Spatio-Temporal Reconstruction for 3D Motion Recovery

Jingyu Yang, *Senior Member, IEEE,* Xin Guo, Kun Li, *Member, IEEE,* Meiyuan Wang, Yu-Kun Lai, *Member, IEEE,* and Feng Wu, *Fellow, IEEE*

***Abstract*—This paper addresses the challenge of 3D motion recovery by exploiting the spatio-temporal correlations of corrupted 3D skeleton sequences. We propose a new 3D motion recovery method using spatio-temporal reconstruction, which uses joint low-rank and sparse priors to exploit *temporal* correlation and an isometric constraint for *spatial* correlation. The proposed model is formulated as a constrained optimization problem, which is efficiently solved by the augmented Lagrangian method with a Gauss-Newton solver for the subproblem of isometric optimization. Experimental results on the CMU motion capture dataset, Edinburgh dataset and two Kinect datasets demonstrate that the proposed approach achieves better motion recovery than state-of-the-art methods. The proposed method is applicable to Kinect-like skeleton tracking devices and pose estimation methods that cannot provide accurate estimation of complex motions, especially in the presence of occlusion.**

***Index Terms*—3D skeleton, motion recovery, spatio-temporal, sparse, occlusion.**

I. Introduction

OBSERVATION of human activities has always been an active research topic in computer vision and computer graphics, which includes many research fields, *e.g.*, pose estimation [1], [2], gesture recognition [3], [4], motion prediction [5], [6], and 3D reconstruction [7], [8]. One of the key technologies in these fields is the accurate estimation of human motion. However, few motion capture devices could seize accurate human motion. Traditional motion capture systems have increased the research cost of these specific fields with their numerous shortcomings: inconvenient implementation, expensive prices, difficulty to maintain, and requirement of many manual operations. Microsoft Kinect for Xbox 360 (“Kinect”) has shed a light on human motion capture. With the advent of Microsoft Kinect and similar devices, significant effort and advances [9], [10] have been made in recent years for low-cost, accessible human motion tracking systems. This however is achieved at the cost of sacrificing capture accuracy, so skeletons captured by these low-cost and portable devices such as the Kinect often suffer from severe joint drifting and motion jitter, especially in the presence of self-occlusion or object occlusion [11]. The accuracy of skeleton estimation is more satisfactory for controlled non-occluded simple motions, such as standing upright and walking forward, which has apparent limitations in real-world circumstances.

This paper addresses the challenge of recovering accurate and smooth human motions from corrupted 3D skeleton sequences which is a fundamental problem in human motion estimation. Our method is based on the observation of both *spatial* and *temporal* inner correlation in skeleton sequences, and thus is able to accurately recover clean and smooth skeleton motions. In our model, the skeleton sequence is regularized by joint low-rank and sparse priors to exploit *temporal* correlation between frames and simultaneously by an isometric prior to exploit *spatial* correlation of skeleton structure. To efficiently solve this model, we derive an alternating direction method under the augmented Lagrangian multiplier (ADM_ALM) framework. The effectiveness of our method is demonstrated by experiments on the CMU dataset [12], Edinburgh dataset [13] [14] and two real captured Kinect datasets [15], obtaining better recovery accuracy than state-of-the-art methods.

The contributions of this paper are summarized as follows:

- We propose a novel *spatio-temporal* reconstruction model to recover accurate and smooth motions from corrupted 3D skeleton sequences. The sparse and low-rank constraints guarantee the plausibility of human motions to ensure smooth recovery of motion sequences, while the isometric constraint promotes the isometry of bone lengths to ensure accurate recovery of joint positions. The proposed method significantly extends 3D motion reconstruction methods for direct recovery of 3D skeletons, unlike most previous methods relying on 2D images or 1D motion trajectories;
- We derive an ADM_ALM algorithm to decouple non-differentiable terms into simpler subproblems, and integrate the Gauss-Newton method to solve the non-liner subproblem.

As the extended version of our previous conference papers [16], [17], this paper exploits both spatial and temporal constraints in a uniform framework and adds a sparse prior with wavelet transform to improve the smoothness of recovery. The comparison of these three methods are summed up in Table I.

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant 61571322 and Grant 61771339.

Jingyu Yang, Xin Guo and Meiyuan Wang are with the School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China.s

Kun Li is with the Tianjin Key Laboratory of Cognitive Computing and Application, College of Intelligence and Computing, Tianjin University, Tianjin 300350, China.

Yu-Kun Lai is with the School of Computer Science and Informatics, Cardiff University, Wales, UK.

Feng Wu is with University of Science and Technology of China, Hefei, China.

Corresponding author: Kun Li (lik@tju.edu.cn)

TABLE I
METHOD COMPARISON OF THIS WORK AND OUR PREVIOUS WORK.

Method	Priors	Benefits
Wang [16]	low rank	robust to noise and outliers
Li [17]	low rank + isometry	robust to noise and outliers and exploring spatial correlation
Ours	low rank + isometry + sparse smoothness	robust to noise and outliers and exploring spatio-temporal correlation

We also perform thorough evaluation and include in-depth discussion. Related work is summarized in Section II. Section III describes the proposed 3D motion recovery method, including motivation and a novel recovery model. Section IV provides the experimental results, and the conclusions are finally drawn in Section V.

II. Related Work

Motion recover is a challenging problem in computer graphics and computer vision, and thus has attracted more and more attentions. Various models and algorithms have been developed across different fields such as pose estimation and trajectory reconstruction. We review related work of two categories: model-based methods and learning-based methods.

A. Model-based Methods

To recover 3D motion information from 2D images is a highly ill-posed problem due to many factors such as inaccurate joint detection and motion occlusion. In model-based methods, motion/pose information is recovered by solving optimization models that incorporate priors for regularization. Menier *et al.* [18] developed a generative model on a skeletal articulated structure to estimate 3D motion information from multiple views, which is solved via the expectation-maximization (EM) approach. This method is robust to several types of perturbations in the model or data, but the requirement of multi-view input limits its application to the more common single-view scenarios. For motion recovery from a monocular camera, Park *et al.* [19] reconstructed a 3D smooth articulated trajectory from a 2-D trajectory extracted from a monocular image sequence by using the spherical coordinate representation of a relative trajectory. To tackle the NP-hard binary quadratic programming, a branch-and-bound routine with binary relaxation is used to approximate the solution. To overcome the inefficiency of branch-and-bound searching, Valmadre *et al.* [20] proposed a dynamic programming approach combining articulation constraints with temporal smoothness. Leonardos *et al.* [21] introduced spherical tangent bundles and a Riemannian Extended Kalman Filter (REKF) model into the human motion reconstruction, achieving accurate reconstruction from image sequences with corrupted skeletons. Many methods, such as skeleton-driven skinning [22], [23] and character animation [24], rely on the foreknown accurate skeleton structure obtained from capture systems or skeleton estimation methods.

Various motion recovery models have been also designed for the recovery of degraded motion data from motion capture systems or body sensing devices. Due to the physical constraints of human bodies such as articulated structure of skeletons and speed-limited motion, motion trajectories of skeletons only lie in the manifold/subspace of their ambient signal space. Some works [16], [25], [26] used low-rank matrix completion to exploit such low-dimensional structure so that missing measurements could be recovered from captured data. However, these models did not consider the *spatial* correlation due to the skeleton structure, which would result in large joint errors in challenging cases. To address this limitation, Li *et al.* [17] explored the *spatial* correlation between skeleton sequences by introducing an isometry constraint, which encourages the bone length to be consistent. Despite the prominent performance for most cases, their results still contain slight jittering due to the lack of *temporal* regularization.

The reviewed works show that model-based methods have achieved promising performance in various motion recovery tasks. The key for accurate recovery is to fully take advantages of various correlation by imposing powerful priors. Along this avenue, our method exploits temporal correlation via joint low-rank and sparse priors, and exploits spatial correlation via an isometric constraint. As a result, the proposed method achieves accurate recovery for real motion data, and is robust to various types of degradation such as noise and occlusion.

B. Learning-based Methods

In contrast to model-based methods, another category of approaches recover 3D motion signals via learning techniques. Toshev *et al.* [27] formulated the estimation of human poses from RGB images as a joint regression problem solved by a Deep Neural Network. Ouyang *et al.* [28] fused three types of features, including appearance score [29], mixture type [30] and deformation [31], into a deep model to learn human poses. However, as an image-based method, the performance could be affected by image quality. Since the prevalence of depth sensors [32], [33], pose estimation benefits a lot from depth information. Wei *et al.* [34] developed an automatic motion capture system by integrating depth data, full-body geometry, silhouette information, and temporal pose priors into a *Maximum A Posteriori* (MAP) framework, achieving state-of-the-art capture accuracy. However, it is difficult to give accurate pose hypothesis when the body part is invisible due to occlusion.

There are also data-driven approaches to recover 3D motion from incomplete and/or corrupted observations. Shotton *et al.* [35] synthesized full-body motion from sparse control signals by learning a series of local models from a database of human motion. To automatically detect and repair corrupted/wrong joints, Chai *et al.* [36] adopted local PCA (Principal Component Analysis) to produce a manifold that includes various types of human motion data, and applied it for synthesizing movements from low dimensional signals. Aristidou *et al.* [37] proposed to automatically analyze and fix motion capture sequences by using self-similarity analysis, but they focused on the suppression of joint rotation errors and did not consider motion dynamics or bone-length violations. Saito *et al.* [38] recovered corrupted skeletons by finding a subspace of valid motions, namely the motion manifold. After learning the motion manifold with convolutional autoencoders, corrupted skeletons are projected onto the motion manifold, and valid motions are finally rebuilt through inverse projection.

Learning-based methods are able to learn highly non-linear mappings such as the image-to-joint regression. However, huge amount of ground-truth motion data is difficult to acquire, particularly for the dataset scale required by the deep learning paradigm. It would be interesting to investigate motion recovery with learning techniques requiring less labeled data, such as few-shot learning, semi-supervised learning, or unsupervised learning.

III. The Proposed Method

In this section, we first present the motivation of the proposed model, and then detail the proposed 3D skeleton recovery model which explores spatio-temporal correlation with joint low-rank and sparse priors and an isometric prior. Finally, we derive an efficient algorithm under the ALM framework.

A. Motivation

Human skeleton sequences captured by devices like the Kinect are often polluted by severe noise or outliers, especially in the presence of self-occlusion or object occlusion, which makes the skeleton recovery problem challenging yet important for practical applications. Most skeleton recovery approaches [21], [19], [18] require either RGB-D images or silhouettes as auxiliary input which are not always available. We observe that a skeleton sequence is a set of time series that lies in a low-dimensional subspace, and is possible to be recovered from a partially-observed and/or noisy version. Specifically, we observe the following priors for skeleton signals:

1) Isometric Prior: As shown in Fig. 1(b), the motion trajectories of a parent joint and its child joint, *e.g.*, joint 3 and joint 4, are often nearly parallel as the length of the rigid bone is constant over the time. We also note that such an isometry property only occurs between the parent joint and child joint, corresponding to the ends of a bone. As shown in Fig. 1(b), there is no obvious correlation between the trajectories of joint 3 and joint 19. Therefore, we impose an isometric constraint $E_{iso}(\mathbf{A})$ to encourage isometry during the recovery.

2) Low-Rank Prior: Human motions lie in a low-dimensional subspace [36]. Low-rank approximation is a recent advance in low-dimensional representation of signals. To investigate the potential of a low-rank prior in modeling skeleton signals, we form a motion matrix $\mathbf{D}$ by concatenating the skeleton positions over time (see the definition in Eq. (1)), and evaluate low-rank approximation accuracy in terms of relative error $RE := \|\mathbf{D} - \bar{\mathbf{D}}\|_F / \|\mathbf{D}\|_F$, where $\mathbf{D}$ denotes the input motion matrix and $\bar{\mathbf{D}}$ is the approximated matrix with a small rank, and $\|.\|_F$ represents the Frobenius-norm of a matrix. As shown in Fig. 1(c), the approximation errors for all the five skeleton matrices decay dramatically and approach to zero as the rank of the reconstructed matrix increases, which suggests the low-rankness of the skeleton matrix [39].

3) Sparse prior: The types of human motion are limited due to physical structure. We observe that skeleton trajectories, as shown in Fig. 1(b), are piece-wise smooth with discontinuities at turning points of the motion, and x, y and z components of motion trajectories can also be considered as 1-D temporal piece-wise smooth signals with a number of singularities, which are able to be efficiently represented by wavelet transforms [40]–[42].

To verify this, Fig. 1(d) shows the energy compaction efficiency in terms of normalized energy with respect to the percentage of retained largest wavelet coefficients of motion signals. The curves in Fig. 1(d) indicate that the wavelet transform is able to approximate joint trajectories with only a small fraction of non-zero coefficients, and hence has a sparse representation. Reliable approximation of motion signals would require about 10% of wavelet coefficients, including not only the DC component but also many other meaningful components. We also evaluate the approximation performances of four well-known (bi-)orthogonal wavelet transforms with similar filter lengths and vanishing moments, *e.g.*, Coiflets (coif3), Symlets (sym5), Biorthogonal wavelets (bior4.4), and Daubechies wavelets (db5) on a sequence in Fig. 1(d). The four wavelets are equally powerful in representing the 3D motion data of skeleton with only a small fraction of coefficients. In our implementation, we use the Daubechies wavelet transform for its slightly better performance although others yield similar results. This motivates us to use a wavelet sparsity prior to model the temporal correlation of joint motions, complementing the low-rank prior that emphasizes both spatial and temporal correlation.

Based on the key observations above, we propose a skeleton recovery model from partially-observed and noisy data with the isometric prior and joint low-rank and sparse priors.

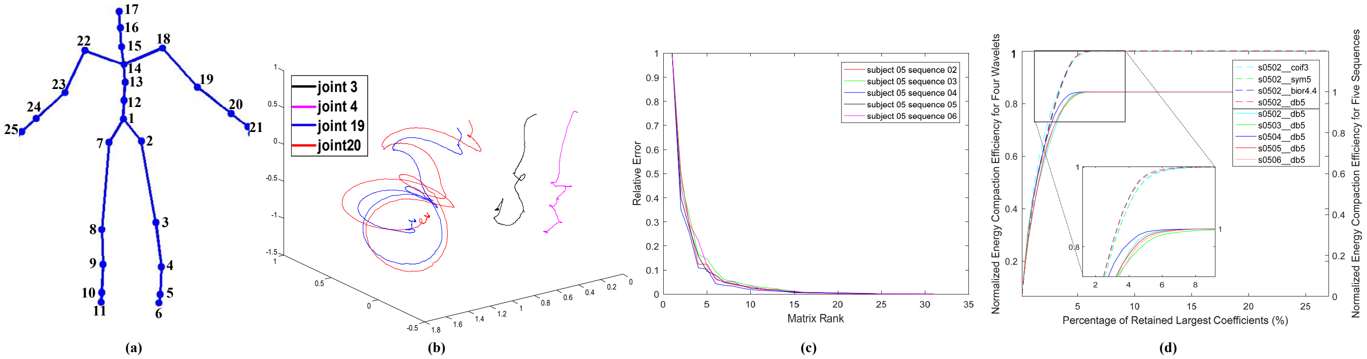
B. The Proposed Model

Let $\mathbf{n}_i = (n_{ix}, n_{iy}, n_{iz})^\top$ be the i-th joint of the skeleton, where n_{ix}, n_{iy} and n_{iz} represent the joint's x, y and z coordinates, respectively, $i \in \{1, 2, \cdots, S\}$, and S is the number of skeletal joints. Denote by $\mathbf{n}_i^t$ the coordinates of the i-th joint at frame t, and by T the number of frames. The corrupted motion matrix $\mathbf{D} \in \mathbb{R}^{3T \times S}$ is denoted by:

$$\mathbf{D} = \begin{pmatrix} \mathbf{n}_1^1 & \cdots & \mathbf{n}_S^1 \\ \vdots & \ddots & \vdots \\ \mathbf{n}_1^T & \cdots & \mathbf{n}_S^T \end{pmatrix}, \tag{1}$$

where each group of three rows corresponds to a skeleton at one frame, and each column corresponds to the temporal trajectory of one joint. We assume an additive observation model:

$$\mathbf{D} = \mathbf{A} + \mathbf{E}, \tag{2}$$



(a) (b) (c) (d)

Fig. 1. Motivation for the proposed priors: (a) skeleton structure, (b) motion trajectories of skeletal joints No. 3, 4, 19 and 20 of subject 05 sequence 02 in the CMU dataset [12], (c) relative error *w.r.t.* matrix rank, and (d) energy compaction efficiency for four wavelets (left y-axis) on a sequence and for five sequences with the same Daubechies wavelet (right y-axis) *w.r.t.* the percentage of retained largest wavelet coefficients.

where $\mathbf{A}$ is the latent clean skeleton matrix, and $\mathbf{E}$ represents the error matrix. Skeleton corruption often happens in challenging scenarios such as occlusion, so the error matrix $\mathbf{E}$ should be sparse. Base on the three observations in Sec. III-A, the 3D motion recovery problem can be formulated as

$$\begin{aligned} \min_{\mathbf{A},\mathbf{E}} rank(\mathbf{A}) + \lambda \left\|\mathbf{E}\right\|_0 + \frac{\gamma}{2} E_{\text{iso}}\left(\mathbf{A}\right) + \mu E_{\text{smooth}}\left(\mathbf{A}\right) \\ \text{s.t.} \quad \mathbf{D} = \mathbf{A} + \mathbf{E}, \end{aligned} \tag{3}$$

where $rank(\mathbf{A})$ is the rank of matrix $\mathbf{A}$, $\left\|\mathbf{E}\right\|_0$ is the ℓ_0-norm of matrix $\mathbf{E}$ which represents the number of non-zero entries in the matrix, $E_{\text{iso}}\left(\mathbf{A}\right)$ is the isometry term encouraging bone-length isometry, $E_{\text{smooth}}\left(\mathbf{A}\right)$ is the smoothness term to ensure smooth motion recovery, and $\lambda > 0$, $\gamma > 0$, $\mu > 0$ are weighting parameters to balance these terms.

The problem in Eq. (3) is NP-hard due to $rank(\mathbf{A})$ and $\left\|\mathbf{E}\right\|_0$. This is made tractable by replacing $rank(\mathbf{A})$ with its convex substitute known as the nuclear norm $\left\|\mathbf{A}\right\|_* := \sum_i \sigma_i$, where σ_i is a singular value of matrix $\mathbf{A}$, and by replacing ℓ_0 norm of matrix $\mathbf{E}$ with the ℓ_1 norm $\left\|\mathbf{E}\right\|_1 := \sum_{ij} \left|\mathbf{E}_{ij}\right|$ [43]. So, we obtain the following optimization problem:

$$\begin{aligned} \min_{\mathbf{A},\mathbf{E}} \left\|\mathbf{A}\right\|_* + \lambda \left\|\mathbf{E}\right\|_1 + \frac{\gamma}{2} E_{\text{iso}}\left(\mathbf{A}\right) + \mu E_{\text{smooth}}\left(\mathbf{A}\right) \\ \text{s.t.} \quad \mathbf{D} = \mathbf{A} + \mathbf{E}, \end{aligned} \tag{4}$$

where the isometry term $E_{\text{iso}}\left(\mathbf{A}\right)$ and smoothness term $E_{\text{smooth}}\left(\mathbf{A}\right)$ are detailed in Section III-B1 and Section III-B2, respectively. Low-rankness measured by the nuclear norm regularizes that the rows of the motion matrix $\mathbf{A}$ are highly linearly dependent as the motion patterns of human skeleton lie in a low-dimensional subspace in the ambient signal space, which implies that the motion matrix $\mathbf{A}$ can be expressed as linear combinations of some basis poses. Proper selection of the parameter λ is crucial to recovery accuracy [44]: λ should be small enough to remove noise (by keeping the variance low to obtain high stability), and large enough not to overshrink the original matrix (by keeping the bias low for flexible motion). γ and μ are set to balance the energy of corresponding terms. Since the error of the isometry term could be extremely small due to the bone-length error to the fourth order (details will be provided later), γ should be large enough to maintain the importance of the isometry term while μ should be small enough to provide sufficient flexibility in formulating the wavelet term. See Section IV for the parameters used in our experiments.

1) Exploring Spatial Correlation with Isometry: The isometry term E_{iso} is designed to model the spatial correlation of motions on the skeleton structure, usually known as an articulation skeleton, so that the recovered motions are reasonable. An articulated skeleton is usually described by a tree structure, where each node represents a skeletal joint and each edge between nodes represents a bone. The body size is fixed for a particular actor and the bones have constant lengths over time.

Therefore, we exploit spatial coherence of skeletons by promoting isometry (*i.e.* length preservation) of bones [17]. Let $G := (\mathcal{V}, \mathcal{E})$ be a skeleton, where $\mathcal{V}$ is the set of skeletal joints, and $\mathcal{E}$ is the set of bones. e_{ij} represents the bone of the skeleton connecting the i-th and the j-th joints. We introduce an energy term that penalizes non-isometric deformation:

$$E_{\text{iso}}\left(\mathbf{A}\right) = \sum_{t=1}^{T} \sum_{e_{ij} \in \mathcal{E}} \left(d^2\left(\mathbf{n}_i^t, \mathbf{n}_j^t\right) - l_{ij}^2\right)^2, \tag{5}$$

where l_{ij} is the bone length between two joints. $d\left(\mathbf{n}_i^t, \mathbf{n}_j^t\right)$ denotes the distance between joints $\mathbf{n}_i^t$ and $\mathbf{n}_j^t$, and $d\left(\mathbf{n}_i^t, \mathbf{n}_j^t\right) := \left\|\mathbf{n}_i^t - \mathbf{n}_j^t\right\|_2$ is the Euclidean distance between $\mathbf{n}_i^t$ and $\mathbf{n}_j^t$ at time instance t.

The isometry term aims at preserving the bone lengths of the skeleton. Such a constraint helps to avoid inaccurate recovery in which relative positions of joints are beyond a reasonable range.

2) Exploring Temporal Correlation with Wavelet Transform: The types of human motion are limited due to physical structure. Motion trajectories are mainly smooth signals with singularities, which can be well modeled by wavelet transform. Let $\mathbf{W}$ be the wavelet basis matrix with J-level decompositions ($J = 2$ in our experiments). The wavelet coefficients of skeleton motion should be sparse. Therefore, the sparseness of the smoothness term $E_{\text{smooth}}\left(\mathbf{A}\right)$ is measured by the ℓ_1-norm :

$$E_{\text{smooth}}\left(\mathbf{A}\right) = \left\|\mathbf{W}\mathbf{A}\right\|_1 . \tag{6}$$

Substituting the smoothness term (6) and the isometry term (5) into Eq. (4), the 3D motion recovery model is rewritten as:

$$\min_{\mathbf{A},\mathbf{E}} \left\|\mathbf{A}\right\|_* + \lambda\left\|\mathbf{E}\right\|_1 + \frac{\gamma}{2}E_{\text{iso}}\left(\mathbf{A}\right) + \mu\left\|\mathbf{W}\mathbf{A}\right\|_1. \quad (7)$$

The isometry term exploits the spatial correlation of a skeleton sequence by suppressing positional errors of skeletal joints, while the smoothness term with sparse prior exploits the temporal correlation of skeleton sequence by ensuring the piece-wise smoothness of the recovered motion. In this way, the proposed model is able to fully exploit the characteristics of skeleton motions.

C. Augmented Lagrangian Algorithm

The proposed model (7) contains a low-rank term, a non-differentiable term (wavelet term), and a nonlinear term (isometry term), which are difficult to optimize simultaneously. Therefore, we introduce two auxiliary variables $\mathbf{C}$ and $\mathbf{N}$ to decouple these terms, resulting in the following formulation.

$$\begin{aligned}\min_{\mathbf{A},\mathbf{E},\mathbf{C},\mathbf{N}} & \left\|\mathbf{A}\right\|_* + \lambda\left\|\mathbf{E}\right\|_1 + \mu\left\|\mathbf{C}\right\|_1 + \frac{\gamma}{2}E_{\text{iso}}\left(\mathbf{N}\right)\\ \text{s.t.} \quad & \mathbf{D} = \mathbf{A} + \mathbf{E}, \quad \mathbf{N} = \mathbf{A}, \quad \mathbf{C} = \mathbf{W}\mathbf{A}.\end{aligned} \quad (8)$$

To convert Problem (8) with equality constraints into unconstrained optimization, we utilize the augmented Lagrangian method [45]. For compact notation, denote the variable set by $\mathbf{\Theta} \triangleq \{\mathbf{A},\mathbf{E},\mathbf{C},\mathbf{N}\}$, the multiplier set by $\boldsymbol{\mathcal{Z}} \triangleq \{\mathbf{Z}_1,\mathbf{Z}_2,\mathbf{Z}_3\}$, and the penalty parameter set by $\boldsymbol{\rho} \triangleq \{\rho_1,\rho_2,\rho_3\}$. The augmented Lagrangian function of (8) is defined as follows.

$$\begin{aligned}L(\mathbf{\Theta},\boldsymbol{\mathcal{Z}},\boldsymbol{\rho}) = {} & \left\|\mathbf{A}\right\|_* + \lambda\left\|\mathbf{E}\right\|_1 + \frac{\gamma}{2}E_{\text{iso}}\left(\mathbf{N}\right) + \mu\left\|\mathbf{C}\right\|_1\\ & + \langle\mathbf{Z}_1,\mathbf{E}-\mathbf{D}+\mathbf{A}\rangle + \frac{\rho_1}{2}\left\|\mathbf{E}-\mathbf{D}+\mathbf{A}\right\|_F^2\\ & + \langle\mathbf{Z}_2,\mathbf{N}-\mathbf{A}\rangle + \frac{\rho_2}{2}\left\|\mathbf{N}-\mathbf{A}\right\|_F^2\\ & + \langle\mathbf{Z}_3,\mathbf{C}-\mathbf{W}\mathbf{A}\rangle + \frac{\rho_3}{2}\left\|\mathbf{C}-\mathbf{W}\mathbf{A}\right\|_F^2, \quad (9)\end{aligned}$$

where $\langle\cdot,\cdot\rangle$ denotes the inner product of two matrices considered as long vectors.

Under the ALM framework, the original Problem (8) is solved by iteratively minimizing the unconstrained augmented Lagrangian function [45].

$$\begin{cases}\mathbf{\Theta}^{k+1} = \min_{\mathbf{\Theta}} L\big(\mathbf{\Theta},\boldsymbol{\mathcal{Z}}^k,\boldsymbol{\rho}^k\big),\\ \text{Update } \boldsymbol{\mathcal{Z}}^{k+1} \text{ and } \boldsymbol{\rho}^{k+1},\end{cases} \quad (10)$$

where the update of multipliers and penalty parameters are detailed in Algorithm 1.

However, jointly optimizing variables in $\mathbf{\Theta}$ in (10) is still difficult since the three regularizer terms in terms of $\mathbf{A}$, $\mathbf{E}$, and $\mathbf{C}$, respectively, are non-differentiable. Note that, under mild conditions, the alternating optimization converges to the solution of the original joint optimization [46]. We resort to the alternating direction method (ADM) to solve variables in $\mathbf{\Theta}$ separately as subproblems instead of directly solving Problem (10). In each subproblem, only one variable is optimized while other variables are fixed at their up-to-date values. As a result, each subproblem becomes simpler and easier to solve as many terms in (8) are irrelevant as constants. The alternating optimization of subproblems are detailed as follows:

1) $\mathbf{C}$-subproblem: Those terms irrelevant to $\mathbf{C}$ are considered as constants. Then, we obtain the following ℓ_1-norm minimization:

$$\begin{aligned}\mathbf{C}^{k+1} = \arg\min_{\mathbf{C}^k} & \, \mu\left\|\mathbf{C}^k\right\|_1 + \left\langle\mathbf{Z}_{\mathbf{3}}^k,\mathbf{C}^k{-}\mathbf{W}\mathbf{A}^k\right\rangle\\ & + \frac{\rho_3}{2}\left\|\mathbf{C}^k{-}\mathbf{W}\mathbf{A}^k\right\|_F^2, \quad (11)\end{aligned}$$

which has the following explicit solution: $S_{\frac{\mu}{\rho_3}}\left(\mathbf{W}\mathbf{A}^k - \frac{1}{\rho_3}\mathbf{Z}_3^k\right)$. The shrinkage operator $S_\delta(x) := \operatorname{sgn}(x)\max(|x| - \delta, 0)$ is applied to the matrix entry-wise.

2) $\mathbf{E}$-subproblem: Similarly, the $\mathbf{E}$-subproblem is equivalent to the following optimization problem ignoring constant terms with respect to $\mathbf{E}$:

$$\begin{aligned}\mathbf{E}^{k+1} = \arg\min_{\mathbf{E}^k} & \, \lambda\left\|\mathbf{E}^k\right\|_1 + \left\langle\mathbf{Z}_{\mathbf{1}}^k,\mathbf{E}^k{-}\mathbf{D}{+}\mathbf{A}^k\right\rangle\\ & + \frac{\rho_1}{2}\left\|\mathbf{E}^k{-}\mathbf{D}{+}\mathbf{A}^k\right\|_F^2, \quad (12)\end{aligned}$$

which has the following explicit solution: $S_{\lambda/\rho_1}\big(\mathbf{D} - \mathbf{A}^k - \frac{1}{\rho_1}\mathbf{Z}_1^k\big)$.

3) $\mathbf{A}$-subproblem: The A-subproblem is the following nuclear norm minimization problem

$$\begin{aligned}\mathbf{A}^{k+1} = & \arg\min_{\mathbf{A}^k}\left\|\mathbf{A}^k\right\|_*\\ & + \left\langle\mathbf{Z}_{\mathbf{1}}^k,\mathbf{E}^k{-}\mathbf{D}{+}\mathbf{A}^k\right\rangle + \frac{\rho_1}{2}\left\|\mathbf{E}^k{-}\mathbf{D}{+}\mathbf{A}^k\right\|_F^2\\ & + \left\langle\mathbf{Z}_{\mathbf{2}}^k,\mathbf{N}^k{-}\mathbf{A}^k\right\rangle + \frac{\rho_2}{2}\left\|\mathbf{N}^k{-}\mathbf{A}^k\right\|_F^2\\ & + \left\langle\mathbf{Z}_{\mathbf{3}}^k,\mathbf{C}^k{-}\mathbf{W}\mathbf{A}^k\right\rangle + \frac{\rho_3}{2}\left\|\mathbf{C}^k{-}\mathbf{W}\mathbf{A}^k\right\|_F^2. \quad (13)\end{aligned}$$

Note that Eq. (13) is not a standard nuclear minimization problem that has the closed-form solution. For easier optimization, we choose an orthogonal wavelet basis *Daubechies 10*, which implies $\mathbf{W}^\top\mathbf{W} = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix. Then, we have $\left\langle\mathbf{Z}_{\mathbf{3}}^k,\mathbf{C}^k{-}\mathbf{W}\mathbf{A}^k\right\rangle = \left\langle\mathbf{W}^\top\mathbf{Z}_{\mathbf{3}}^k,\mathbf{W}^\top\mathbf{C}^k - \mathbf{A}^k\right\rangle$ and $\left\|\mathbf{C}^k{-}\mathbf{W}\mathbf{A}^k\right\|_F = \left\|\mathbf{W}^\top\mathbf{C}^k{-}\mathbf{A}^k\right\|_F$. With substitution, Eq. (13) is transformed into the following standard nuclear norm minimization:

$$\begin{aligned}\min_{\mathbf{A}} & \left\|\mathbf{A}\right\|_* + \langle\mathbf{Z}_{\mathbf{1}},\mathbf{E}-\mathbf{D}+\mathbf{A}\rangle + \frac{\rho_1}{2}\left\|\mathbf{E}-\mathbf{D}+\mathbf{A}\right\|_F^2\\ & + \langle\mathbf{Z}_{\mathbf{2}},\mathbf{N}-\mathbf{A}\rangle + \frac{\rho_2}{2}\left\|\mathbf{N}-\mathbf{A}\right\|_F^2\\ & + \left\langle\mathbf{W}^\top\mathbf{Z}_{\mathbf{3}},\mathbf{W}^\top\mathbf{C}-\mathbf{A}\right\rangle + \frac{\rho_3}{2}\left\|\mathbf{W}^\top\mathbf{C}{-}\mathbf{A}\right\|_F^2. \quad (14)\end{aligned}$$

Suppose $\mathbf{H}_1^{k+1} = \mathbf{D}{-}\mathbf{E}^{k+1}{-}\frac{1}{\rho_1}\mathbf{Z}_{\mathbf{1}}^k$, $\mathbf{H}_2^{k+1} = \mathbf{N}^{k+1}{+}\frac{1}{\rho_2}\mathbf{Z}_2^k$, and $\mathbf{H}_3^{k+1} = \mathbf{W}^\top\mathbf{C}^{k+1} + \frac{1}{\rho_3}\mathbf{W}^\top\mathbf{Z}_3^k$. Then the solution of Eq. (14) is the closed-form singular value thresholding: $\mathbf{A}^{k+1} = \mathbf{U}S_\delta(\mathbf{\Lambda})\mathbf{V}^\top$, where $(\mathbf{U},\mathbf{\Lambda},\mathbf{V}) = svd\left(\frac{\rho_1^k\mathbf{H}_1^k+\rho_2^k\mathbf{H}_2^k+\rho_3^k\mathbf{H}_3^k}{\rho_1^k+\rho_2^k+\rho_3^k}\right)$ and $\delta = 1/(\rho_1^k + \rho_2^k + \rho_3^k)$.

4) $\mathbf{N}$-subproblem: By applying the $\mathbf{N}$-subproblem, the proxy variable $\mathbf{N}$ is regularized to conform to the isometry constraint, otherwise the skeleton would deform to unreasonable shapes. Such a constraint is passed to the target variable $\mathbf{A}$ by solving the $\mathbf{A}$-subproblem, which involves the auxiliary

variable $\mathbf{N}^k$, in the iterative optimization procedure. The $\mathbf{N}$-subproblem is a nonlinear least squares (NLS) problem:

$$\min_{\mathbf{N}} L\left(\mathbf{A}, \mathbf{E}, \mathbf{N}, \mathbf{C}, \mathbf{Z}_1, \mathbf{Z}_2, \mathbf{Z}_3\right), \tag{15}$$

which does not have a closed-form solution as previous three sub-problems. We solve the NLS using the Gauss-Newton method. Specifically, we rewrite Eq. (5) into the following standard NLS problem:

$$E_{\text{iso}}\left(\mathbf{N}\right) = \left\|\mathbf{F}\left(\mathbf{N}\right)\right\|^2, \mathbf{F}\left(\mathbf{N}\right) = \left[r_{11}\left(\mathbf{N}\right), \cdots, r_{TH}\left(\mathbf{N}\right)\right]^\top, \tag{16}$$

where $r_{th}(\cdot)$ is the energy term related to the h-th bone of the t-th frame. Given $\mathbf{N}^k$, we linearize $\mathbf{F}\left(\mathbf{N}\right)$ by the first-order Taylor expansion:

$$\mathbf{F}\left(\mathbf{N}\right) \approx \mathbf{F}\left(\mathbf{N}^k\right) + \mathbf{J}\left(\mathbf{N}^k\right)\boldsymbol{\delta}, \quad \boldsymbol{\delta} = \mathbf{N} - \mathbf{N}^k, \tag{17}$$

where $\mathbf{J}\left(\mathbf{N}^k\right)$ is the Jacobian of $\mathbf{F}$ evaluated at $\mathbf{N}^k$, and $\boldsymbol{\delta}$ is the deviation against $\mathbf{N}^k$. Instead of solving Eq. (15), we optimize the approximate objective function to obtain the update of $\mathbf{N}$ to decrease the energy cost:

$$\boldsymbol{\delta}^{k+1} = \arg\min_{\boldsymbol{\delta}} \left\|\mathbf{F}\left(\mathbf{N}^k\right) + \mathbf{J}\left(\mathbf{N}^k\right)\boldsymbol{\delta}\right\|^2. \tag{18}$$

The optimal update step $\boldsymbol{\delta}^{k+1}$ is the solution of the corresponding normal equations:

$$\mathbf{J}\left(\mathbf{N}^k\right)^\top \mathbf{J}\left(\mathbf{N}^k\right)\boldsymbol{\delta} = -\mathbf{J}\left(\mathbf{N}^k\right)^\top \mathbf{F}\left(\mathbf{N}^k\right), \tag{19}$$

which can be solved using iterative solution techniques like preconditioned conjugate gradient (PCG). Previous works [47], [48] demonstrate the feasibility of this strategy in a GPU optimization framework for dynamics simulation and non-rigid registration, respectively. Combining Eq. (9) with the Gauss-Newton solver, the unknown update $\boldsymbol{\delta}^{k+1}$ can be solved by:

$$\begin{aligned}\boldsymbol{\delta}^{k+1} = &\left(\gamma\mathbf{J}^T\mathbf{J} + \rho_2^k\mathbf{I}\right)^{-1} \times \\ &\left(-\mathbf{Z}_2^k - \rho_2^k\left(\mathbf{N}^k - \mathbf{A}^k\right) - \gamma\mathbf{J}^\top\mathbf{F}\right),\end{aligned} \tag{20}$$

where $\mathbf{J}$ and $\mathbf{F}$ refer to $\mathbf{J}\left(\mathbf{N}^k\right)$ and $\mathbf{F}\left(\mathbf{N}^k\right)$, respectively. The overall ALM algorithm is summarized in Algorithm 1.

D. Convergence Analysis

The global convergence of ALM is proven in the case of two blocks, but it does not naturally apply to the cases of three or more blocks [46]. However, many signal processing tasks usually involve ALM problems of multiple blocks [26], [49], including our model (9) with four blocks, i.e., A, E, N, and C. Under mild conditions, the iteratively-updated variables of the ALM algorithm with multiple blocks converge to the Karush−Kuhn−Tucker (KKT) conditions, which are necessary conditions of the first-order optimality. We refer interested readers to dedicated literatures [46], [49]. Numerically, our algorithm usually converges to promising results after 30 iterations and is stable for various sequences. Fig. 2 shows three typical examples on the decreasing of the normalized total energy in iterations.

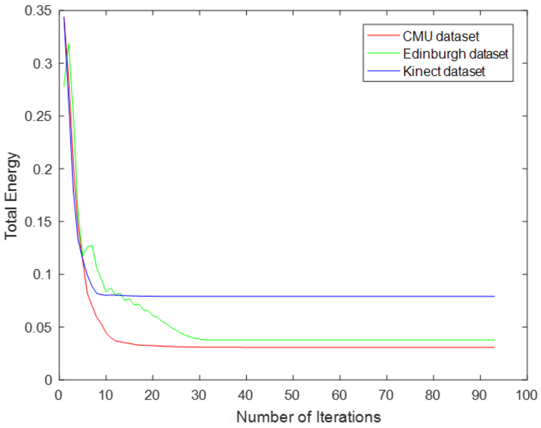


Fig. 2. The convergence curves of random sequences in CMU dataset [12], Edinburgh dataset [13], [14] and Kinect dataset.

Algorithm 1: ADM_ALM algorithm for 3D motion recovery

```
Input: observed skeleton matrix $\mathbf{D} \in \mathbf{R}^{m\times n}$
Initialize: $\mathbf{A}^0 = \mathbf{0}$, $\mathbf{E}^0 = \mathbf{0}$, $\mathbf{N}^0 = \mathbf{D}$,
            $\mathbf{Z}_1^0 = \mathbf{0}$, $\mathbf{Z}_2^0 = \mathbf{0}$,$\mathbf{Z}_3^0 = \mathbf{0}$,
            $\rho_1, \rho_2, \rho_3 > 0$, $\eta_1, \eta_2, \eta_3 > 1$, $maxIter = 1000$
while not converged do
    $\boldsymbol{\delta}^k = \left(\gamma\mathbf{J}^T\mathbf{J} + \rho_2^k\mathbf{I}\right)^{-1}$
        $\times\left[-\mathbf{Z}_2^k - \rho_2^k\left(\mathbf{N}^k - \mathbf{A}^k\right) - \gamma\mathbf{J}^T r\left(\mathbf{N}^k\right)\right]$
    $\mathbf{N}^{k+1} = \mathbf{N}^k + \delta^k$
    $\mathbf{C}^{k+1} = S_{\frac{\mu}{\rho_3^k}}\left(\mathbf{W}\mathbf{A}^k - \frac{1}{\rho_3}\mathbf{Z_3}^k\right)$
    $\mathbf{E}^{k+1} = S_{\frac{\lambda}{\rho_1^k}}\left(\mathbf{D} - \mathbf{A}^k - \frac{1}{\rho_1^k}\mathbf{Z}_1^k\right)$
    $\mathbf{H}_1^{k+1} = \mathbf{D} - \mathbf{E}^{k+1} - \frac{1}{\rho_1}\mathbf{Z}_1^k$
    $\mathbf{H}_2^{k+1} = \mathbf{N}^{k+1} + \frac{1}{\rho_2}\mathbf{Z}_2^k$
    $\mathbf{H}_3^{k+1} = \mathbf{W}^\top\mathbf{C}^{k+1} + \frac{1}{\rho_3}\mathbf{W}^\top\mathbf{Z}_3^k$
    $\mathbf{A}^{k+1} =$
    $M_{\frac{1}{\rho_1^k+\rho_2^k+\rho_3^k}}\left(\frac{\rho_1^k\mathbf{H}_1^{k+1}+\rho_2^k\mathbf{H}_2^{k+1}+\rho_3^k\mathbf{H}_3^{k+1}}{\rho_1^k+\rho_2^k+\rho_3^k}\right)$
    $\mathbf{Z}_1^{k+1} = \mathbf{Z}_1^k + \rho_1^k\left(\mathbf{E}^{k+1} - \mathbf{D} + \mathbf{A}^{k+1}\right)$
    $\mathbf{Z}_2^{k+1} = \mathbf{Z}_2^k + \rho_2^k\left(\mathbf{N}^{k+1} - \mathbf{A}^{k+1}\right)$
    $\mathbf{Z}_3^{k+1} = \mathbf{Z}_3^k + \rho_3^k\left(\mathbf{C}^{k+1} - \mathbf{W}\mathbf{A}^{k+1}\right)$
    $\rho_1^{k+1} = \eta_1\rho_1^k, \eta_1 > 1$
    $\rho_2^{k+1} = \eta_2\rho_2^k, \eta_2 > 1$
    $\rho_3^{k+1} = \eta_3\rho_3^k, \eta_3 > 1$
End while
Output: A, E
```

IV. Experimental Results

In this section, we first test the influence of the parameters on the recovery quality (Section IV-A) and then evaluate the proposed method on the public CMU dataset [12], Edinburgh dataset [13] [14] (Section IV-B) and two real datasets [15] captured by Kinect v2.0 (Section IV-C). Both quantitative and

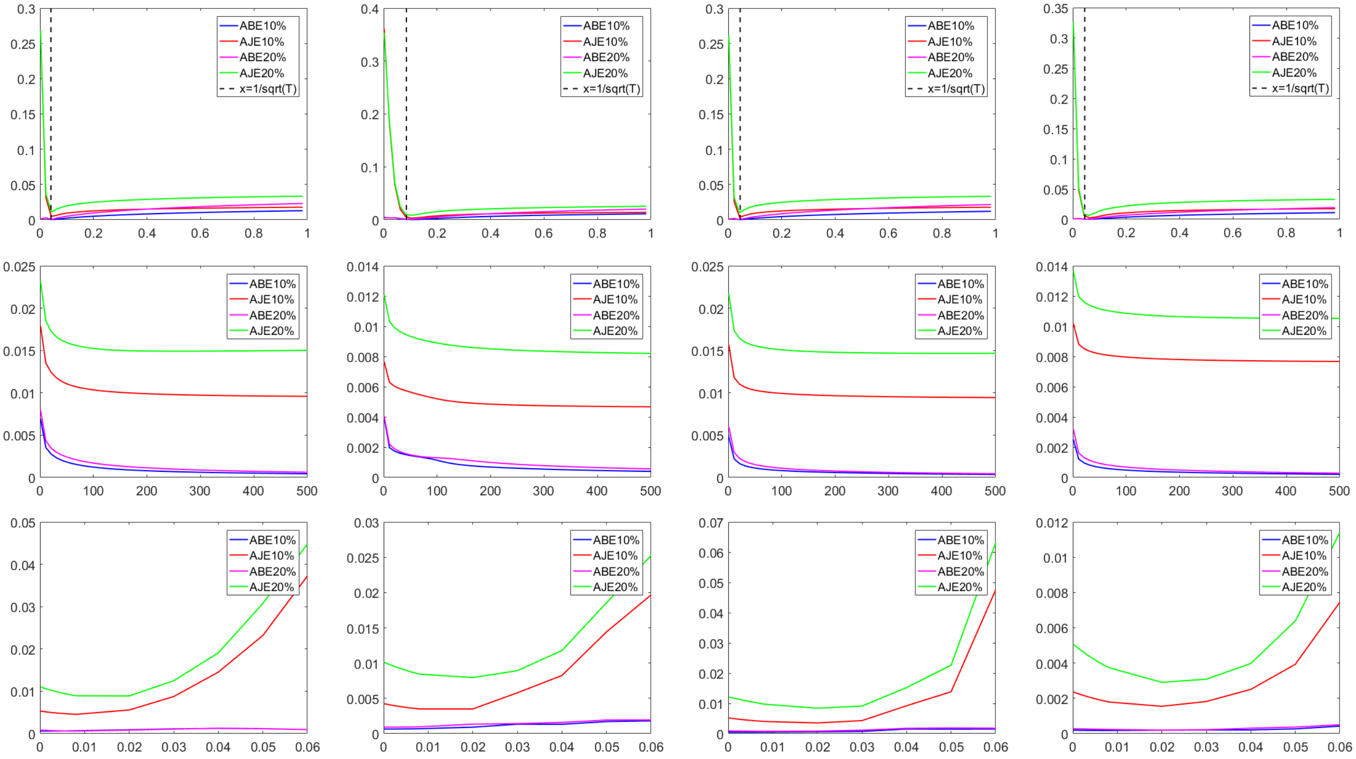


Fig. 3. The average bone error (ABE) and the average joint error (AJE) (y-axis) comparison on four randomly selected sequences (from left to right) of the CMU dataset [12] for 10% and 20% corrupted cases with respect to the parameters of λ (Top), γ (Middle), and μ (Bottom) (x-axis).

qualitative results are presented. For quantitative evaluation, the metrics of recovery error, known as Average Joint Error (AJE) and Average Bone-length Error (ABE), are calculated as: $\omega = \frac{1}{ST}\sum_{t}\sum_{p} d\left(\tilde{\mathbf{n}}_p^t, \mathbf{n}_p^t\right)$ and $\xi = \frac{1}{T(S-1)}\sum_{t}\sum_{e_{ij}\in\mathcal{E}}\left|\tilde{l}_{ij}^t - l_{ij}^t\right|$, respectively. $\tilde{\mathbf{n}}_p^t$ and $\mathbf{n}_p^t$ are the ground truth and reconstructed joint positions. $\tilde{l}_{ij}^t$ and l_{ij}^t are the ground truth and reconstructed bone lengths of the i-th bone at the t-th frame. ω and ξ represent the average absolute difference over joints and bones in all the frames, respectively. Finally, the running times of all the methods are reported in Section IV-D.

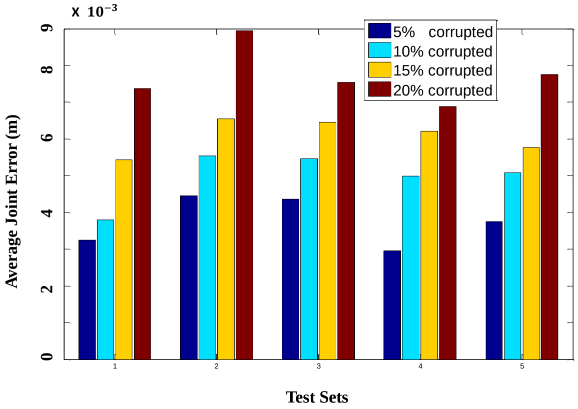


Fig. 4. Average joint errors in meters for different sequences of subject 09 using our method for motion recovery.

A. Parameter Sensitivity Experiments

In order to demonstrate the generality of our chosen values of parameters, we test the influence of the parameters on the recovery quality by randomly selecting four sequences from the CMU dataset [12]. We evaluate the AJE and ABE by tuning each parameter over the interesting part of the parameter space while setting other parameters at the fixed reasonable values: $\lambda = 1/\sqrt{T}$, $\gamma = 100$, $\mu = 0.01$.

1) λ: This parameter adjusts the importance of the data term and the sparse term. The error matrix $\mathbf{E}$ should be sparse because of occlusion in the real world. According to the principal component analysis [44], we set $\lambda = 1/\sqrt{T}$ in our implementation, where T is the number of frames. The dashed line in the top row of Fig. 3 shows that our chosen value consistently gives the minimum function error.

2) γ: This parameter promotes isometry of bones on the skeleton by exploiting spatial coherence of skeletons. As this term is typically small due to the fourth order of error, we choose $\gamma = 100$ for balancing the importance of the data term and the length preservation term. The middle row of Fig. 3 shows that the error curve gradually declines and then tends to be the same after $\gamma = 100$. Therefore, we set $\gamma = 100$ in our experiments.

3) μ: The human motion can be modeled by wavelet transform owing to physical structure characteristics. Given that the human movement is smooth with singularities, we add sparse prior to the wavelet term. The strength of wavelet approximation is controlled by the weight μ associated with the smoothness term. μ should be large enough to emphasize the sparseness of the smoothness and small enough to provide sufficient flexibility in formulating the wavelet term. The bottom row of Fig. 3 shows the error curves, and so we set $\mu = 0.01$ for all our experiments which consistently gives good results.

4) Other relevant parameter settings: There is no requirement for the sequence length and the number of joints in our algorithm, as long as computer memory permits. Considering the time consumption and computational precision, it is better to take 50 to 300 frames. In the experiment, we set $\rho_1 = \rho_2 = \rho_3 = 0.5$, $\eta_1 = \eta_2 = \eta_3 = 1.3$.

Based on the results in Fig. 3 and the associated analysis, we found that the performance of our model was stable with respective to various sequences and parameters. Therefore, we use a set of fixed parameters in the rest of the experiments (in Section IV-B and Section IV-C) instead of tuning parameters for each sequence.

TABLE II
AVERAGE JOINT ERROR COMPARISON FOR SKELETON RECOVERY ON THE CMU DATASET [12] WITH DIFFERENT CORRUPTION PERCENTAGES (M).

Sub.	Method	5%	10%	15%	20%	80%
05	Wang [16]	0.066	0.070	0.073	0.075	0.093
	Li [17]	**0.002**	**0.005**	0.008	0.011	0.074
	Ours	**0.002**	**0.005**	**0.007**	**0.009**	**0.064**
09	Wang [16]	0.076	0.079	0.083	0.085	0.122
	Li [17]	0.005	0.007	0.010	0.014	0.073
	Ours	**0.004**	**0.005**	**0.006**	**0.008**	**0.063**
13	Wang [16]	0.073	0.076	0.078	0.085	0.105
	Li [17]	0.008	0.011	**0.012**	**0.016**	**0.072**
	Ours	**0.006**	**0.010**	**0.012**	0.017	0.073
24	Wang [16]	0.069	0.075	0.078	0.082	0.114
	Li [17]	0.004	0.007	**0.008**	0.013	0.075
	Ours	**0.003**	**0.005**	**0.008**	**0.012**	**0.070**
56	Wang [16]	0.068	0.073	0.077	0.080	0.118
	Li [17]	**0.003**	0.006	0.009	0.013	0.074
	Ours	**0.003**	**0.005**	**0.008**	**0.010**	**0.068**
86	Wang [16]	0.072	0.075	0.077	0.083	0.121
	Li [17]	**0.003**	**0.006**	**0.008**	0.012	0.076
	Ours	**0.003**	**0.006**	0.009	**0.011**	**0.066**
93	Wang [16]	0.068	0.071	0.075	0.081	0.098
	Li [17]	0.003	0.008	0.010	0.011	0.081
	Ours	**0.001**	**0.005**	**0.008**	**0.010**	**0.069**
115	Wang [16]	0.072	0.076	0.078	0.085	0.130
	Li [17]	0.007	0.005	0.012	**0.015**	0.062
	Ours	**0.006**	**0.004**	**0.009**	**0.015**	**0.052**
140	Wang [16]	0.061	0.063	0.070	0.080	0.108
	Li [17]	**0.003**	**0.005**	0.010	0.012	0.053
	Ours	0.004	0.006	**0.009**	**0.011**	**0.049**
Total	Wang [16]	0.068	0.073	0.079	0.081	0.104
	Li [17]	0.005	0.007	0.009	0.012	0.073
	Ours	**0.004**	**0.006**	**0.007**	**0.010**	**0.067**

B. Results on Public Datasets

In this section, we evaluate the performance of our method on the CMU dataset [12] and the Edinburgh dataset [13] [14] in terms of accuracy and smoothness. Each skeleton in the CMU dataset [12] contains 25 skeletal joints (with finger joints removed) and 24 bones (as demonstrated in Fig. 1(a)), while the skeleton in the Edinburgh dataset and Kinect dataset has 21 joints (also without finger joints) and 20 bones. The CMU dataset captures human motion by placing makers on every subject and recording the markers' positions, and thus can be used as ground truth for evaluation. We simulate corruptions in the skeleton data. Specifically, random noise is added to a fraction of entities in the ground-truth skeleton matrix $\tilde{\mathbf{A}}$, obtaining the observed skeleton matrix $\mathbf{D}$. The noise in the polluted joints is uniformly distributed in the range of [-25 25] *cm* in each spatial dimension. This range is selected according to the average length of arms and legs because noisy joints are unlikely to go beyond this range. The length of bones is computed according to the given skeleton. As for the skeleton captured without ground truth, the bone length is estimated by the average of the bone lengths over the less-corrupted sequence. Recovery errors in terms of AJE are presented in Fig. 4. Four different percentages of polluted entities in $\mathbf{D}$, *i.e.*, 5%, 10%, 15%, and 20% are tested, which are similar to real captured situation. As shown in Fig. 4, our method can achieve consistent recovery errors for different motion sequences of the same subject with respect to the same proportion of corruption.

We also measure the average bone-length error (ABE) in centimeters in Fig. 5, compared with our previous work [16], [17]. It can be observed that our method and the method in [17] recover the skeletons with more accurate bone lengths due to the isometry constraint than the method in [16]. Our method has the smallest error due to the sparse constraint with wavelet transform which guarantees the smoothness of the recovered trajectory and the robustness of the method. Fig. 6 shows the trajectories (3D positions over time) of the root joint recovered by different methods. It can be seen that our method can recover a more smooth and stable joint motion trajectory thanks to the proposed temporal regularization. In a word, our method recovers the corrupted skeleton sequences with high accuracy and reasonable smoothness by exploiting the temporal and spatial correlations of the skeleton matrix.

For detailed comparison, we test the recovery performance of several skeleton sequences from different subjects with various motions and temporal durations. Specifically, we use sequences from subject 05 to subject 140, including a variety of actions such as running, bending, kicking, dancing, *etc*. Table II gives quantitative evaluation (average joint errors in meters) for different subjects. One extreme case with 80% corrupted elements is included. It can be seen that our method and the method in [17] obviously outperform the method in [16] due to the use of isometry constraint. Thanks to the proposed sparse constraint that guarantees the plausibility of human motions to ensure smooth recovery of motion sequence, our method achieves the most accurate recovery result for most cases. Even for the case with 80% corrupted elements, our method can still reconstruct reasonable motion within 0.07 *m* in terms of AJE. In five cases, the errors of the method in [17] are smaller than ours by 0.001 because the proposed sparse constraint improves the smoothness at the expense of slight drop of precision especially for complex

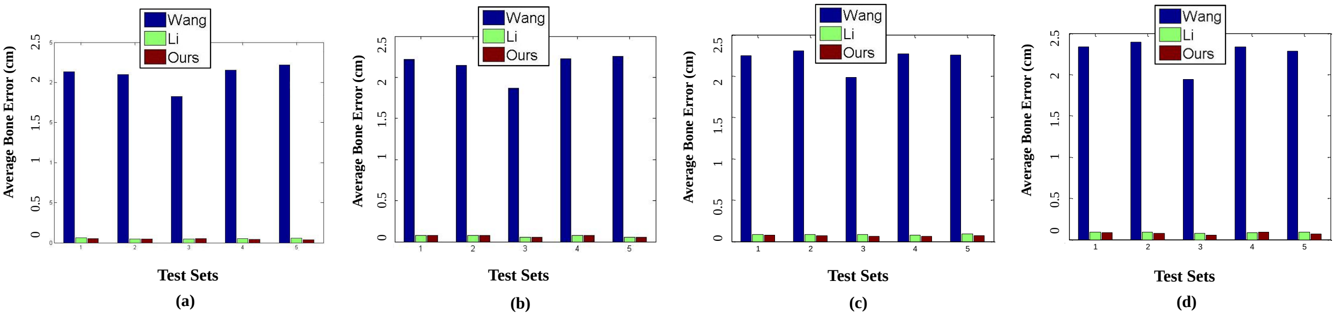


Fig. 5. Average bone-length errors in centimeters for different sequences of subject 09 recovered by Wang [16], Li [17] and our method. Different corrupted percentages are compared: (a) 5%, (b) 10%, (c) 15%, and (d) 20%.

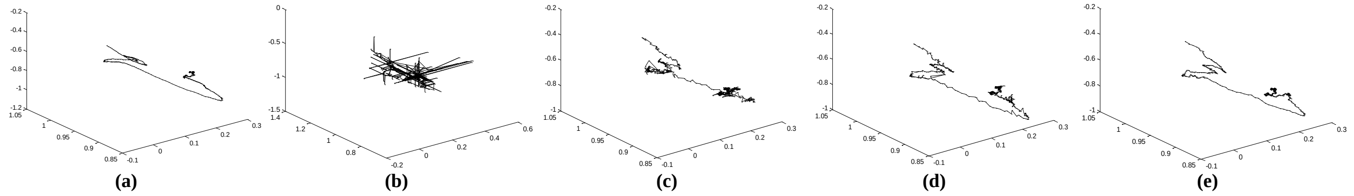


Fig. 6. Comparison results of trajectories (3D positions over time) of joint No.1 (the root joint) of subject 05 sequence 02: (a) ground truth, (b) 20% damaged trajectory, (c) recovered trajectory by [16], (d) recovered trajectory by [17], and (e) recovered trajectory by our method.

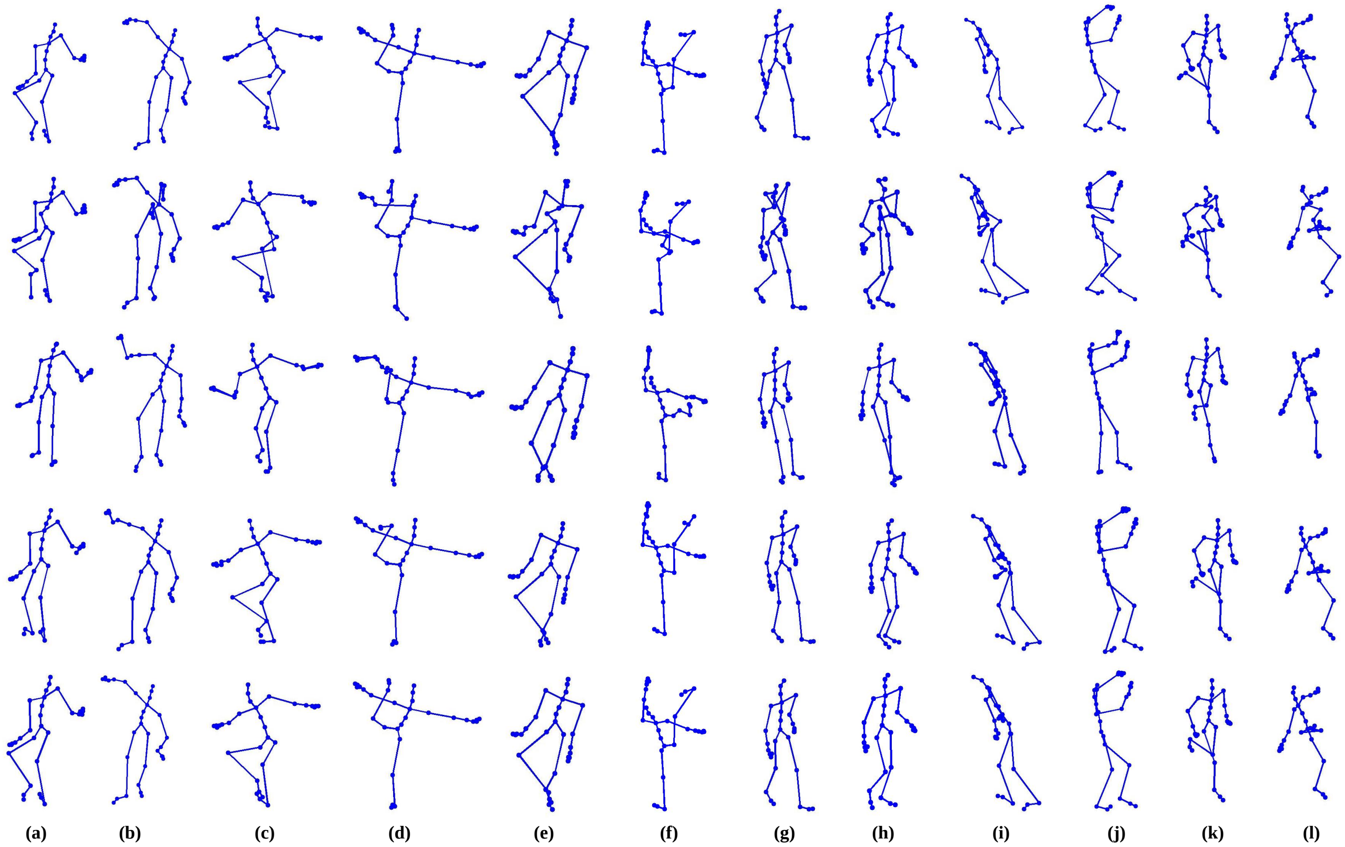


Fig. 7. Comparison results for different sequences of different subjects in the CMU dataset [12]. From top to bottom: ground-truth skeletons, corrupted skeletons, skeletons recovered by [16], [17], and our method. The corruption rates are: (a)-(c) 5%, (d)-(f) 10%, (g)-(i) 15%, and (j)-(l) 20%.

motions. Fig. 7 gives 12 examples for different subjects with different motions and different corruptions, compared with two methods. The method in [16] recovers reasonable skeleton motions by sacrificing some motion details, but the recovered

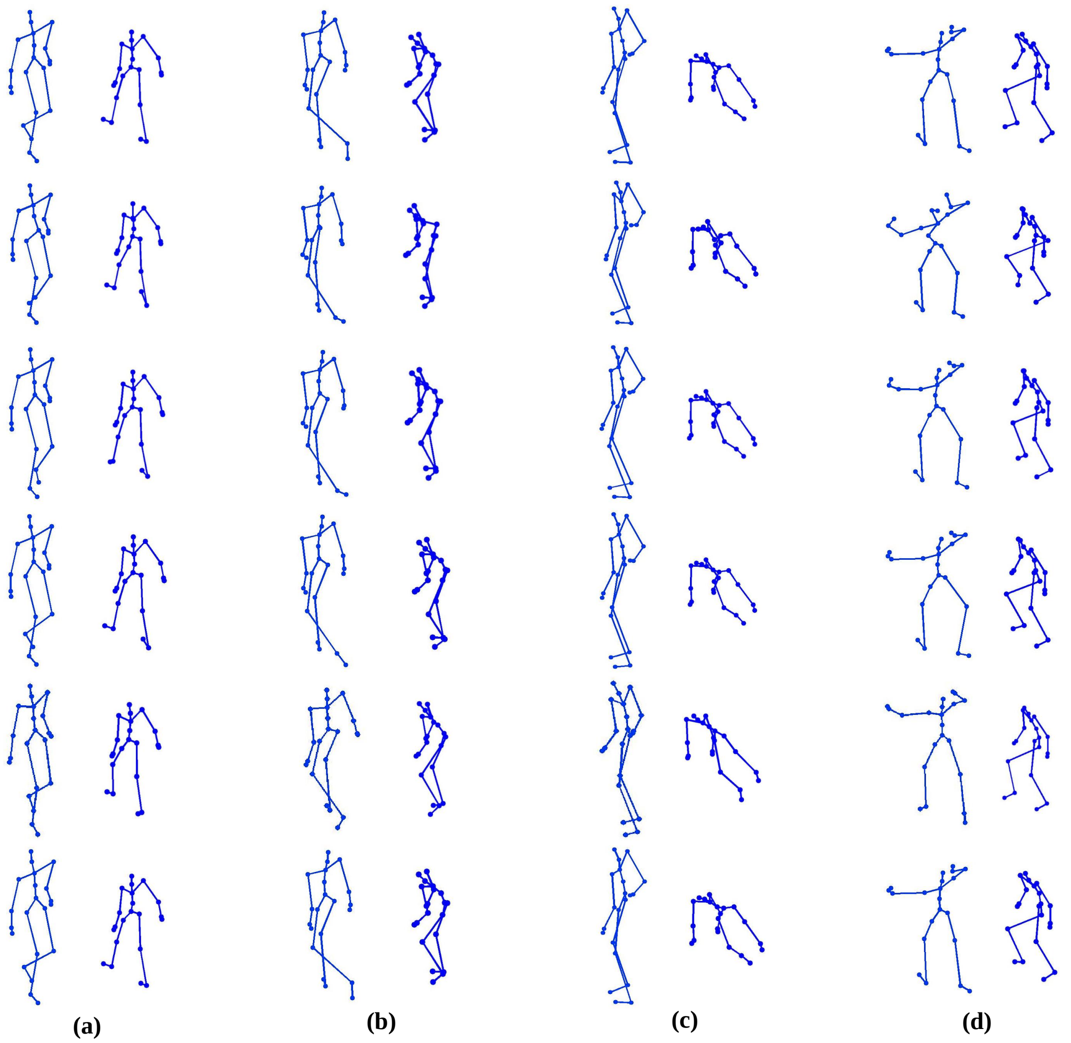


Fig. 8. Comparison results for different sequences of different subjects in the CMU dataset [12] (left in each subfigure) and the Edinburgh dataset [13] [14] (right in each subfigure). From top to bottom: ground-truth skeletons, corrupted skeletons, skeletons recovered by [16], [17], [14], and our method. The corruption rates are: (a) 5%, (b) 10%, (c) 15%, and (d) 20%.

motions are too rigid compared with ground truth. The method in [17] recovers more accurate skeleton motions thanks to the isometry constraint, but the recovered motions still suffer from jitter artifacts. Our method is able to recover accurate motion with rich details thanks to the elegant design of spatio-temporal constraints.

We also compare with two popular deep learning methods: method [14] in Table III on the CMU dataset [12] and in Table IV on the Edinburgh dataset [13] [14], and method [50] in Table V on the CMU dataset [12]. The visual comparison is presented in Fig. 8 and Fig. 9. Because the method in [14] contains many pre-processing steps and handles the BVH format data, we compare with this method in a separate table and figure. The noise in the polluted joints is uniformly distributed in the range of [-2.5, 2.5] inches and five different percentages of polluted entities in $\mathbf{D}$ are tested: 5%, 10%, 15%, 20%, and 80%. Consistent with the method in [14], 21 joints are used. As shown in Table III and Table IV, the method in [14] has the lowest accuracy. This is mainly due to the pre-processing for deep learning including scaling to a unified skeleton structure, removing global translation around the xz plane and global rotation around the y-axis, and limiting one foot on the floor, which ensures the smoothness of the movement but the accuracy is lost. Therefore, the recovered skeleton sequences by this method look visually good and smooth without jitter during the time, but the accuracy of the joints is not very high. Fig. 8 shows the qualitative comparison of the four methods. Due to the constraint on the foot and the removal of global rotation around the y axis, the method in [14] has a serious deviation at the shoulders and the feet. On the contrary, our method achieves the most accurate recovery without any pre-processing. Moreover, our method can deal with data of arbitrary format including original global coordinates. For method [50], we set the number of joints to 25 and the other parameters to the best provided by the author. Since the LSTM (long short term memory)-based model is better than the time-window-based model, we only compare with the former. Following the author, we randomly selected

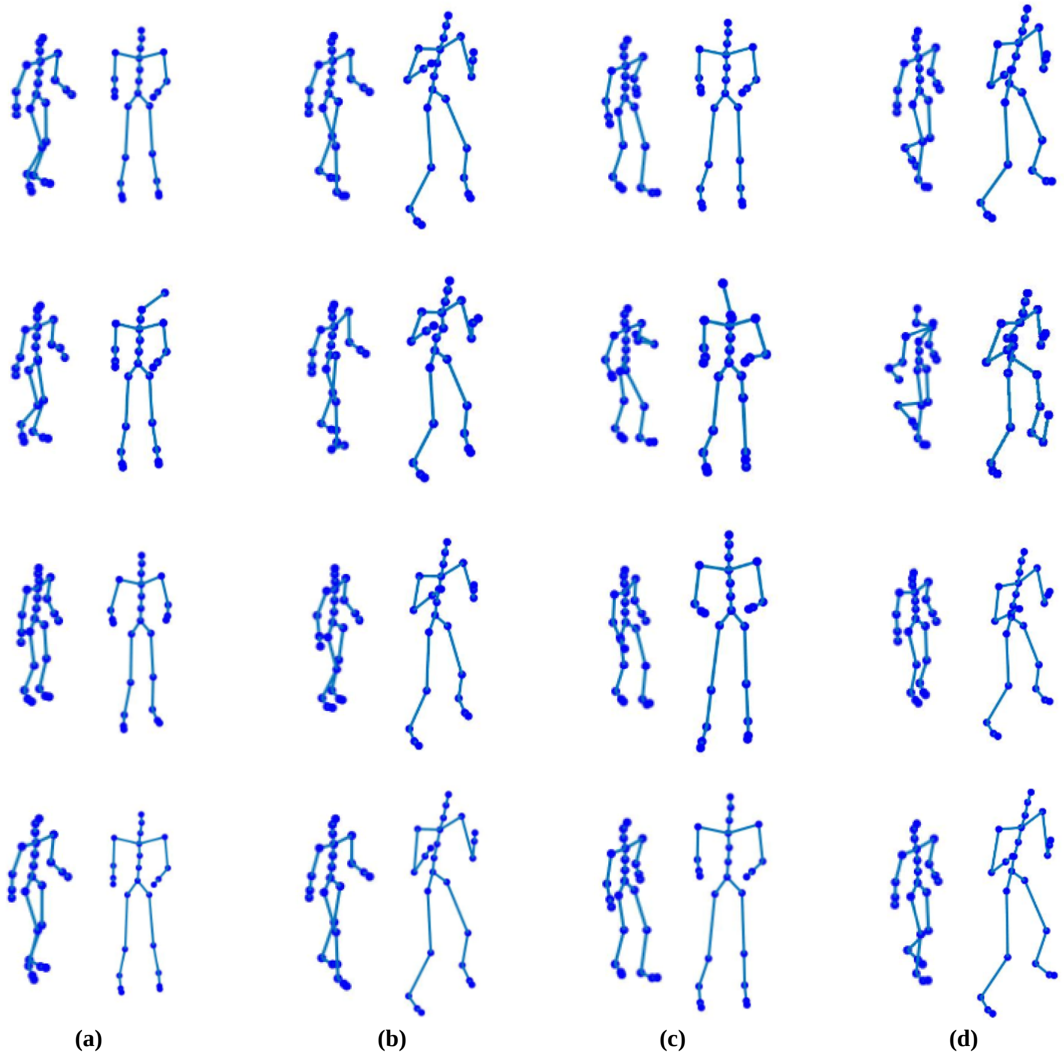


Fig. 9. Comparison results for sequence 01 of subject 05 (left in each subfigure) and sequence 01 of subject 14 (right in each subfigure) in the CMU dataset [12] . From top to bottom: ground-truth skeletons, corrupted skeletons, skeletons recovered by [50], and our method. The corruption rates are: (a) 5%, (b) 10%, (c) 15%, and (d) 20%.

25 different folders from the CMU dataset [12] for training, testing and validation. We use sequence 14_01 and sequence 05_01 as test sequences, in which subject 14 is in the training set, while subject 05 is not in the training set. The input of the network in [50] requires a mask to indicate the missing joint points. Because the simulated noise data is generated randomly, we set the mask to all 1 for our input. The whole noise matrix is input and the output of the network is taken as the final result. As shown in Table V, method [50] has the lowest accuracy, especially for sequence 05_01, due to the insufficient generalization ability of the model and the need for precise mask in denoising. Fig. 9 shows the qualitative comparison of [50] and ours. The whole network structure in [50] tends to recover human structure without considering accuracy, especially in the case of subjects not included in the training set. Moreover, method [50] requires retraining data for different skeleton structure, which cannot be restored for occluded Kinect data. However, our method does not have such requirements and only need tens of frames to recover more accurate motion.

C. Results on Kinect Data

The motions captured by most motion capture devices such as the Kinect often suffer from severe joint drifting and motion jitter, especially for occlusion. To validate the performance of our method in practical applications, we experiment on two real captured Kinect datasets collected by ourselves and [15]. The actor starts at a non-occluded pose, and the motion is very slight in the first few frames. Therefore, we choose the average length for each bone in the first several frames as the reference bone length l_{ij} in Eq. (5).

Fig. 10 shows the comparison results for two frames of two Kinect datasets. It can be seen that the method in [16] reconstructs reasonable motions from the corrupted skeletons, but the recovered skeletons lose many motion details, and the motion looks rigid and unnatural. The method in [17] and our method recover accurate motion with certain motion details preserved. We also measure the 3D trajectory of the

TABLE III
AVERAGE JOINT ERROR COMPARISON FOR SKELETON RECOVERY ON CMU DATASET [12] (BVH FORMAT) WITH DIFFERENT CORRUPTION PERCENTAGES.

Sub.	Method	5%	10%	15%	20%	80%
05_01	Wang [16]	0.058	0.125	0.185	0.251	0.904
	Li [17]	0.030	0.077	0.117	0.170	0.743
	Holden [14]	0.637	0.672	0.753	0.856	1.636
	Ours	**0.013**	**0.033**	**0.057**	**0.085**	**0.564**
05_14	Wang [16]	0.077	0.156	0.229	0.312	1.028
	Li [17]	0.045	0.097	0.154	0.215	0.858
	Holden [14]	0.952	1.012	1.088	1.141	1.867
	Ours	**0.017**	**0.031**	**0.055**	**0.095**	**0.582**
13_29	Wang [16]	0.044	0.094	0.145	0.190	0.733
	Li [17]	0.021	0.051	0.088	0.128	0.707
	Holden [14]	0.923	0.976	1.037	1.113	1.880
	Ours	**0.010**	**0.024**	**0.045**	**0.071**	**0.527**
13_38	Wang [16]	0.047	0.094	0.140	0.185	0.700
	Li [17]	0.027	0.057	0.092	0.133	0.688
	Holden [14]	0.919	1.015	1.040	1.104	1.711
	Ours	**0.012**	**0.031**	**0.061**	**0.095**	**0.647**

TABLE IV
AVERAGE JOINT ERROR COMPARISON FOR SKELETON RECOVERY ON THE EDINBURGH DATASET [13] [14] WITH DIFFERENT CORRUPTION PERCENTAGES.

Sub.	Method	5%	10%	15%	20%	80%
07	Wang [16]	0.126	0.181	0.240	0.308	1.160
	Li [17]	0.063	0.108	0.161	0.219	0.884
	Holden [14]	0.905	0.996	1.016	1.093	1.896
	Ours	**0.013**	**0.030**	**0.058**	**0.091**	**0.679**
08	Wang [16]	0.112	0.166	0.225	0.292	1.160
	Li [17]	0.058	0.103	0.152	0.210	0.894
	Holden [14]	1.253	1.410	1.498	1.597	2.242
	Ours	**0.011**	**0.029**	**0.052**	**0.082**	**0.657**
09	Wang [16]	0.134	0.193	0.252	0.323	1.178
	Li [17]	0.062	0.113	0.163	0.227	0.907
	Holden [14]	0.948	1.029	1.097	1.183	1.919
	Ours	**0.011**	**0.031**	**0.053**	**0.085**	**0.651**
10	Wang [16]	0.121	0.177	0.237	0.300	1.162
	Li [17]	0.061	0.110	0.162	0.217	0.886
	Holden [14]	0.740	0.842	0.904	1.007	1.809
	Ours	**0.011**	**0.028**	**0.054**	**0.080**	**0.645**

TABLE V
AVERAGE JOINT ERROR COMPARISON FOR SKELETON RECOVERY ON CMU DATASET [12] WITH DIFFERENT CORRUPTION PERCENTAGES.

Sub.	Method	5%	10%	15%	20%	80%
05_01	Kucherenko [50]	0.329	0.330	0.326	0.325	0.331
	Ours	**0.002**	**0.005**	**0.007**	**0.009**	**0.044**
14_01	Kucherenko [50]	0.076	0.076	0.074	0.074	0.075
	Ours	**0.005**	**0.006**	**0.007**	**0.009**	**0.039**

captured motion in Fig. 11. For better visualization, the regions highlighted by rectangles are enlarged and shown aside. As shown in Fig. 11(a), jitter artifacts often happen in human motion captured by Kinect, *e.g.*, highlighted points in the figure, where the Kinect device suddenly loses the location of dynamic human body. The method in [16] and the method in [17] both filter out some obvious outliers, but fail to smooth the whole trajectory. The method in [16] would even cause error during the recovery procedure. On the contrary, our method recovers accurate and smooth motion thanks to the sparse wavelet constraint. Hence, in practical cases which have complex human motions and are lack of ground-truth bone length to ensure isometry, our method can still recover reasonable and smooth motion.

D. Running Times

The running times for the CMU dataset [12] are given in Table VI. All the experiments are performed on a desktop computer with an Intel i5-4690K 3.5GHz CPU and 8GB RAM. Four skeleton sequences with an increasing frame length are tested. In order to compare with the deep learning method [14], we divide each sequence into 240 overlapping windows according to the method in [14] and calculate the average recovery time for all the methods. The running times of the method in [16], the method in [17], the method in [14] and our method are $0.5764s$, $16.4939s$, $34.4910s$, and $19.7044s$, respectively.

TABLE VI
THE RUNNING TIMES ON THE CMU DATASET.

Sequences	Frame Length	Methods	Running Time(s)
Sub.21 Seq.03	272	Wang [16]	1.0983
		Li [17]	22.7150
		Ours	24.7055
Sub.115 Seq.05	584	Wang [16]	1.4027
		Li [17]	47.5670
		Ours	60.7618
Sub.140 Seq.04	1100	Wang [16]	4.4831
		Li [17]	96.0281
		Ours	148.7628
Sub.56 Seq.06	6784	Wang [16]	23.9200
		Li [17]	478.3740
		Ours	586.2765

V. CONCLUSIONS

This paper proposes a novel skeleton recovery method using spatio-temporal reconstruction. The corrupted skeleton sequence is integrated into a skeleton matrix, and we use a joint low-rank and sparse prior to exploit temporal correlation and an isometric constraint for spatial correlation. The whole model is solved under an iterative ALM framework, and a Gauss-Newton solver is introduced to solve the nonlinear least squares subproblem. Experimental results on the public CMU dataset, the Edinburgh dataset and two real captured Kinect datasets demonstrate the accuracy and robustness of the proposed method compared with state-of-the-art methods. Our method can be used to pre-process a large amount of damaged skeletons to improve the accuracy of downstream applications.

Our method also has some limitations to be overcome in future work: 1) It is not very effective for the cases with loss or

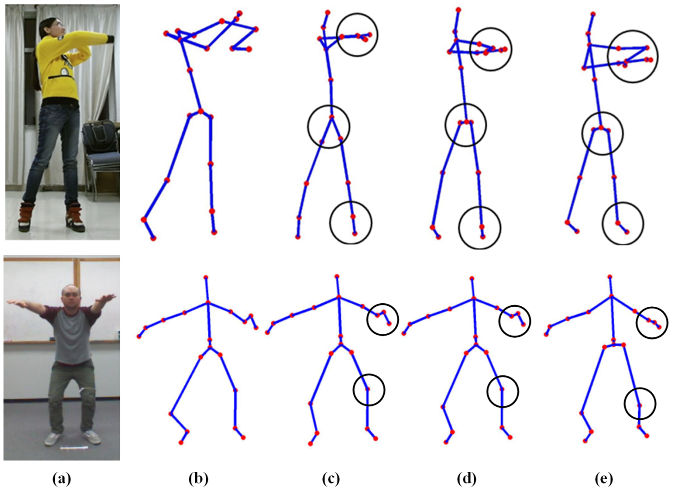


Fig. 10. Comparison results for our Kinect sequence (top) and another Kinect sequence [15] (bottom). (a) captured color image, (b) captured Kinect skeleton, (c) recovered skeleton by [16], (d) recovered skeleton by [17], and (e) recovered skeleton by our method.

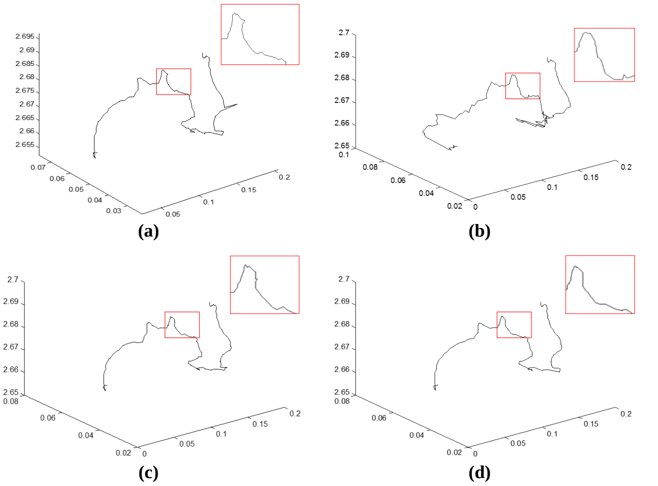


Fig. 11. Comparison results of trajectories of the root joint of a Kinect sequence: (a) motion captured by Kinect, (b) recovered trajectory by [16], (c) recovered trajectory by [17] and (d) recovered trajectory by our method.

damage of multiple continuous frames, and the computational complexity rapidly increases with the increase of matrix size. 2) Besides the isometry property, we also note that more structure information can be considered, *e.g.*, the relative position of the skeleton joints. However, this would make the model more difficult to optimized.

Acknowledgement

The authors are grateful to Dr. D. Holden and Dr. T. Kucherenko for their code and helpful discussions, and thank the Associate Editor and anonymous reviewers for their comments which significantly helped to improve this paper.

References

[1] S. Liu, Y. Li, and G. Hua, "Human pose estimation in video via structured space learning and halfway temporal evaluation," *IEEE TCSVT*, 2018.

[2] Y. Wang, Y. Liu, X. Tong, Q. Dai, and P. Tan, "Outdoor markerless motion capture with sparse handheld video cameras," *IEEE TVCG*, 2017.

[3] Y. Li, Q. Miao, K. Tian, Y. Fan, X. Xu, R. Li, and J. Song, "Large-scale gesture recognition with a fusion of RGB-D data based on saliency theory and C3D model," *IEEE TCSVT*, vol. 28, no. 10, pp. 2956–2964, 2017.

[4] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3D skeletons as points in a lie group," in *IEEE CVPR*, 2014, pp. 588–595.

[5] L. Chen, J. Lu, Z. Song, and J. Zhou, "Part-activated deep reinforcement learning for action prediction," in *ECCV*, 2018, pp. 421–436.

[6] L.-Y. Gui, Y.-X. Wang, D. Ramanan, and J. M. Moura, "Few-shot human motion prediction via meta-learning," in *ECCV*, 2018.

[7] T. Yu, K. Guo, F. Xu, Y. Dong, Z. Su, J. Zhao, J. Li, Q. Dai, and Y. Liu, "Bodyfusion: Real-time capture of human motion and surface geometry using a single depth camera," in *IEEE ICCV*. IEEE, October 2017.

[8] H. Zhu, Y. Liu, J. Fan, Q. Dai, and X. Cao, "Video-based outdoor human reconstruction," *IEEE TCSVT*, vol. 27, no. 4, pp. 760–770, 2017.

[9] B. Wang, Z. Chen, and J. Chen, "Gesture recognition by using Kinect skeleton tracking system," in *Int. Con. Intelligent Human-Machine Systems and Cybernetics*, 2013, pp. 1119–1119.

[10] D. S. Alexiadis and P. Daras, "Quaternionic signal processing techniques for automatic evaluation of dance performances from MoCap data," *IEEE TMM*, vol. 16, no. 5, pp. 1391–1406, 2014.

[11] S. Obdrzalek, G. Kurillo, F. Ofli, R. Bajcsy, E. Seto, H. Jimison, and M. Pavel, "Accuracy and robustness of Kinect pose estimation in the context of coaching of elderly population," in *International Conference of the IEEE Engineering in Medicine and Biology Society*, 2012, pp. 1188–1193.

[12] CMU, "Carne1gie-mellon university mocap database," *http://mocap.cs.cmu.edu/*.

[13] D. Holden, J. Saito, T. Komura, and T. Joyce, "Learning motion manifolds with convolutional autoencoders," in *SIGGRAPH Asia 2015 Technical Briefs*. ACM, 2015, p. 18.

[14] D. Holden, J. Saito, and T. Komura, "A deep learning framework for character motion synthesis and editing," *ACM ToG*, vol. 35, no. 4, p. 138, 2016.

[15] C. Chen, "Utd-mhad: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor," in *IEEE ICIP*, 2015.

[16] M. Wang, K. Li, F. Wu, Y.-K. Lai, and J. Yang, "3-D motion recovery via low rank matrix analysis," in *IEEE VCIP*, 2016, pp. 1–4.

[17] K. Li, M. Wang, Y.-K. Lai, J. Yang, and F. Wu, "3-D motion recovery via low rank matrix restoration on articulation graphs," in *IEEE ICME*, 2017, pp. 721–726.

[18] C. Menier, E. Boyer, and B. Raffin, "3D skeleton-based body pose recovery," in *Int. Sym. 3D DPVT*, 2006, pp. 389–396.

[19] H. S. Park and Y. Sheikh, "3D reconstruction of a smooth articulated trajectory from a monocular image sequence," in *ICCV*, 2011, pp. 201–208.

[20] J. Valmadre, Y. Zhu, S. Sridharan, and S. Lucey, "Efficient articulated trajectory reconstruction using dynamic programming and filters," *ECCV*, pp. 72–85, 2012.

[21] S. Leonardos, X. Zhou, and K. Daniilidis, "Articulated motion estimation from a monocular image sequence using spherical tangent bundles," in *IEEE ICRA*, 2016.

[22] X. Yang, A. Somasekharan, and J. J. Zhang, "Curve skeleton skinning for human and creature characters," *Computer Animation and Virtual Worlds*, vol. 17, no. 3-4, pp. 281–292, 2006.

[23] W. Cheng, R. Cheng, X. Lei, and S. Dai, "Automatic skeleton generation and character skinning," in *IEEE Int. Sym. VR Innovation*, 2011, pp. 299–304.

[24] I. Baran and J. Popović, "Automatic rigging and animation of 3D characters," *ACM ToG*, vol. 26, no. 3, p. 72, 2007.

[25] Y. Feng, J. Xiao, Y. Zhuang, X. Yang, J. J. Zhang, and S. Rong, "Exploiting temporal stability and low-rank structure for motion capture data refinement," *Information Sciences*, vol. 277, no. 2, pp. 777–793, 2014.

[26] W. Hu, Z. Wang, S. Liu, X. Yang, G. Yu, and J. J. Zhang, "Motion capture data completion via truncated nuclear norm regularization," *IEEE SPL*, vol. 25, no. 2, pp. 258–262, 2018.

[27] A. Toshev and C. Szegedy, "DeepPose: Human pose estimation via deep neural networks," in *IEEE CVPR*, 2013, pp. 1653–1660.

[28] W. Ouyang, X. Chu, and X. Wang, "Multi-source deep learning for human pose estimation," in *IEEE CVPR*, 2014, pp. 2337–2344.

[29] Y. Yang and D. Ramanan, "Articulated pose estimation with flexible mixtures-of-parts," in *IEEE CVPR*, 2011, pp. 1385–1392.

[30] F. Wang and Y. Li, “Beyond physical connections: Tree models in human pose estimation,” in *IEEE CVPR*, 2013, pp. 596–603.

[31] Y. Tian, C. L. Zitnick, and S. G. Narasimhan, “Exploring the spatial hierarchy of mixture models for human pose estimation,” in *ECCV*, 2012, pp. 256–269.

[32] J. Lei, C. Zhang, Y. Fang, Z. Gu, N. Ling, and C. Hou, “Depth sensation enhancement for multiple virtual view rendering,” *IEEE TMM*, vol. 17, no. 4, pp. 457–469, 2015.

[33] J. Lei, J. Liu, H. Zhang, Z. Gu, N. Ling, and C. Hou, “Motion and structure information based adaptive weighted depth video estimation,” *IEEE TBroadcast.*, vol. 61, no. 3, pp. 416–424, 2015.

[34] X. Wei, P. Zhang, and J. Chai, “Accurate realtime full-body motion capture using a single depth camera,” *ACM ToG*, vol. 31, no. 6, pp. 439–445, 2012.

[35] J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, and A. Kipman, “Efficient human pose estimation from single depth images,” *IEEE TPAMI*, vol. 35, no. 12, pp. 2821–2840, 2013.

[36] J. Chai and J. K. Hodgins, “Performance animation from low-dimensional control signals,” *ACM ToG*, vol. 24, no. 3, pp. 686–696, 2005.

[37] A. Aristidou, D. Cohen-Or, J. K. Hodgins, and A. Shamir, “Self-similarity analysis for motion capture cleaning,” in *Computer Graphics Forum*, vol. 37, no. 2, 2018, pp. 297–309.

[38] J. Saito, D. Holden, and T. Komura, “Learning motion manifolds with convolutional autoencoders,” in *SIGGRAPH Asia Technical Briefs*, 2015.

[39] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, “The convex geometry of linear inverse problems,” *Foundations of Computational Mathematics*, vol. 12, no. 6, pp. 805–849, 2012.

[40] B. Sturm, *A Wavelet Tour of Signal Processing*. Academic Press, 1999.

[41] C. Torrence and G. P. Compo, “A practical guide to wavelet analysis,” *Bulletin of the American Meteorological Society*, vol. 79, no. 79, pp. 61–78, 1998.

[42] Y. Xu, X. Yang, H. Ling, and H. Ji, “A new texture descriptor using multifractal analysis in multi-orientation wavelet pyramid,” in *IEEE CVPR*, 2010, pp. 161–168.

[43] E. J. Candès, X. Li, Y. Ma, and J. Wright, “Robust principal component analysis,” *Journal of the ACM*, vol. 1, no. 1, pp. 1–73, 2000.

[44] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma, “Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization,” in *NeurIPS*, 2009, pp. 2080–2088.

[45] Z. Lin, M. Chen, and Y. Ma, “The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices,” *arXiv preprint arXiv:1009.5055*, 2010.

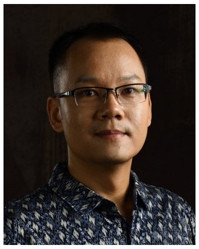
[46] C. Chen, Y. Shen, and Y. You, “On the convergence analysis of the alternating direction method of multipliers with three blocks,” in *Abstract and Applied Analysis*, 2013.

[47] M. Zollhöfer, J. Thies, M. Colaianni, M. Stamminger, and G. Greiner, “Interactive model-based reconstruction of the human head using an RGB-D sensor,” *Computer Animation and Virtual Worlds*, vol. 25, no. 3-4, pp. 213–222, 2014.

[48] W. Daniel, B. Jan, S. Markus, S. Andr, and F. Dieter, “Efficient GPU data structures and methods to solve sparse linear systems in dynamics applications,” in *Computer Graphics Forum*, 2013, pp. 16–26.

[49] Y. Xu, W. Yin, Z. Wen, and Y. Zhang, “An alternating direction algorithm for matrix completion with nonnegative factors,” *Frontiers of Mathematics in China*, vol. 7, no. 2, pp. 365–384, 2012.

[50] T. Kucherenko and H. Kjellström, “A neural network approach to missing marker reconstruction,” *arXiv preprint arXiv:1803.02665*, 2018.



Jingyu Yang (M10-SM17) received the B.E. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2003, and Ph.D. (Hons.) degree from Tsinghua University, Beijing, in 2009. He has been a Faculty Member with Tianjin University, China, since 2009, where he is currently a Professor with the School of Electrical and Information Engineering. He was with Microsoft Research Asia (MSRA) in 2011, and the Signal Processing Laboratory, EPFL, Lausanne, Switzerland, in 2012, and from 2014 to 2015. His research interests include image/video processing, 3D imaging, and computer vision. He has authored or co-authored over 80 high quality research papers. As a co-author, he got the best 10% paper award in IEEE VCIP 2016 and the Platinum Best Paper award in IEEE ICME 2017. He served as special session chair in VCIP 2016 and area chair in ICIP 2017.

Xin Guo received the B.E. degree from the School of Electrical and Information Engineering, Tianjin University, Tianjin, China, in 2017. She is currently pursuing the M.E. degree at the School of Electrical and Information Engineering, Tianjin University, Tianjin, China. Her research interests are mainly in 3D human motion recovery.

Kun Li received the B.E. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2006, and the master and Ph.D. degrees from Tsinghua University, Beijing, in 2011. She visited École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, in 2012 and from 2014 to 2015. She is currently an Associate Professor with the School of Computer Science and Technology, Tianjin University, Tianjin, China. Her research interests include dynamic scene 3D reconstruction and image/video processing. She was selected into Peiyang Scholar Program of Tianjin University in 2016, and got the Platinum Best Paper award in IEEE ICME 2017.

Meiyuan Wang received her B. E. degree in Electronic Information Engineering, Dalian University of Technology in 2015, and received her M.E. degree at the School of Electrical and Information Engineering, Tianjin University in 2018. She is currently working in Hangzhou as a software engineer.

Yu-Kun Lai received his bachelor and Ph.D. degrees in computer science from Tsinghua University in 2003 and 2008, respectively. He is currently a Senior Lecturer of Visual Computing in the School of Computer Science & Informatics, Cardiff University. His research interests include computer graphics, geometry processing, image processing and computer vision. He is on the editorial board of The Visual Computer.

Feng Wu (M’99-SM’06-F’13)received the B.S. degree in Electrical Engineering from XIDIAN University in 1992. He received the M.S. and Ph.D. degrees in Computer Science from Harbin Institute of Technology in 1996 and 1999, respectively. Now he is a Professor in University of Science and Technology of China and the dean of School of Information Science and Technology. Before that, he was principle researcher and research manager with Microsoft Research Asia. His research interests include image and video compression, media communication, and media analysis and synthesis. He has authored or co-authored over 200 high quality papers. He has 77 granted US patents. His 15 techniques have been adopted into international video coding standards. As a co-author, he got the best paper award in IEEE T-CSVT 2009, PCM 2008 and SPIE VCIP 2007. Wu has been a Fellow of IEEE. He serves as an associate editor in IEEE Transactions on Circuits and System for Video Technology, IEEE Transactions on Multimedia and several other International journals. He got IEEE Circuits and Systems Society 2012 Best Associate Editor Award. He also serves as TPC chair in MMSP 2011, VCIP 2010 and PCM 2009, and Special sessions chair in ICME 2010 and ISCAS 2013.