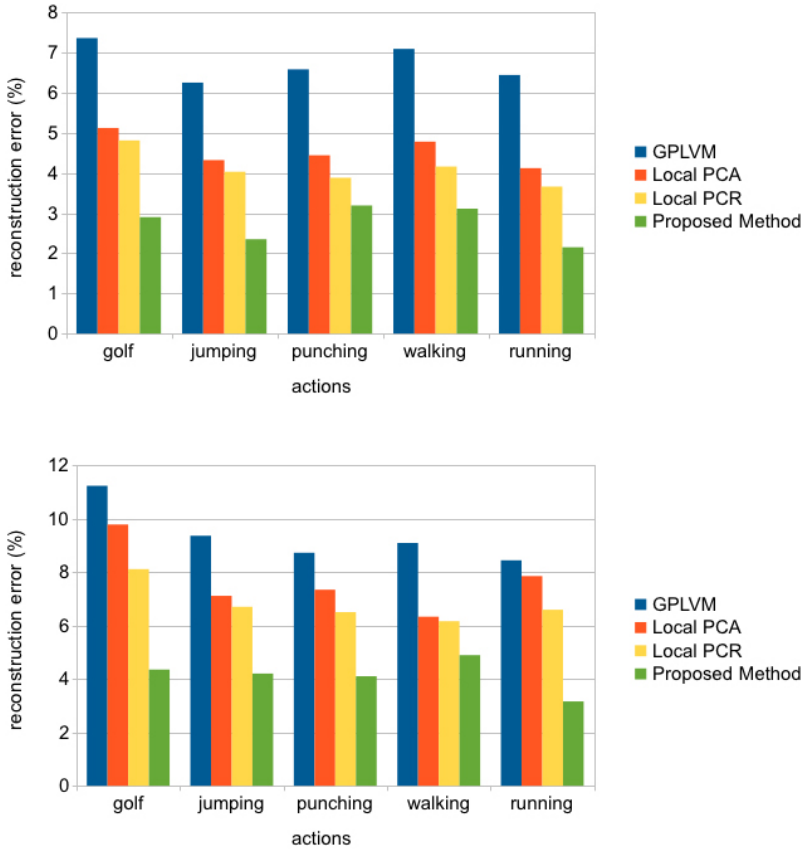

Fig. 3. A comparison of the proposed methodology to three popular algorithms: GPLVM, local PCA, and local PCR. The percentage error while using six (up), and four joints (down) for reconstruction of the motion of the character.

reconstruction error than others. By testing on different databases, we also verified the proposed model's power.

6 Conclusion

In this paper, a new local regression model was presented for reconstructing natural full-body human motion based on as few user-specified constraints as possible. The proposed method, which uses a data-driven approach, utilizes several nearest motion examples to construct a group of online local regression models for online motion reconstruction. However, based on the same defined constraints and motion database, the proposed method has a better force of constraint than the previous local models and thus

Table 2. Average reconstruction errors for four methods on different databases.

	69888 poses	1.1 M poses
GPLVM	5.86	26.57
LPCA	3.92	3.02
LPCR	3.38	2.75
Proposed Method	2.01	1.56

can reconstruct more realistic human motions. Therefore, our proposed model is suitable for the next generation of hardware devices to exploit the motion capture system for a common use.

On the other hand, the proposed method has three limitations. Firstly, like other data-driven approaches, the database is crucial for the quality of reconstructed motion. The system will not produce a desired motion if the training data does not contain any desired motion patterns. For example, if the walking motion pattern is not included in the database, our system cannot reconstruct desired walking data. Secondly, user-specified constraints are also crucial for the final results. In fact, if user-specified constraints are not natural or self-conflicting, the reconstruction result will not be a realistic human motion that satisfies the user's constraints. Finally, the motion data must be previously arranged for online search. Like most local modeling approaches, a specific data arrangement structure is applied for motion data to accelerate the searching process.

References

1. Vicon Motion Capture Solution, from `http://www.vicon.com/`, accessed 10/05/2014
2. Xsens Motion Capture Solution, from `http://www.xsens.com/`, accessed 10/05/2014
3. Liu, H., He, F., Cai, X., Chen, X., Chen, Z.: Performance-Bbased Control Interfaces Using Mixture of Factor Analyzers. The Visual Computer, 27(6-8), pp. 595–603. Springer (2011)
4. Keogh, E., Palpanas, T., Zordan, V. B., Gunopulos, D., Cardle, M.: Indexing Large Human-Motion Databases. In: 30th International Conference on Very Large Databases, pp. 780–791 (2004)
5. Müller, M., Röder, T., Clausen, M.: Efficient Content-Based Retrieval of Motion Capture Data. ACM Transactions on Graphics. 24(3), pp. 677–685 (2005)
6. Kovar, L., Gleicher, M.: Flexible Automatic Motion Blending with Registration Curves. In: ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 214–224. Eurographics Association (2003)
7. Mousas, C., Newbury, P., Anagnistopoulos C.-N.: Evaluating the Covariance Matrix Constraints for Data-Driven Statistical Human Motion Reconstruction. In Proc. of the 30th Spring Conference on Computer Graphics, ACM Press, New York (2014)
8. Wei, X. K., Chai, J.: Intuitive Interactive Human-Character Posing with Millions of Example Poses. IEEE Computer Graphics and Applications, 31(4), pp. 78–88. IEEE (2011)
9. Li, Y., Wang, T., Shum, H. Y.: Motion Texture: a Two-Level Statistical Model for Character Motion Synthesis. ACM Transactions on Graphics, 21(3), pp. 465–472. ACM Press, New York (2002)
10. Brand, M., Hertzmann, A.: Style Machines. In: 27th Annual Conference on Computer Graphics and Interactive Techniques, pp. 183–192. ACM Press, New York (2000)

11. Mousas, C., Newbury, P., Anagnistopoulos C.-N.: Motion Style transfer in Correlated Motion Spaces. In Proc. of the 12nd International Symposium on Smart Graphics, Springer (2014)
12. Weise, T., Bouaziz, S., Li, H., Pauly, M.: Realtime Performance-Based Facial Animation. ACM Transactions on Graphics, 30(4), Article No.77. ACM Press, New York (2011)
13. Bregler, C., Covell, M., Slaney, M.: Video Rewrite: Driving Visual Speech with Audio. In: 24th Annual Conference on Computer Graphics and Interactive Techniques, pp. 353–360. ACM Press, New York (1997)
14. Brand, M.: Voice Puppetry. In: 26th Annual Conference on Computer Graphics and Interactive Techniques, pp. 21–28. ACM Press, New York (1999)
15. Mousas, C., Newbury, P., Anagnistopoulos C.-N.: Efficient Hand-Over Motion Reconstruction. In Proc. of the 22nd International Conference on Computer Graphics, Visualization and Computer Vision. (2014)
16. Wheatland, N., Jörg, S., Zordan, V.: Automatic Hand-Over Animation Using Principle Component Analysis. In: Motion on Games, pp. 175–180. ACM Press, New York (2013)
17. Grochow, K., Martin, S. L., Hertzmann, A., Popović, Z.: Style-Based Inverse Kinematics. ACM Transactions on Graphics, 23(3), pp. 522–531. ACM Press, New York (2004)
18. Chai, J., Hodgins, J. K.: Performance Animation from Low-Dimensional Control Signals. ACM Transactions on Graphics. 24(3), pp. 686–696. ACM Press, New York (2005)
19. Liu, H., Wei, X., Chai, J., Ha, I., Rhee, T.: Realtime Human Motion Control with a Small Number of Inertial Sensors. In: Symposium on Interactive 3D Graphics and Games, pp. 133–140. ACM Press, New York (2011)
20. Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A.: Real-Time Human Pose Recognition in Parts from a Single Depth Image. In: IEEE Conference on Computer Vision and Pattern Recognition,pp. 1297–1304. IEEE Press (2011)
21. Wei, X., Zhang, P., Chai, J.: Accurate Realtime Full-Body Motion Capture Using a Single Depth Camera. ACM Transactions on Graphics,31(6), Article No. 188. ACM Press, New York (2012)
22. Semwal, S. K., Hightower, R., Stansfield, S.: Mapping Algorithms for Real-Time Control of an Avatar Using Eight Sensors. Presence: Teleoperators and Virtual Environments, 7(1), pp. 1–21. MIT Press (1998)
23. Slyper, R., Hodgins, J. K.: Action Capture with Accelerometers. In: ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 193–199. Eurographics Association (2008)
24. Carnegie Mellon University, Motion Capture Database, from `http://mocap.cs.cmu.edu/`, accessed 10/05/2014
25. Lourakis, M.: Levmar: Levenberg-Marquardt Nonlinear Least Squares Algorithms in C/C++. `http://www.ics.forth.gr/~lourakis/levmar`, accessed 10/05/2014