Bilateral Human Motion Filtering

Nicolas Courty

BILATERAL HUMAN MOTION FILTERING

Nicolas Courty

Valoria/Samsara – European University of Brittany
Campus de Tohannic – 56000 Vannes – France
Nicolas.Courty@univ-ubs.fr

ABSTRACT

Human motions are now frequently used in several applications ranging from computer animation to biomechanics analysis. Acquiring such data can be performed in several ways, but none of them is fully accurate and usually a de-noising process is required as post-treatment. This paper presents a new method to process motion data that tends to preserve some characteristic features of human motions. It is based on an adaptation of the well-known bilateral filter to orientation data. We give an algorithm that computes the filter response, and we show practical results obtained on real motion capture data.

1. INTRODUCTION

Using human motion data has become increasingly popular in several applications such as digital effects production, video games, virtual reality applications, video analysis, medicine or biomechanics. Acquiring such type of data can be done in various settings, ranging from optical devices to inertial or magnetic systems. Optical devices with passive or active markers usually show the best accuracy with high frequencies capture but at the expense of wearing special suits. Markerless motion capture [4, 5], based on vision algorithms, offers an interesting alternative that imposes less constraints, but yet has only been partially solved. Other equipments, such as inertial, mechanics or magnetic systems suffer from calibration problems and can encounter drifting issues over time. New methods such as Prakash [18] are very promising and brings to life the possibilities of *on-set* motion capture systems. Nevertheless, none of these methods provide fully accurate results, and usually a filtering step is required to smooth data and remove noise from the raw signal. Classical de-noising methods are based on local operators that smooth the input signal (such as a Gaussian blur or a Butterworth filter filter which are commonly used by animators) or on subspace techniques such as PCA [20] (and its variations) that seeks to preserve the principal features of the motion.

Though, human motion have specific features that need to be taken into account. For instance, communicative gestures such as non-verbal communication gestures are characterized by rapid and subtle changes that influence greatly the perceived meaning of the gesture [10]. These high frequency information produce subtle details that human beings are able to interpret and decrypt. It is thus of primary interest to be able to preserve those aspects while canceling the inherent noise of the capture system. In this paper, we propose an adaptation of the well-known bilateral filter to rotation data, thus making it suitable to treat human motion. We argue that for the de-noising purpose, the bilateral filter tends to preserve some characteristic features of human motion such as rapid changes in the velocity profile.

The remainder of this paper is organized as follow: we first begin by a short related work on bilateral filtering. Section 3 present the type of data involved in the representation of a human motion and some possible way to handle those data in a filtering framework. Section 4 presents our Bilateral filter for motion data, while section 5 show results

2. RELATED WORK

Bilateral filtering is a well-known technique in signal and image processing. First introduced with its current name by Tomasi *et al.* [21], it has been used in several contexts such as image denoising [2, 14], computational photography [15, 6, 17, 1], stylization [22], optical flow computation [24] or even biomedical imaging [23]. Several theoritical studies have revealed its intrinsic nature and limitations [7, 3]. In the context of image denoising, it has been shown can be related to the classical PDEs such as heat diffusion or the Perona-Malik equation [3]. The reasons for its success are its simplicity to design (it usually implies a local averaging scheme) and to parameterize (only the spatial extent and the contrast preserving strength are required). Bilateral filter has also been extended to other types of data. In [12, 9], it is used as a smoothing operator for $3D$ meshes. In [16], Paris and colleagues have used bilateral filtering to smooth a $2D$ orientation field by incorporating a mapping into the complex plane $\mathbb{C}$. In this sense, their works can be related to our technique. To the best of our knowledge, no existing method uses an adaptation of the bilateral filtering to manifold-valued signal such as $3D$ rotation time series.

3. ORIENTATION DATA FILTERING

In this section we first recall some general facts about the representation of rotations with quaternions and we give our definition of the motion. Secondly, we will review existing methods to filter orientation data.

3.1 Motion representation

A motion $\mathbf{M}$ can be represented as a time series of rotation vectors, each rotation representing a rotation of a particular joint in an articulated figure. Those rotations are now frequently defined as unit quaternions [19, 8], since they have proved to be relatively compact and efficient. Let us recall for clarity some facts about representing rotations with unit quaternions. The quaternion space $\mathbb{H}$ is spanned by a real axis and three imaginary axis $\mathbf{i}$, $\mathbf{j}$ and $\mathbf{k}$ under Hamilton's conventions. A quaternion $\mathbf{q}$ is a 4-uple of real values (w,x,y,z). Unit quaternions ($\|\mathbf{q}\| = 1$) can be used to parameterize rotations in $\mathbb{R}^3$, and can be considered as a point on the unit hyper-sphere S^3. As shown by Euler, any rotation map $\in SO(3)$ can be represented by an angle θ around an

arbitrary axis $\mathbf{v}$. This leads to an intuitive representation of the quaternion as an ordered pair of a real and a vector, *i.e.* $\mathbf{q} = (w, \mathbf{a})$ with $w = \cos\frac{\theta}{2}$ and $\mathbf{a} = \sin\frac{\theta}{2}\mathbf{v}$. The multiplications of two quaternions is defined but not commutative, *i.e.* $\mathbf{q_1 q_2} \neq \mathbf{q_2 q_1}$. The definition of a motion with quaternions for a given kinematic structure is finally:

$$\mathbf{M} = \{\mathbf{q_i}(\mathbf{t}) | i \in [0 \cdots n], \quad t \in [0 \cdots m]\} \tag{1}$$

where n is the number of quaternions used to represent a posture of the skeleton, and m the number of postures in the motion. Hence, $\mathbf{M}$ is a time series of rotation vectors.

3.2 Filtering orientation data

Filtering rotation data is a difficult problem that comes from the non-linearity of the unit quaternion space. Let $\mathbf{X} = \{x_i\}$ be a signal with elements in $\mathbb{R}^n$. The classical convolution operation with a filter mask $(m_{-k}, \ldots, m_k)$ of size $2k+1$ gives the following filter response at the ith element:

$$\mathbf{H}(x_i) = m_{-k}x_{i-k} + \ldots + m_0 x_0 + \ldots + m_k x_{i+k}$$

This operation does not transpose to a signal $\mathbf{Y} = \{\mathbf{q}_i\}$ of elements in $SO(3)$ because the addition is not correctly defined for two unit quaternions since the result is no longer a unit quaternion. A possible solution would be to consider the embedding space $\mathbb{R}^4$, perform computation on quaternions such as vectors of this linear space and then re-normalize the result. Though, this solution can lead to strange behaviors when data are not sufficiently dense enough [13]. Another solution considers a global linearization of the input signal [11], by using for instance the exponential mapping between S^3 and $\mathbb{R}^3$. This method also suffers from problems since there is no such global mapping (e.g. the exponential mapping is ill-defined at the antipode of the identity quaternion), and therefore some singularities may corrupt the result. The concept of local linearization was first used by Fang and colleagues [8]. It consists in decomposing the input signal into a succession of linear displacements between each consecutive samples, filter those displacements, and finally construct the filtered signal by integrating those displacements. As pointed out in [13], this integration yields drifting problems over time. Lee and Shin [13] have proposed a filter design that avoids this problem. The key idea is to consider angular displacement between each samples as a linear displacement in $\mathbb{R}^3$, filter this vector counterparts and construct the signal back through exponentiation, thus avoiding the drifting problem induced by integration in the method of Fang et al. [8]. Moreover, they demonstrated that their construction protocol leads to a linear time-invariant class of filters (LTI filters). Their framework defines the output response of a filter $\mathbf{H}$ as:

$$\mathbf{H}(\mathbf{q}_i) = \mathbf{q}_i Exp(\sum_{r=-k}^{k-1} b_r w_{i+r}) \tag{2}$$

where

$$w_i = Log(\mathbf{q}_i^{-1}\mathbf{q}_{i+1})$$

are the local linearizations of the input signal, and b_r scalars derived from the traditional filter mask coefficients m_j and defined such that:

$$b_r = \begin{cases} \sum_{j=-k}^{r} m_j & if \quad -k \leq r < 0 \\ \sum_{j=r+1}^{k} m_j & if \quad 0 \leq r < k \end{cases} \tag{3}$$

This construction method has been chosen in order to design our bilateral orientation filter.

4. BILATERAL MOTION FILTERING

We first begin this section by recalling the classical form of the traditional bilateral filter. We then present its adaptation to orientation data.

4.1 Bilateral filtering in a linear space

In its original form [21], the bilateral filter is a weighted average of a sample i of the original signal $\mathbf{X} = \{x_i\}$, given by:

$$\mathbf{BF}(x_i) = \frac{\sum_{r=-k}^{k} W(i,r)x_{i+r}}{\sum_{r=-k}^{k} W(i,r)} \tag{4}$$

where $W(i,r)$ are the weights of the filter and are given by a combination of functions of the temporal distance $W_t(i,r)$ and the geometric distance between the samples i and $i+r$: $W_g(i,r)$. W functions are smoothly decaying functions, usually Gaussian functions. In this case, $W(i,r)$ writes:

$$\begin{aligned} W(i,r) &= W_t(i,r) * W_g(i,r) \\ W_t(i,r) &= exp(-\frac{d^2(i,i+r)}{2\sigma_t^2}) = exp(-\frac{|r|}{2\sigma_t^2}) \\ W_g(i,r) &= exp(-\frac{d^2(x_i,x_{i+r})}{2\sigma_g^2}) = exp(-\frac{|x_{i+r}-x_i|^2)}{2\sigma_g^2}) \end{aligned}$$

The idea behind this definition is that both near samples and samples with close-by values will have more influence on the final result. σ_t and σ_g set their relative strength and are generally used to privilege one of these two aspects.

4.2 Bilateral filtering on rotation data

In order to adapt the bilateral filter to orientation data, we first need to choose a metric between rotations. It is common to use the length of the *geodesic* path between two elements on the hypersphere (geodesic distance). This choice is important as it conditions some of the filter properties (see below). This distance is given for two unit quaternions by:

$$d(\mathbf{q}_1, \mathbf{q}_2) = \left\|\log(\mathbf{q}_1^{-1}\mathbf{q}_2)\right\| \tag{5}$$

We now adopt the construction method of Lee and Shin [13] described in the previous section to build our filter. The m_j coefficients used in equation 3 are given by:

$$m_j = W(i,r) = exp(-\frac{|r|}{2\sigma_t^2}) exp(-\frac{\left\|\log(\mathbf{q}_i^{-1}\mathbf{q}_{i+r})\right\|)}{2\sigma_g^2})$$

Those coefficients characterize the Bilateral Orientation filter. They have to be computed with respect to a sliding window over the signal. As the distance between each samples of the signal has to be evaluated several times, it can be convenient to pre-compute all these distances as a band matrix $\mathbf{D}$ as depicted in figure 1. The symmetry of the distance function allows to store only the upper-diagonal part of the matrix[1].

Algorithm 1 gives the final algorithm that computes the bilateral orientation filter (BOF) for a rotation signal $\mathbf{T}$.

[1] In this case, $\mathbf{D}(i,k)$ becomes $\mathbf{D}(k,i)$ if $i > k$.

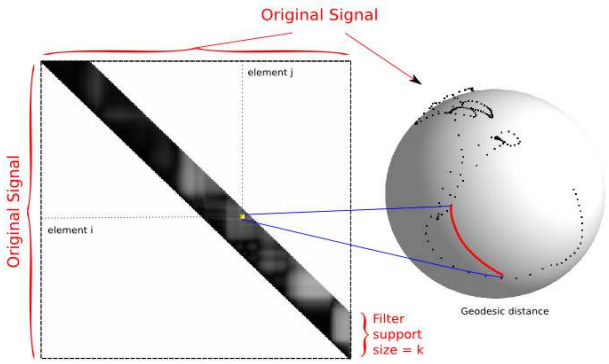

Figure 1: **Precomputation table**. This figure shows a band matrix which contains the distance information used by the filter. Each element correspond to a geodesic distance between samples. Normally, the quaternionic signal belongs to the unit hypersphere in $4D$, but has been represented here, without loss of generality, as the unit sphere in $3D$.

The filter construction presented in [13] guaranties that the filter is linear time-invariant. In our case, this proposition does not hold anymore since the filter coefficients depend on the input signal. Nevertheless, we demonstrate that our bilateral orientation filter keeps interesting properties:

Proposition 1 *The bilateral orientation filter is coordinate-invariant, that is to say that for any* $\mathbf{a},\mathbf{b} \in S^3$, $\mathbf{BOF}(\mathbf{aq_ib}) = \mathbf{aBOF}(\mathbf{q}_i)\mathbf{b}$.

Proof It is sufficient to note that the geodesic metric is also coordinate invariant: let $\mathbf{a}$ and $\mathbf{b} \in \mathbf{S^3}$. We have $\left\|\log(\mathbf{b}^{-1}\mathbf{q}_i^{-1}\mathbf{a}^{-1}\mathbf{a}\mathbf{q}_j\mathbf{b})\right\| = \left\|\log(\mathbf{b}^{-1}\mathbf{q}_i^{-1}\mathbf{q}_j\mathbf{b})\right\| = \left\|\mathbf{b}^{-1}\log(\mathbf{q}_i^{-1}\mathbf{q}_j)\mathbf{b}\right\| = \left\|\log(\mathbf{q}_i^{-1}\mathbf{q}_j)\right\|$, knowing that $\log(\mathbf{aqb}) = \mathbf{a}\log(\mathbf{q})\mathbf{b}$. The rest of the demonstration is the same as in [13].

Proposition 2 *The bilateral orientation filter is time-invariant,*

Proof It is sufficient to note that the geodesic metric is trivially time-invariant. The rest of the demonstration is the same as in [13].

In the case of human motion processing, we simply filter every joint orientation with our filter. As the joints are organized into a hierarchy of joints (the wrist depends on the elbow which depends on the shoulder, etc.), the coordinate invariance property is strongly desirable since the result of the filtering operation will be the same whereas the joints orientations are expressed in local or global coordinate frames.

5. RESULTS

We now present some results obtained with our new filter on real motion capture data. These data represent a complete human body with two hands for a total of 75 joints. The performed motion correspond to a sign language motion. It is depicted in Figure 4.a. The BO filter was first tested on the rotation of the left hand. The original signal is presented in 2.a. We added manually to this signal a quaternionic Gaussian noise with variance $\sigma = 0.18$ radians (Figure 2.d). For

Algorithm 1 Bilateral filtering of a rotation signal $\mathbf{T}$ of a set of rotation $\mathbf{q_i}$ of size $n = 2k+1$

for all $\mathbf{q}_i \in \mathbf{T}$ **do**
  Compute $\mathbf{w}_i = \log(\mathbf{q}_i^{-1}\mathbf{q}_{i+1})$
end for
for $i = 0$ to n **do**
  for $r = -k$ to k **do**
    $m_r = \exp(-\frac{|r|}{2\sigma_t^2})\exp(-\frac{\mathbf{D}(i,i+r)}{2\sigma_g^2})$
  end for
  for $r = -k$ to k **do**
    $m_r = \frac{m_r}{\sum_{i=0}^{n} m_i}$ {Normalization}
  end for
  for $r = -k$ to k **do**
    if $-k \leq r < 0$ **then**
      $b_r = \sum_{j=-k}^{r} a_j$
    else if $0 \leq r < k$ **then**
      $b_r = \sum_{j=r+1}^{k} a_j$
    end if
  end for
  add to $\overline{\mathbf{T}}$: $\mathbf{BOF}(\mathbf{q}_i) = \mathbf{q}_i \exp(\sum_{r=-k}^{k-1} b_r w_{i+r})$
end for
return $\overline{\mathbf{T}}$

comparison purposes, we first filter the noisy signal with a Gaussian filter (Figure 2.b) with variance $\sigma_t = 1.0$, then with the Bilateral Orientation filter (Figure 2.c) with temporal variance $\sigma_t = 1.0$ and spatial variance $\sigma_g = 0.1$. Both filters were applied five times to the noisy signal. Boundary conditions were handled by mirroring the signal at both extremities. It is interesting to notice how the overall shape of the signal and the principal features have been recovered through the filtering process. Figure 2.e and .f shows the angular velocity of the original signal compared to the final signal. It is computed as $||\log(\mathbf{q}_i^{-1}\mathbf{q}_{i+1})||\ rad.s^{-1}$. While Gaussian blur exhibits less peaks in the signal and a globally less important magnitude, the BO filter preserves the overall aspect of the velocity profile, at the expense of amplifying in some cases the speed magnitude.

Figure 3 shows the evolution of the Mean Square Error (MSE) computed between the original signal and the filtered signal for different values of the σ_g parameter. σ_t was set in this experiment to 1.0. For orientation data, we have chosen the following measure between two signals $\mathbf{q1}$ and $\mathbf{q2}$ (of same length $N+1$):

$$MSE = \frac{1}{N}\sum_{i=0}^{N} ||\log(\mathbf{q1}_i^{-1}\mathbf{q2}_i))||^2$$

One can observe that the choice of the spatial variance impacts on the quality of the de-noising process. Indeed, this figure suggests that applying an average of 5 times the filter yields generally good results.

We then processed the entire motion (75 rotation time series corresponding to every joints). The length of the motion was about 150 frames. Our implementation yields a computation time on a standard laptop of 300 ms. Figure 4 illustrates the impact of filtering the rotational components of the motion on the resulting trajectories of the end effectors (in our case, the hands) expressed in the cartesian space. Figure 4.a gives a short outline of the test motion. The test hand

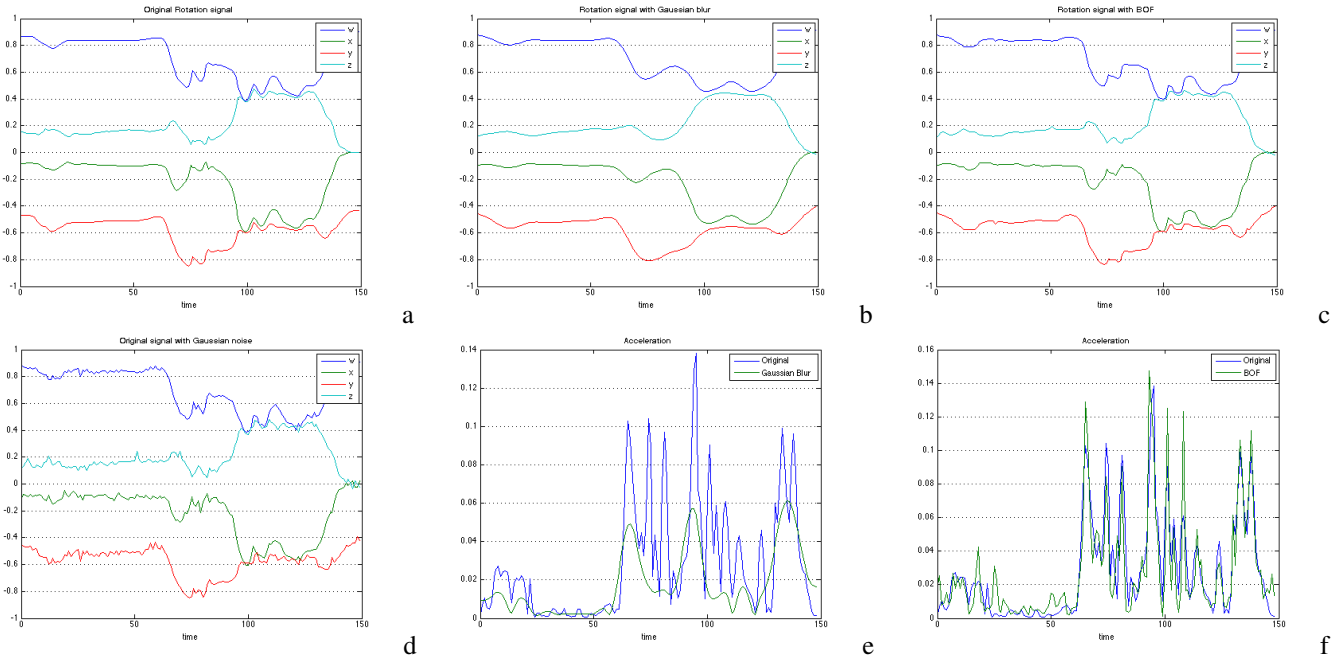

Figure 2: **Filtering real data**. (a) Original signal (d) plus Gaussian noise (b) filtered with a Gaussian Kernel $\sigma_t = 1.0$ (c) filtered with BO filter $\sigma_t = 1.0$, $\sigma_g = 0.1$ (e,f) Comparisons between original and filtered signals angular velocities.

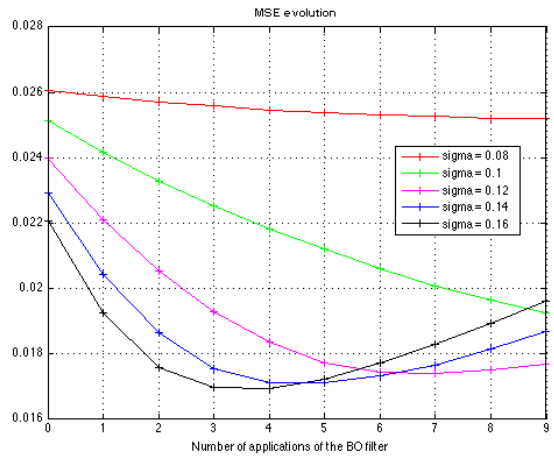

Figure 3: **MSE Evolution**. This diagram shows the evolution of Mean Square Error (MSE) between the original and the filtered signal along several application of the Bilateral Orientation filter for different values of σ_g.

trajectory has been represented in red. This trajectory cumulates in some sense all the errors on the previous articulations along the kinematic chain (*i.e.* elbow, shoulder, etc.). This propagation effect magnifies at the same time the effect of the filtering process. For example, processing the whole motion with Gaussian filtering leads a global diminution of the motion's energy, thus leading to a smoother trajectory but with less amplitude (Figure 4.c). In this last case, some details of the hand motion are lost (bottom left of the trajectory). Those details are in fact small and quick repetitions that are used in the case of sign language to outline a particular idea. We can see that in the case of Bilateral Orientation filter (Figure 4.d), this pattern is conserved. Moreover, the global amplitude compares better to the original signal.

6. CONCLUSION AND FUTURE WORKS

In this paper we have presented a new and original filter for human motion data. Based on bilateral filtering, it processes orientation data without any global linearization process. The filter is designed to preserve some characteristic features of human motion such as rapid change in accelerations, and exhibits interesting properties such as coordinate and time invariances. The filter was tested on real motion capture data, and preliminary results are very promising. We plan to complete this work by studying more in depth the properties of our filter and compare with other state-of-the-art de-noising methods such as PCA or Kernel-PCA. Other applications such as motion stylization and caricature will also be considered.

Acknowledgement The author thanks S. Gibet and P.-F. Marteau for useful discussions on the subject. The motions used in this paper as examples have been acquired through the HuGex Robea project funded by CNRS, France.

REFERENCES

[1] S. Bae, S. Paris, and F. Durand. Two-scale tone management for photographic look. *ACM Trans. Graph*, 25(3):637–645, 2006.

[2] E. Bennett and L. McMillan. Video enhancement using per-pixel virtual exposures. *ACM Trans. Graph*, 24(3):845–852, 2005.

[3] A. Buades, B. Coll, and J.-M. Morel. Neighborhood filters and PDE's. *Numerische Mathematik*, 105(1):1–34, 2006.

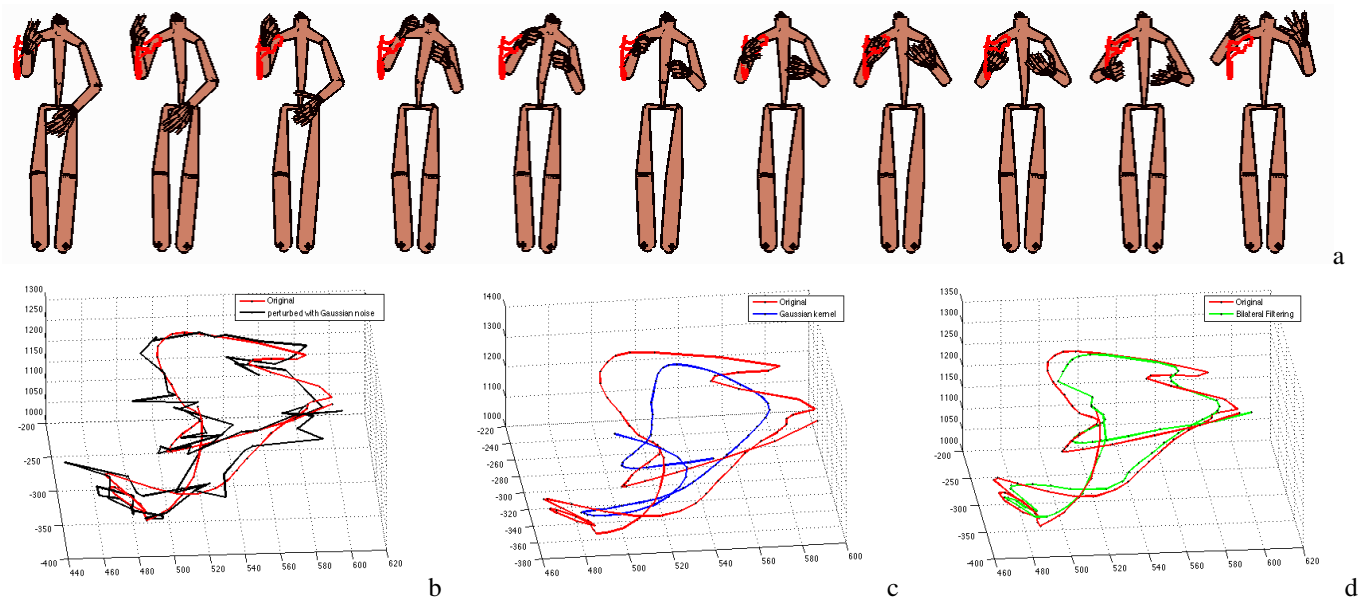

Figure 4: **Motion strip of the test sequence**. (a) Ten poses along the test motion. The red curve shows the right hand trajectory which is used in the following illustrations. (b,c,d) Comparisons between original signal and reconstruction of the right hand trajectory (in cartesian coordinates) : (b) effects of gaussian noise (c) Smoothing using a Gaussian Kernel (d) Smoothing using bilateral filtering.

[4] G.. Cheung, S. Baker, and T. Kanade. Visual hull alignment and refinement across time: A 3D reconstruction algorithm combining shape-from-silhouette with stereo. In *CVPR*, pages 375–382, 2003.

[5] J. Deutscher and I. Reid. Articulated body motion capture by stochastic search. *Int. Journal of Computer Vision*, 61(2):185–205, 2005.

[6] F. Durand and J. Dorsey. Fast bilateral filtering for the display of high-dynamic-range images. *ACM Trans. Graph*, 21(3):257–266, 2002.

[7] M. Elad. On the origin of the bilateral filter and ways to improve it. *IEEE Transactions on Image Processing*, 11(10):1141–1151, 2002.

[8] Y. C. Fang, C. C. Hsieh, M. J. Kim, J. J. Chang, and T. C. Woo. Real time motion fairing with unit quaternions. *Computer-Aided Design*, 30(3):191–198, 1998.

[9] S. Fleishman, I. Drori, and D. Cohen-Or. Bilateral mesh denoising. *ACM Trans. Graph*, 22(3):950–953, July 2003.

[10] A. Héloir, N. Courty, S. Gibet, and F. Multon. Temporal alignment of communicative gesture sequences. *Computer Animation and Virtual Worlds*, 17:347–357, July 2006.

[11] J. Johnstone and J. Williams. Rational control of orientation for animation. In *Graphics Interface '95*, pages 179–186, May 1995.

[12] T. Jones, F. Durand, and M. Desbrun. Non-iterative, feature-preserving mesh smoothing. *ACM Trans. Graph*, 22(3), July 2003.

[13] J. Lee and S. Y. Shin. General construction of time-domain filters for orientation data. *IEEE Trans. on Visualization and Computer Graphics*, 8(2):119–128, 2002.

[14] C. Liu, W. Freeman, R. Szeliski, and S. Bing Kang. Noise estimation from a single image. In *CVPR*, pages 901–908. IEEE Computer Society, 2006.

[15] B. Oh, M. Chen, J. Dorsey, and F. Durand. Image-based modeling and photo editing. In *Proc. of Siggraph*, pages 433–442, 2001.

[16] S. Paris, H. Briceño, and F.-X. Sillion. Capture of hair geometry from multiple images. *ACM Trans. Graph*, 23(3):712–719, 2004.

[17] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama. Digital photography with flash and no-flash image pairs. *ACM Trans. Graph*, 23(3):664–672, 2004.

[18] R. Raskar, H. Nii, B. DeDecker, Y. Hashimoto, J. Summet, D. Moore, Y. Zhao, J. Westhues, P. Dietz, M. Inami, S. Nayar, J. Barnwell, M. Noland, P. Bekaert, V. Branzoi, and E. Burns. Prakash: Lighting-aware motion capture using photosensing markers and multiplexed illumination. *ACM Trans. Graph.*, 26(3), August 2007. session : Performance Capture.

[19] K. Shoemake. Animating rotation with quaternion curves. In *Proc. of Siggraph*, pages 245–254, 1985.

[20] T. Tangkuampien and D. Suter. Human motion de-noising via greedy kernel principal component analysis filtering. In *ICPR*, pages 457–460, 2006.

[21] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *ICCV*, pages 839–846, 1998.

[22] H. Winnemöller, S. Olsen, and B. Gooch. Real-time video abstraction. *ACM Trans. Graph*, 25(3):1221–1226, 2006.

[23] W. Wong, A. Chung, and S. Yu. Trilateral filtering for biomedical images. In *ISBI*, pages 820–823. IEEE, 2004.

[24] J. Xiao, H. Cheng, H. Sawhney, C. Rao, and M. Isnardi. Bilateral filtering-based optical flow estimation with occlusion detection. In *ECCV*, volume 3951 of *Lecture Notes in Computer Science*, pages 211–224, 2006.