Denoising method of motion capture data based on neural network

Yongqiong Zhu
School of Art, Wuhan Business University, Wuhan, China
yongqiongzhu@gmail.com

Abstract. Optical motion capture systems are prone to data loss or errors during the capture process, which can cause noise in the data. In order to suppress the influence of noise, it is necessary to denoise the captured data to improve the quality of the data and make the denoised motion data reflect the actual motion of the actor as accurately and completely as possible. Aiming at how to effectively denoising motion capture data, researchers have proposed many effective approaches. These approaches are mainly divided into the following three categories: based on filtering strategies, algorithms based on matrix low-rank theory, and data-driven algorithms. Among them, the data-driven denoising method has gradually become the mainstream technology. At present, many studies in this area have achieved certain results, but there is still space for further improvement. Based on the data-driven method, this paper proposes a neural network-based machine learning algorithm to implement denoising processing for motion capture data. Experiments show that this algorithm can significantly recover noisy data.

1. Introduction
Due to the influences of optical on the sensor and the occlusion of marker points caused by human motion during the capture process, the data of some marker points is lost or wrong, which causes noise in mocap data. To suppress the effects of noise, the captured data needs to be denoised to improve the quality. At the same time, in the process of denoising, it needs to protect the details of the motion capture data to the greatest extent to obtain the optimal approximation of the original real data, so that the denoised motion data reflects the actual actor as accurately and completely as possible motion.

The structure of this paper is as follows. In the second part, the related work of the research and the objectives of this research are introduced. The third part introduces the system model of this paper, and gives the way to achieve it. Finally, our algorithm is given. The fourth part is experimental simulation to prove our method. The fifth part is the conclusion.

2. Related work
Aiming at how to effectively denoise motion capture data, researchers have proposed many effective algorithms. These algorithms are mainly divided into the following three categories: denoising methods based on filtering strategies, algorithms based on matrix low-rank theory, and data-driven algorithms.

(1) Denoising method based on filtering strategy
There are three main types: the first type is based on linear non-time-varying filter technology, and its typical representatives include methods using low-pass filters and filtering masks (FM). Low-pass...
filter denoising technology is currently used in commercial motion capture systems, such as the Vicon system. The filtering mask was first introduced into motion data denoising by scholars such as J. Lee of CMU [1]. They proposed a linear non-time-varying filtering framework. The main idea of the framework is to first transform the original orientation data into a similar vector space, then use the filtering mask technology to perform vector smooth conversion, and finally transfer the converted result back to the azimuth space, thereby reducing the sharpness of the original data to achieve the purpose of removing noise interference. The second type is based on Kalman filter technology, whose main idea is to sequentially filter motion data containing noise [2]. The third type is the denoising technology based on space-time filter. In view of the temporal continuity of motion capture data and the constraints of the spatial topology between individual joint points, H. Lou [3] and other scholars have constructed a series of spatiotemporal filters to denoise human motion data through sample learning.

Signal processing algorithms based on filtering [4] are very fast and effective when processing small-scale noise data, but none of these algorithms explicitly use the human body structure information and spatial and temporal domain information implicit in human motion capture data. Therefore, when the above algorithm is used for noise recovery, this information cannot be effectively maintained and cannot be used to deal with large-scale data distortion problems.

(2) Method based on matrix low rank filling

Some researchers have introduced the idea of low-rank matrix completion into the reconstruction of missing motion data [5]. The advantage is that no training data set is needed to achieve the purpose of missing data recovery. For example, Lai [6] explored the low rank structural characteristics of the motion data matrix and used the singular value threshold (SVT) method to complete the missing motion data. Without the need for any training data, this method is satisfactory for the reconstruction of missing motion sequences with the same pose, but for long motion sequences with different semantics, the recovery effect is not significant. In addition, Tan and other scholars noticed that the motion trajectory segmented data matrix has a better low rank, and proposed to reorganize the motion sequence into a combination of different trajectory segments, and then perform local matrix complement recovery to achieve a better reconstruction effect.

The biggest advantage of this type of algorithm is that it does not require any training samples and solves the problem of missing samples. However, if there are large-scale missing points in the motion sequence, the recovery effect is not good.

(3) Data-driven approach

In recent years, thanks to the development of new motion capture devices and the improvement of capture technology, mocap data has shown explosive growth, provided enough samples for data-driven algorithms, and promoted the improvement of the processing capabilities of these algorithms. So that it can fully consider the complexity and diversity of human movement. For example, Xiao [7] and other scholars found that in different types of movements, the movements of certain parts of the human body are similar, so they proposed to divide each pose of the human body into 5 parts, and use the training samples to train the dictionary of these 5 parts respectively, and then perform sparse representation on the motion sequence containing noise to remove noise. Other machine learning models are often used in the training process.

At present, due to the adequacy of mocap data, data-driven denoising methods have gradually become the mainstream technology for denoising motion capture data [8]. At present, many studies in this area have achieved certain results, but there is still space for further improvement. Based on the data-driven method, this paper proposes a neural network-based algorithm to implement denoising for motion capture data.
3. Design

3.1. System model
This paper designs a data-driven BP neural network model to learn motion capture data so that noise can be found. The neural network model is three layers, the input layer has two neurons, the hidden layer is set to three neurons, and the output layer is one neuron.

Settings for the input layer:
Input unit 1: Assume that the captured character has n marked point, and each marked point has three coordinates of x, y, and z in the three-dimensional coordinate space, so for a frame t containing n marked points, it can be expressed as a matrix as:

$$f(t) = C^t_{n \times 3}$$

This matrix records the spatial coordinates of each marker.
Input unit 2: Records the presence or absence of the coordinates of each marked point. Markers without coordinates have a value of 0, otherwise they are 1

$$x(t) = M^t_{n \times 3}$$

The value of M is either 0 or 1.
The value of the bias term $b_1 = 0.5$.
Hidden layer:
Set the weights of the input and hidden layer and hidden and output layer units to $w_{ij}$, i is the previous layer unit node, and j is the current layer unit. Learning rate $\theta = 0.9$. The value of the bias term $b_2 = 0.5$.
Output layer:
Since it is a binary classification problem, set the output label $: t = 1$
The sigmoid of the neural network is set to

$$s(z) = \frac{1}{1 + e^{-z}}$$

The designed neural network model (Fig. 1) is as follows:

![Neural network model](attachment:image.png)
3.2. Strategy

The learning process of a neural network consists of two steps:

1. Forward propagation gets the output layer error $e$
   
   The input layer input sample values to each hidden layer, and the output results of the hidden layer are input to the output layer.
   
   For example, the input for node 3 is:
   
   $$z_3 = f(t)w_{13} + x(t)w_{23} + b_1$$  \hspace{1cm} (4)
   
   The output of node 3 is:
   
   $$s(z_3) = \frac{1}{1+e^{-z_3}}$$  \hspace{1cm} (5)
   
   Calculation of other nodes and so on.
   
   The final error is:
   
   $$e = s(z_6) - t$$  \hspace{1cm} (6)

2. Backpropagation Pass
   
   Calculate the error based on the results of the output layer. Inversely correct the weight of each layer unit according to the error until the error is reduced to an acceptable level. The method used is the gradient descent method. When the error signal for each neuron is computed, the weights coefficients of each neuron input node may be modified.
   
   $$new \ w_{ij} = old \ w_{ij} - \theta \cdot \frac{\partial e}{\partial w_{ij}}$$  \hspace{1cm} (7)

4. Simulation

The simulation experiment uses the human skeleton model and human motion data capture library provided by Carnegie Mellon University (CMU). The database can be obtained from http://mocap.cs.cmu.edu. The human skeleton model in the database includes a total of 31 joint points, and the data formats are BVH, ASF/AMC and C3D. The IDE used in this experiment is PyCharm, the language is Python, and the configured data packets mainly include numpy, matplotlib, etc., and the data format used is ASF/AMC.

Each joint point is organized in a tree structure, where the hips are root nodes, and joint represents the set of marker points. It can be described mathematically as $|\text{joint}| = 31$. Since each joint has corresponding coordinates in the actual physical three-dimensional space, the 3D space attribute dimension of each frame of data is: $n = |\text{joint}| \times 3 = 93$. Noise often only exists in some joints in some frame sequences. We adopt the following scheme to add noise: Given motion capture data of $m \times n$ ($m$ is the number of frames, $n$ dimension is 93), randomly generated Gaussian noise $Gp \times n$ ($\varepsilon, \sigma$) with $\varepsilon = 1$ mean and $\sigma = 1.2$ variance ($p$ is noise-containing Frames).

This article uses 517 frames of data from 07_05-walk of CMU and the first 1200 frames of data from 14_46 which motion semantics are walking, sitting, and testing. We randomly selecting 400 frames as noise interference frames, and dividing the 1200 frames into 6 segments with 200 frames in each segment. Look at the recovery results of the frame sequence after sampling the denoising method used in this article. We compare the experimental results with the singular value thresholding, SVT method in.
5. Conclusion
Compared with other methods, the biggest advantage of data-driven algorithms is that they can automatically discover and learn the potential time and space domain information of motion sequences. However, these algorithms may suffer from missing samples. For example, they cannot handle motion types that do not yet exist in the database, and they cannot obtain samples of the same motion category for long sequences with multiple motion semantics. Therefore, the existing data-driven denoising methods also have space for improvement. The machine learning algorithm based on neural network proposed in this paper to achieve denoising of motion capture data has achieved good results.
Acknowledgements
This paper is supported by grant from the academic team of Wuhan Business University (2020TD001).

References
[1] Lee J, Shin S Y. General construction of time-domain filters for orientation data [J]. IEEE Transactions on Visualization and Computer Graphics, 2002, 8 (2): 119-128
[2] Shin H J, Lee J, Shin S Y, et al. Computer puppetry: an importance based approach [J]. ACM Transactions on Graphics, 2001, 20 (2): 67-94
[3] Lou H, Chai J, Example-based human motion denoising [J]. IEEE Transactions on Visualization and Computer Graphics, 2010, 16 (5): 870-879
[4] Anandakrishnan N. Image Denoising & Restitution Using Fuzzy Technique [J]. 2018.
[5] Fan J, Yang C, Udell M. Robust Non-Linear Matrix Factorization for Dictionary Learning, Denoising, and Clustering [J]. arXiv preprint arXiv: 2005.01317, 2020.
[6] Lai R Y Q, Yuen P C, Lee K K W. Motion capture data completion and denoising by singular value thresholding [C] //Proceedings of Eurographics. Aire-la-Ville: Eurographics Association Press, 2011: 45-48
[7] XIAO J., FENG Y., HU W.: Predicting missing markers in human motion capture using l1-sparse representation. Computer Animation and Virtual Worlds 22, 2-3 (2011), 221–228. 1, 2
[8] Y. Yang, C. Hou, Y. Lang and C. Li, "Motion Classification Based on Noisy Micro-Doppler Signatures." 2019 International Radar Conference (RADAR), TOULON, France, 2019, pp. 1-4, doi: 10.1109/RADAR41533.2019.171374.