Motion Capture Data Completion and Denoising by Singular Value Thresholding

Ranch Y.Q. Lai [1] and Pong C Yuen [1] and Kelvin K.W. Lee[2]

[1]Department of Computer Science, Hong Kong Baptist University
[2]Department of Communication studies, Hong Kong Baptist University

Abstract

Human motion is of high articulation and correlation.When a human motion sequence is represented by a matrix, the matrix will be approximately low-rank. This low-rank property has been used by previous manifold-based approaches such as PCA and GPLVM. Encouraging results yielded by those approaches show that the low-rank property is of interest and importance for animating human motion. However, none of those approaches explicitly exploits it for motion capture data processing. In this paper, we propose to deal with motion capture data based on recently developed low-rank matrix completion theory and algorithms. Unlike previous approaches, the proposed method relies on low-rank prior instead of motion prior. To verify it effectiveness for dealing with motion capture data, we show that incomplete human motion can be effectively reconstructed. We also demonstrate that a noise-corrupted motion can be nicely recovered.

Categories and Subject Descriptors (according to ACM CCS): I.7 [Computer Graphics]: Computer Graphics—Animation

1. Introduction

A well-known truth about human motion is that it is articulated: the entire motion is described by the movement of joints which are connected by inflexible bones. The movement of one joint is correlated with that of another. If we record the trajectories of all joints and represent them as a motion matrix, this correlation is completely contained in the matrix. The problem is how to extract and model the correlation from the matrix. When the correlation is linear, the best property to be used will be the rank, which by definition measures the linear dependence of the rows (or columns) of a matrix. The higher the correlation is, the lower the rank will be. In this paper, we apply a low-rank matrix completion algorithm referred to as Singular Value Threshold(SVT) [CCS10] to motion capture data processing. By presenting two experiments on dealing with human motion(namely, completion and denoising), we demonstrate that the low-rank property of motion sequences can be exploited as a prior effectively in an explicit way different from previous approaches.

Related work Previous work concerning linear correlation of human motion is mostly based on linear dimension reduction [SHP04, LWS02, CH07, etc]. Non-linear correlation has also been considered and modeled by latent variable models [WFH08, GMHP04] and other non-linear dimension reduction methods [LE06]. Given some keyframing information and motion capture data, Pullen and Bregler [PB02] proposed to compete(fill in) the missing degrees of freedom based on correlation analysis of joint angles . Lou and Chai [LC10] proposed a method for motion capture data denoising and missing value completion. They first construct a set of bases from *clean* motion capture data, and then given a noisy motion signal, the denoised motion is estimated by trading off between reconstruction error from the bases and the observation likelihood. Chai and Hodgins [CH05] proposed an online local linear model for incompletion motion reconstruction. Performance of the proposed method is compared with and claimed superior to global PCA, GPLVM and LLM(see reference therein). Their approach relies on a large data set for constructing motion prior.Notice that most data-driven approaches including the aforementioned ones use motion capture data as a prior(motion prior), see also [LC10, CH07] for examples. Other than what we discussed above, previous work related to motion capture data processing includes compression [TWC*09, Ari06], segmentation and classification [LFAJ10, LE06], etc. On the other side, research on low-rank matrix completion theory and algorithms has become booming since a recent introduction of (nonsmooth but convex) nuclear norm approximation to solve the originally NP-hard problem [Faz02]. It can be shown that under some conditions, incomplete low-rank matrix can be recovered exactly using such approximation [CR09]. Many algorithms have been and are being developed, trying to solve the matrix completion problem as fast and as accurate as possible [MGC09, CCS10, etc]. Applications related to low-rank matrix completion can be found in realm of computer vision and image processing, see [JLSX10] for example. To our knowledge, however, there is no previous work on application of low-rank completion theory to human motion modeling.

Our work is similar to [LC10] and [CH05]. However, the approach taken is quit different since we use rank prior instead of motion prior. In others words, we do not need any motion capture data as training sets. Experimental results show the effectiveness of the low-rank prior for human motion completion and denoising.

The rest of this paper is organized as follows. In section 2, we first briefly introduce the SVT algorithm and discuss the low-rank property of motion capture data as well as how to make use of it. We then provide two preliminary applications of this property in section 3 followed by the conclusion and discussion of future work in section 4.

2. Overview of the proposed method

Low rank matrix completion considers the optimization problem $P_0 : \min_{\mathbf{X}}\{\ \text{Rank}(\mathbf{X}) : \mathbf{X} \in \mathcal{C}\}$, where $\mathcal{C}$ is some known constraint set usually described by linear equalities and convex inequalities. SVT is one of the many algorithms designed to solve such a problem. It is based on the fact that truncating the singular values of $\mathbf{Y}$ by threshold λ is the solution to the problem $P_1 : \min_{\mathbf{X}}\{\frac{1}{2}||\mathbf{X}-\mathbf{Y}||_F^2 + \lambda||\mathbf{X}||_*\}$, where $||\mathbf{X}||_*$ is the nuclear norm for approximating the rank of $\mathbf{X}$. Instead of solving P_0 directly, SVT solves P_1 as an approximation. To handle the constraint $\mathbf{X} \in \mathcal{C}$, it iteratively projects $\mathbf{X}$ between objective and the set $\mathcal{C}$. When applied to motion capture data, we can see SVT as consisting of two terms, namely likelihood and prior. The likelihood term is measured by the Frobenius norm, and the prior is just the low-rank property. Since the prior plays an important role, we would like to make sure the correctness of this prior for all kinds of motion sequences, despite the fact that many previous approaches have adopted this assumption.

We use $\mathbf{Y} = [\mathbf{y}_1, ..., \mathbf{y}_n]$ to represent a motion sequence, where each $\mathbf{y}_i \in \mathbf{R}^m$ represents a frame. We assume that $n > m$. To check that whether a motion $\mathbf{Y}$ has a low-rank structure, we can find the spectrum of $\mathbf{Y}^T\mathbf{Y}$ and observe how fast it decays. To do so, we collect 112 motion sequences, each from one of the 112 subjects in CMU mocap database.

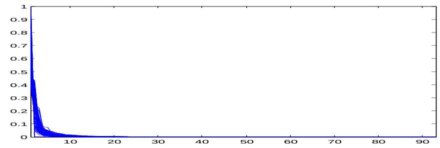

Figure 1: *Normalized spectrum of various motion sequences*

The subjects cover various motion styles including walking, jumping, running, dancing, etc. For each motion, its eigenvalues (denoted by $\mathbf{s} \in \mathbf{R}^m$) are calculated and normalized such that $||\mathbf{s}||_1 = 1$. The normalized spectrum is showed in figure 1, from which we can see that the spectrum concentrates in the bottom-left area and decays very rapidly, implying the correctness of the low-rank prior.

3. Applications

We now adopt low-rank matrix completion theory to exploit this low-rank prior explicitly. To our knowledge, this adoption is the first time in computer animation domain. Different from previous approaches which are based on linear or nonlinear dimension reduction, our method directly works on the original (high dimensional) space and does not require extra training data.

Our experiments are based on CMU mocap database, in which human skeleton model is described by the acclaim asf file. The skeleton consists of 31 joints including the root joint. The degrees of freedom (DOF) for each joint vary from 1 to 3. There are totally 62 DOF's for a skeleton. The motion data are stored in the acclaim amc file, which contains the recorded data for the DOF's.We use Matlab and a mocap toolbox [Law] for decoding asf/amc files and for displaying motion. Each motion sequence is represented by an $m \times n$ matrix, where $m = 31 \times 3 = 93$. Although other motion sequences also work well, for consistence, the motion used for the following experiments is trial 1, subject 7 which is a walking sequence.

Reconstructing missing joints We first consider the situation when some entries of the motion $\mathbf{Y}$ are missing. Given only such an incomplete motion, and with no physical information(e.g. joint limit, bone length) or training set available, how can we recover the unknown entries? We solve this problem by using SVT. Following the notional convention in [CCS10], a projection operator P_Ω is defined such that it extracts the entries of a matrix indexed by Ω and reshapes them as a vector. Now the reconstructed motion is given by the solution to the following optimization problem with variable $\mathbf{X}$:

$$\begin{array}{ll} \text{minimize} & \text{Rank}(\mathbf{X}) \\ \text{subject to} & P_\Omega(\mathbf{X}) = P_\Omega(\mathbf{Y}) \end{array} \quad (1)$$

Figure 2 shows a recovery of 50% randomly selected missing joints(marked in red) from an incomplete motion(blue) with mean square error of 0.0253. The recovered motion(green) is of rank 16. Note that the motion is shown at every 50 frames

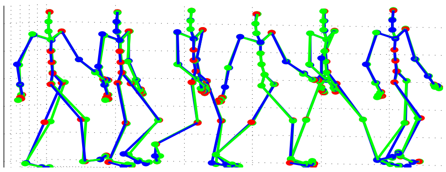

Figure 2: *Reconstructing an incomplete motion*

We can see that the even though up to 50 percent of motion data are missing, we can still recover the motion very effectively, and the resulting motion is of low-rank. We expect that potential applications related to this preliminary example can range from mocap data processing, motion editing to interactive motion synthesis. For instance, in one application, the animator sketches the positions of some joints, the computer then automatically competes the rest joints(similar to [CH05]). This will greatly help reduce tedious and burdensome labor work. In another application, a novice animator modifies (nearly randomly) an originally smooth motion. Fortunately, the computer discovers that the low-rank structure is destroyed. Then it automatically modifies the motion again to trade off between the low-rank constraint (so that the motion is not invalid) and the intention of the novice animator. In this case, the assumption is that for a motion to be good(in the sense of physically valid and well-coordinated), it must first satisfy the low-rank constraint.

Motion capture data denoising Here we provide another example showing how to exploit the low-rank property for recovering human motion corrupted by noise. Different from the denoising given in [LC10], our method does not require motion prior. Given a single noise-corrupted motion $\tilde{\mathbf{Y}}$, a straightforward denoising approach will be to use SVD directly. If we assume that all entries are unknown and contaminated by noise, and $\mathbf{Y}$ has a fast decaying spectrum, the best estimate of $\mathbf{Y}$ is $\mathbf{U}_{1:r}\Sigma_{1:r}\mathbf{V}_{1:r}^T$, where $\mathbf{U},\mathbf{V}$ and $\mathbf{S}$ are singular vectors and singular values (all in matrix form) of $\tilde{\mathbf{Y}}$ respectively. In fact, it is the solution of the optimization problem $P_2 : \min_{\mathbf{X}}\{||\mathbf{X}-\mathbf{Y}||_F^2, \text{Rank}(\mathbf{X}) \le r\}$, where $||\cdot||_F$ represents Frobenius norm. Another way to look at this problem is to reformulate it equivalently as $P_3 : \min_{\mathbf{X}}\{||\mathbf{X}-\mathbf{Y}||_F^2 + \lambda\text{Rank}(\mathbf{X})\}$ for some λ depending on r. When the noise is large, r should be set to a small value to remove more (corrupted) singular values. This is of course assuming that the noise is not so large that all singular values are corrupted. On the other hand, λ should be set to a large number when r is small according to duality theory. In this case, we trust the low-rank prior more that we trust what we have observed. Since the problems P_2 and P_3 are equivalent and the latter is in fact similar to the idea behind SVT(i.e., P_1), one may argue that denoising by truncated SVD is better than SVT due to complexity issue of the latter. We remove this doubt from the perpestive that SVD can not handle partially corrupted motion capture data while SVT is designed to do so. This is verified by the following experiment.

We first generate a zero-mean Gaussian noise $\mathbf{G}$ with standard deviation denoted by σ. Given a motion $\mathbf{Y}$, we then obtain the corrupted motion by $\tilde{\mathbf{Y}} = \mathbf{Y}+\mathbf{G}$. Based on $\tilde{\mathbf{Y}}$ and a rough guess of standard deviation denoted by $\hat{\sigma}$, we estimate the original motion by solving the following optimization problem with variable $\mathbf{X}$:

$$\begin{aligned} \text{minimize} \quad & \text{Rank}(\mathbf{X}) \\ \text{subject to} \quad & \mathbf{A}P_\Omega(\mathbf{X}) \le \mathbf{b} \qquad (2) \\ & P_{\Omega^c}(\mathbf{X}) = P_{\Omega^c}(\mathbf{Y}) \qquad (3) \end{aligned}$$

where $A \in \mathbf{R}^{k\times mn}$, $\mathbf{b} \in \mathbf{R}^K$ are problem data, Ω^c is complement of Ω. Constraint (2) provides a confidence interval for each entry of $\mathbf{X}$ indicated by the set Ω. Constraint (3) is the ordinary vector equality that restricts X_{ij} to be Y_{ij} for $(i,j) \in \Omega^c$. For our experiment, the constraint (2) is implemented as $-\hat{\sigma}+Y_{ij} < X_{ij} < Y_{ij}+\hat{\sigma}$ for $(i,j) \in \Omega$. we set $\sigma = 3$, $\hat{\sigma} = 2\sigma$, and Ω contains randomly generated indexes covering 30% entries(i.e., 70% corrupted). This problem is solved by SVT and and the result is shown at every 50 frames in figure 3. From the figure we can see that compared to the contaminated motion(top), the denoised motion(bottom, of rank 34) is visually much better, proving that proposed method is effective. Result obtained by SVD truncated at rank 3, which is the best among all truncations, is also plotted(middle) for comparison. The mean squared errors for SVT and truncated SVD are 0.0804 and 0.4499 respectively. If we remove the constraints (2) and (3)(e.g., set $\hat{\sigma}$ to a very large value), and set the parameter λ in SVT to a proper value, then SVT will converge to the solution given by truncated SVD. In this case, truncated SVD is better only in the sense of lower complexity. However, with the constraints in

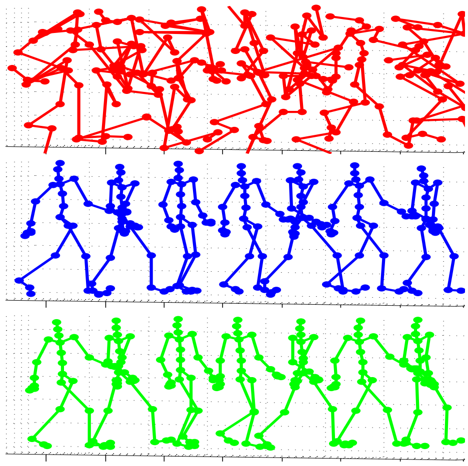

Figure 3: *Recovering a noise-corrupted motion*

place, SVT outperforms truncated SVD. For computer aided animation, we expect the confident interval (3) to provide useful information such as that physical constraints an user constraints provided by animators.

4. Conclusion

This paper presents our recent work on dealing with human motion capture data. We provided two preliminary examples which prove the effectiveness of exploiting the low-rank property. Although the motion sequence used is the walking sequence, similar results are expected if experiments are conducted on other sequences. This is supported by figure 1, which shows that human motion sequences are low-rank. Notice that the completion and denoising processes do not rely on any motion prior. Also notice that there is no physical information or auxiliary training sets involved in the two examples. We believe that if we can incorporate physical models or training sets into the optimization problem, much better results in the two experiments can be obtained. For instance, if we know the bone length, we can constrain the position of a missing joint to obtain a better estimate. This can be done by, for instance, forming an convex inequality constraint extracted from physical model to replace the constraint in (2). We leave this extension to future work.

One limitation of using the low-rank prior might be the complexity issue. For the motion completion experiment, it takes about 10 seconds to estimate 50 percent of missing entries for the motion of length 316 on a modern PC. One reason for that is, although what we have shown is a preliminary problem, it is still very challenging. However, we believe this limitation can be tackled either by the fast advancing optimization technique, or by other adjustment and assistance in real applications.

Another limitation is inherited from low-rank matrix completion theory, which states that when a whole row or column is missing, it is impossible to recover the incomplete matrix. This is not difficult to understand: to recover such a low rank matrix, one solution will be to fill the missing row(or column) with all zeros. Without any additional information(e.g. axillary data, physical constraints or training set), it is very difficult not to arrive at this trivial and useless solution. Even with such information in hand, how to make use of it is another problem. Our future research will mainly focus on solving this problem by using additional information and applying it to more sophisticated applications in computer animation .

Acknowledgment

This project is supported by the Faculty Research Grant of Hong Kong Baptist University.

References

[Ari06] ARIKAN O.: Compression of motion capture databases. *ACM Transactions on Graphics (TOG) 25*, 3 (2006), 890–897. 1

[CCS10] CAI J., CANDÈS E., SHEN Z.: A Singular Value Thresholding Algorithm for Matrix Completion. *Siam Journal of Optimization 20*, 4 (2010), 1956–1982. 1, 2

[CH05] CHAI J., HODGINS J.: Performance animation from low-dimensional control signals. In *ACM SIGGRAPH 2005 Papers* (2005), ACM, pp. 686–696. 1, 2, 3

[CH07] CHAI J., HODGINS J.: Constraint-based motion optimization using a statistical dynamic model. In *ACM SIGGRAPH 2007 papers* (2007), ACM, p. 8. 1

[CR09] CANDÈS E., RECHT B.: Exact matrix completion via convex optimization. *Foundations of Computational Mathematics 9*, 6 (2009), 717–772. 2

[Faz02] FAZEL M.: *Matrix rank minimization with applications*. PhD thesis, PhD thesis, Stanford University, 2002, 2002. 2

[GMHP04] GROCHOW K., MARTIN S., HERTZMANN A., POPOVIĆ Z.: Style-based inverse kinematics. *ACM Transactions on Graphics (TOG) 23*, 3 (2004), 531. 1

[JLSX10] JI H., LIU C., SHEN Z., XU Y.: Robust video denoising using low rank matrix completion. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (2010), IEEE, pp. 1791–1798. 2

[Law] LAWRENCE N.: Mocap toolbox for matlab. *Available online at http://www. cs. man. ac. uk/neill/mocap*. 2

[LC10] LOU H., CHAI J.: Example-Based Human Motion Denoising. *Visualization and Computer Graphics, IEEE Transactions on 16*, 5 (2010), 870–879. 1, 2, 3

[LE06] LEE C., ELGAMMAL A.: Human motion synthesis by motion manifold learning and motion primitive segmentation. *Lecture Notes in Computer Science 4069* (2006), 464–473. 1, 2

[LFAJ10] LI Y., FERMULLER C., ALOIMONOS Y., JI H.: Learning shift-invariant sparse representation of actions. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (2010), IEEE, pp. 2630–2637. 2

[LWS02] LI Y., WANG T., SHUM H.: Motion texture: a two-level statistical model for character motion synthesis. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques* (2002), ACM New York, NY, USA, pp. 465–472. 1

[MGC09] MA S., GOLDFARB D., CHEN L.: Fixed point and Bregman iterative methods for matrix rank minimization. *Mathematical Programming* (2009), 1–33. 2

[PB02] PULLEN K., BREGLER C.: Motion capture assisted animation: Texturing and synthesis. *ACM Transactions on Graphics (TOG) 21*, 3 (2002), 508. 1

[SHP04] SAFONOVA A., HODGINS J., POLLARD N.: Synthesizing physically realistic human motion in low-dimensional, behavior-specific spaces. In *ACM SIGGRAPH 2004 Papers* (2004), ACM, p. 521. 1

[TWC*09] TOURNIER M., WU X., COURTY N., ARNAUD E., REVERET L.: Motion compression using principal geodesics analysis. In *Computer Graphics Forum* (2009), vol. 28, Wiley Online Library, pp. 355–364. 1

[WFH08] WANG J., FLEET D., HERTZMANN A.: Gaussian process dynamical models for human motion. *IEEE transactions on pattern analysis and machine intelligence 30*, 2 (2008), 283–298. 1