Recovering Trajectories of Unmarked Joints in 3D Human Actions Using Latent Space Optimization

Suhas Lohit
Mitsubishi Electric Research Laboratories
Cambridge, MA

Rushil Anirudh
Lawrence Livermore National Laboratories
Livermore, CA

Pavan Turaga *
Arizona State University, Tempe AZ

Abstract

Motion capture (mocap) and time-of-flight based sensing of human actions are becoming increasingly popular modalities to perform robust activity analysis. Applications range from action recognition to quantifying movement quality for health applications. While marker-less motion capture has made great progress, in critical applications such as healthcare, marker-based systems, especially active markers*, are still considered gold-standard. However, there are several practical challenges in both modalities such as visibility, tracking errors, and simply the need to keep marker setup convenient wherein movements are recorded with a reduced marker-set. This implies that certain joint locations will not even be marked-up, making downstream analysis of full body movement challenging. To address this gap, we first pose the problem of reconstructing the unmarked joint data as an ill-posed linear inverse problem. We recover missing joints for a given action by projecting it onto the manifold of human actions, this is achieved by optimizing the latent space representation of a deep autoencoder. Experiments on both mocap and Kinect datasets clearly demonstrate that the proposed method performs very well in recovering semantics of the actions and dynamics of missing joints. We will release all the code and models publicly.*

1. Introduction

With the proliferation of low-cost sensing devices, the use of action and gesture recognition has created new capabilities in health tracking, home-based rehabilitation etc. In many of these applications, the essential task at hand is inferring high-level semantic quantities such as the health of the patient, quality of movement, progression of therapy protocols and intended gestures. In these situations, marker-based motion capture, marker-less devices such as Kinect, and more recently wearables like IMU motion sensors, are considered critical to obtain accurate health tracking information. Marker-based mocap systems, especially with active marker sets, which include uniquely colored LEDs per joint, are more accurate. However, they still suffer from self-occlusion and portability problems due to the need to carefully markup many joints. Also, reducing occlusions would require an expensive setup with a large number of cameras which is not desirable. Devices such as the Kinect have not fully realized their potential in this application space due to similar reasons such as occlusion and tracking errors. These errors tend to have a large impact on assessment of human movement quality in health applications [39]. Further, specific conditions such as postural deformities may result failure in tracking [29] and in the posture not even being detected [1]. Thus, there is a growing need to use reduced marker-sets, while maintaining the accuracy of tracking afforded by marker-based motion capture systems.

In this paper, we present a framing of this problem from the perspective of inverse problems, deep priors and dynamical systems theory. Considering human movement as a dynamical process, estimating its dynamical system is non-trivial and would require us to completely observe all system variables (all joints), due to large degrees of freedom and complex interactions between the joints. In applications requiring inference of higher-level features from movement, the classical approach has been to seek certain invariant representations of the dynamical process that can still be determined from partial observations [40], for e.g. estimating the dynamics of human movement from a few skeletal joint sequences [4, 36]. However, such topological features are not predictive enough for complex datasets. More recently, with the availability of large datasets and neural networks,

we are able to implicitly learn the dynamics much better, leading to significantly higher performance on challenging benchmarks [35, 46, 42, 33] but they tend to be very sensitive to changes in measurement settings during train and test times.

Contributions: In this paper, we focus on the problem of reconstructing trajectories of unmarked/unobserved markers for 3D human action sequences at test-time. That is, given a human skeletal action sequence with a fraction of joints unmarked throughout the action, can we accurately reconstruct the dynamics of those unmarked joints and use the resulting completed action for action recognition without any modifications to the classification architectures? We study this problem when a significant subset of the joints is missing at both training as well as test times, with different rates of dropped joints. In practice, this can occur due to factors such as occlusions, faulty sensors, non-standard postures etc. In this setting, conventional deep learning frameworks fail due to the unaccounted distribution shift between training and testing data sets, and because when the joints are missing, a lot of information regarding the dynamics is lost. To this end,

1. We show that it is possible to effectively recover semantics and dynamics of human actions while having access to as few as 50% of the joints during both training and testing.

2. We pose the problem as an ill-posed linear inversion in the space of actions, and solve it with the help of a deep prior i.e., in the form of a pretrained temporal convolutional decoder. The solution to the inverse problem is obtained by approximately projecting the incomplete action onto the range of the decoder, as shown in Figure 1.

3. The proposed method is a test-time approach that can easily handle changes in the configuration and number of missing joints at test time, when compared to the training set. We demonstrate its effectiveness using real world mocap and Kinect datasets. We validate that the reconstructions recover information about the semantics and dynamics of the actions and show that in most cases the recovered actions perform within 5% of fully visible ground truth data in terms of action recognition on a pre-trained classifier, and yields a very low 3D joint reconstruction error of about 3cm.

2. Related Work

2.1. 3D human action acquisition and completion

The two most common methods of capturing 3D human skeletal actions are using motion capture (mocap) systems, and using depth-sensing cameras like Kinect and RealSense and then using a algorithm to estimate the joint locations. However, both these methods suffer from drawbacks. Not all joints may be recorded mainly due to occlusions and non-standard body types/poses. Moreover, in order to reduce the cost of mocap systems, we would like to employ fewer cameras which leads to the possibility of joints being unseen throughout the actions, which we refer to as unobserved joints. When the final application is action classification, it may be possible to classify directly with fewer joint trajectories, but this suffers from two disadvantages: (a) it is impractical to train a classifier for every combination of unobserved joints (b) the recognition performance is very poor when many joints are missing such as those considered in this paper. As we will show, we can get excellent performance when we first reconstruct the complete set of joint trajectories by exploiting information about the nature of human actions from a training dataset, and then performing the classification using a single pretrained classifier. Earlier works have considered the problem of human action completion in different ways – human action prediction where given a few frames of a human action sequence, the future frames can be predicted employing machine learning algorithms [3, 9, 12], human motion synthesis [8] and editing [14], and pose estimation [37]. There are also traditional methods for human action completion using k-nearest neighbors from the training set [2], and matrix factorization methods [10]. More recently, Yang et al. [43] and [41] propose using low-rank and sparsity priors to model spatial-temporal correlation for 3D human motion recovery problems. Here, the authors enforce the low-rankness of a matrix whose columns are the frames of a test sequence. However, as is in this paper, if the joint trajectories are missing throughout the action, the matrix is already low-rank and the optimization cannot yield the solution. In this paper, we consider the problem of predicting complete joint trajectories given a subset of joints of every frame in the sequence and propose a deep learning-based solution. Kucherenko et al. [17] propose feed-forward pass through trained a LSTM autoencoder as a way of reconstructing human actions. We use a baseline similar to this method in our experiments.

2.2. Deep priors for inverse problems

Several important problems in imaging can be cast as ill-posed linear inverse problems. The forward operation is given by $\mathbf{y} = A\mathbf{x}$, where $\mathbf{y} \in \mathbb{R}^m, \mathbf{x} \in \mathbb{R}^n, m < n$. The goal in inverse imaging is to reconstruct $\mathbf{x}$ from $\mathbf{y}$ which is generally ill-posed by exploiting structure of the desired $\mathbf{x}$ known apriori. In recent years, the prior knowledge comes in the form of a deep generative model (autoencoders, variational autoencoder, generative adversarial networks etc.). The process of reconstruction reduces to the problem of finding the closest point on the range space of the generative model $\hat{\mathbf{x}}$ such that $A\hat{\mathbf{x}} \approx \mathbf{y}$. This idea also has theoretical guarantees as shown by Bora et al. [5] and Shah

Initial reconstruction   *Final reconstruction*   *Ground truth*

Manifold of human action sequences

Test sequence with unobserved joints

$L(Y, A\hat{X}) = ||Y - A\hat{X}||_2^2$

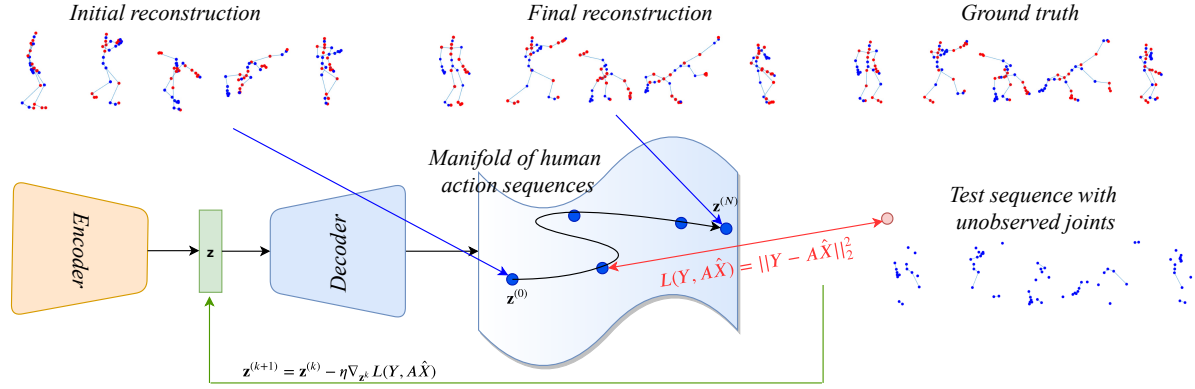$\mathbf{z}^{(k+1)} = \mathbf{z}^{(k)} - \eta \nabla_{\mathbf{z}^k} L(Y, A\hat{X})$

Figure 1: Block diagram illustrating the process of reconstructing dynamics of unobserved joints by projecting onto the manifold of human actions obtained as the range space of a generative model. For a test sequence with several missing joints, we get an initial estimate of the reconstruction using its encoder representation $\mathbf{z}^{(0)}$. Then, we optimize over the encoder representation space in order to minimize the distance between the output of the decoder and the given test sequence, as shown. This procedure can correct for the mismatch between training and test sets and greatly improves the final reconstructed sequence.

and Hegde [31]. Bora et al. [6] showed that, in some cases, the generative model can also be *trained* using the noisy images. Recent papers apply this idea for time-series imputation [22, 44]. However, these techniques are shown to work for simpler time-series classification problems using measurements where some information is available from all dimensions. Also related are the works of Litany et al. [20] which uses graph convolutional deep priors for 3D shape completion, and that of Holden et al. [13] who propose a denoising neural network for frame-wise denoising of motion capture data. In contrast to these works, we consider human action sequences in which certain joints are **completely unobserved** i.e., many dimensions of the time-series are missing. Thus, interpolation techniques in the time domain are not applicable. Also, current methods for skeleton completion is not usually performed at the level of actions (such as Kinect), and **do not take long-range dynamics into account**, which is the goal of this paper and necessary for human action recognition. We use a baseline very similar to [13] which shows the importance of modeling actions rather than individual frames. Also, unlike previous literature, we consider the effect of recovering trajectories of unobserved joints on the **action recognition** performance on a pre-trained classifier, and importantly, the setting where the training set for the autoencoder itself consists of actions with unobserved joints.

2.3. Dynamical systems approach to deal with missing dimensions

In dynamical systems theory, the notion of reconstructing high-dimensional state-spaces from low-dimensional observations has been well-studied. For classical state-estimation approaches to apply, one often needs to make simplifying assumptions for state-dynamics, such as

Markovian and linear dynamics. Such assumptions are not always reflective of the complexity of the task at hand – reconstructing human action sequences. Another approach is to avoid such assumptions, but use methods from non-linear dynamics [21] to estimate surrogate state-spaces. From standard methods in non-linear dynamics, these surrogate state-spaces are only topologically equivalent to the true state-spaces, and do not have enough predictive information for high-level inference [36].

3. Reconstructing dynamics of unmarked joints as an ill-posed linear inverse problem

Recovering the dynamics of unobserved joints is an ill-posed inverse problem since we only have access to partial information of the activity. As we will show, this can be considered analogous to the inverse problems in imaging such as super-resolution, image inpainting or compressive sensing. However, unlike inverse imaging problems, it is not clear what kinds of priors work for human actions. We argue that a deep prior learned from a dataset of human actions acts as a good approximation to the space of all possible dynamical systems for human actions. As a result, we are able to implicitly constrain the dynamics of the recovered actions by restricting the solution to lie on the *action manifold*. We formalize these ideas next.

Let the total number of joints per frame be denoted by J and the number of frames per action sequence by N. Each joint is described by its 3D co-ordinates in space. Thus, by vectorizing, each skeleton can be represented by $3J-$dimensional vector and by stacking the N frames in columns, we represent the human action as a matrix X of size $3J \times N$. Let the number of joints observed be K, so the number of unobserved joints is $J - K$. The mea-

surement operator A then is a sub-sampling operator which drops $3(J-K)$ rows of X to give us the observed action Y. As $J-K$ joints are unobserved, there are $3(J-K)$ rows of Y which are unknown and we replace them with 0 before further processing. Given Y, our eventual goal is to classify the action. As an intermediate step, we first reconstruct $\hat{X}$ from Y which is the main focus of this paper. Clearly, this is an ill-posed linear inverse problem. The advantage of viewing it as such helps us adapt algorithms designed for inverse imaging problems such as image inpainting [27], super-resolution [7, 19] and compressive imaging [18]. In this paper, we adapt the most recent approaches based on generative priors [5, 6, 34]. We note that the advantage of these methods over other methods in inverse imaging such as purely data-driven approaches [7, 18], and unrolled iterative methods [30, 25, 45] is that there is no requirement of paired Y and X for training. Thus, once we have a deep prior for human action sequences, the problem of reconstructing dynamics of unobserved joints can be solved using an optimization problem such that the output of the generative model $\hat{X}$ is closest (in some predefined sense) to the test sequence under consideration Y.

4. Learning the manifold of 3D human actions

In order to approximate space of human actions, we employ a temporal convolutional autoencoder architecture to construct the generative model of human action sequences. We choose an autoencoder over generative adversarial networks [11] or variational autoencoders [16] because – (1) autoencoders are much easier to train compared to the other frameworks and (2) the purpose of using the generative model in this paper is to perform reconstruction of test sequences rather than sampling new actions, which, as we will show, can be readily performed using an autoencoder. We can also employ more complex deep priors which can better model spatial relationships using graphs as recently shown by Wang et al. [38], however this is not the focus of this paper. Note that the term generative model is more broadly used here. Even though we cannot easily sample new actions using an autoencoder, once an initial valid latent space is given, we can move in the latent space in order to refine the output, based on some metric. We discuss this aspect in detail in Section 5.1.

4.1. Autoencoder architecture

As the generative model, we employ a temporal convolutional autoencoder. Both the encoder (E) and decoder (D) consists of a series of 1D convolutional layers operating in the temporal domain with ReLU non-linearity. After every convolution, we use average pooling to reduce the number of frames by half. We then use a fully-connected (FC) layer which produces the encoded/latent representation of the action, denoted by $\mathbf{z}$. The decoder reverses these oper-

ations with a series of transposed convolutional layers. The networks are trained using full/complete actions with access to information of all joint trajectories. The network is trained to minimize the Euclidean loss between the input sequence and the output of the decoder: $L(X, \hat{X}) = \sum_{n=1}^{N} \sum_{j=1}^{J} \left\| X_{n,j} - \hat{X}_{n,j} \right\|_2^2$, where $X_{n,j}$ refers to the jth 3D joint location in the nth frame of the sequence. Other training details are provided in the supplementary material. We note that we can add an additional **adversarial loss** term to the above loss function in order to make the actions more realistic [27]. However, in our experiments, we did not observe any significant improvements using this additional loss term.

4.2. Training the autoencoder with partially observed joint sequences

The autoencoders in Sections 4.1 are trained using complete actions with all joint sequences fully observed, X. However, complete actions may not be available at the training stage. *An important contribution of this paper is showing that we can construct the manifold of human actions by training solely on action sequences with only a subset of the joints observed, Y.* We later show that this protocol leads to superior reconstruction compared to training with full actions. To this end, we modify the loss function as follows. The forward operator A is the sampling operator which has the effect of dropping a subset of the joints. Using the knowledge of A, we use a masked loss function between the ground-truth measured sequence $Y = AX$ and the reconstructed sequence $\hat{X}$: $L(Y, \hat{X}) = \sum_{n=1}^{N} \sum_{j=1}^{J} \left\| Y_{n,j} - A\hat{X}_{n,j} \right\|_2^2$. The network architectures and the training protocols are identical to those trained using fully observed action sequences.

5. Reconstruction via approximate projection onto the action manifold

Once we have the deep prior model, in our case the decoder stage of the autoencoder, the training process is complete. The next step is to use the prior model in order to reconstruct the trajectories of unobserved joints given an incomplete action sequence. To this end, we propose to project the incomplete action to the range space of the generator, which ideally is the same as the manifold of complete human action sequences.

Initialization: Feed-forward pass through the trained autoencoder: As a baseline method, we can simply feed the incomplete action sequence through the autoencoder and use the output of the decoder as the reconstruction. This is expected to fail, because of the distribution shift between

training and test set, especially in the case of the autoencoder trained with complete actions. However, in the case of the autoencoder trained on subsets of joint trajectories, even this simple method can provide a reasonable reconstructed sequence. This is used as initialization for the optimization algorithm below, $\mathbf{z}^{(0)}$.

5.1. Optimizing the latent representation

We can further improve the reconstruction quality from above by directly optimizing the encoded/latent representation, $\mathbf{z}$, such that the Euclidean distance between the reconstructed action sequence and the input incomplete sequence. This method is inspired by Bora et al. [5] where the authors propose this method for inverse problems in imaging. The optimization problem is given by

$$\mathbf{z}^* = \arg\min_{\mathbf{z}} \|Y - AD(\mathbf{z})\|_2^2, \quad \hat{X} = D(\mathbf{z}^*). \quad (1)$$

This ameliorates the train/test distribution-shift issue and allows a better exploration of the output space of the decoder. We solve this optimization problem using gradient descent-type method. As the optimization problem is non-convex, the optimization returns a $\mathbf{z}^*$ which is guaranteed to be only locally optimal. Empirically, we find that the solutions obtained using this procedure provide high quality reconstructions.

6. Measuring reconstruction performance

Our main goal in this paper is to recover trajectories of unobserved joints from incomplete human actions. We choose the following three methods for measuring the quality of reconstructed actions.

6.1. Reconstruction error

We measure the root-mean-square error (RMSE) between the reconstructed actions and the ground-truth actions. For a reconstructed sequence $\hat{X}$ and the corresponding ground-truth sequence X, RMSE is given by
$RMSE(X, \hat{X}) = \sqrt{\frac{1}{NJ} \sum_{n=1}^{N} \sum_{j=1}^{J} \left\| X_{n,j} - \hat{X}_{n,j} \right\|_2^2}$.

6.2. Classification performance

An important reason for reconstructing action sequences is to employ predefined classification pipelines without any modification. Therefore, we train a single action classifier on sequences with information of all joints, feed the reconstructed sequences as test inputs, and use the classification performance as a metric for quality of reconstructed actions.

Classifier architecture: In all our experiments, we employ a simple popular architecture for 3D human action recognition based on temporal convolutional networks

(TCNs) [15]. The classifer consists of a series of temporal convolutional blocks. Each block consists of layers of 1D convolutional layers operating in the temporal domain with ReLU non-linearity. We employ batch normalization for each layer. Residual connections are employed from one block to the next. After every block, average pooling is employed to reduce the number of frames by half. Finally a fully-connected (FC) layer with softmax is used to map to a probability distribution over the classes. As there may be a domain shift between the original actions and the reconstructed actions from the autoencoder, we train a single classifier on complete action sequences. Other training details are provided in the supplementary material. We also visualize the effectiveness of the reconstructions for downstream applications, like classification, with t-SNE embeddings in 2D [23]. We use the feature maps of the penultimate layer of the trained classifier and are shown in the supplement.

7. Experimental results

In the experiments section, we simulate mocap and Kinect with unmarked/unobserved joints using benchmark datasets which have fully observed joints.

7.1. Baselines

Dictionary learning and sparse coding [24, 28]: This is a commonly used algorithm for solving ill-posed problems. We first construct a dictionary based on all the action sequences in the training set. Dictionary learning is performing by using matrix factorization with sparsity and norm constraints on the matrix factors. Given a test sequence, in which joint trajectories are missing, we express the test sequence as a sparse linear combination of the atoms in the dictionary. The linear combination is used as the completed sequence. The sparse code is found using orthogonal matching pursuit. Please see the supplement for details.

Frame-wise completion with deep learning: In order to demonstrate the effect of modeling temporal variations (which is one of the main contributions of this paper), jointly with purely frame-wise information, we construct a baseline very similar to that in Section 4. The only difference is that instead of encoding and decoding actions directly, the action is split into frames and the decoding is done on a per-frame basis. Thus, no temporal correlations are modeled in the deep prior. Further details about the architecture are given in the supplement. This baseline is similar in spirit to the work of Litany et al. [20] for 3D human shape completion.

Decoder reconstruction: As described in Section 4.2, a feed-forward pass through the trained autoencoder itself can be used to complete the sequence. And we use this as a baseline to compare with the further latent space optimization we propose in this paper.

7.2. HDM05 mocap dataset [26]

Dataset details: HDM05 is a large publicly available and challenging database of 3D human actions with 2337 action samples. There are 130 different types of actions performed by 5 subjects and recorded in a laboratory setting using an optical motion capture system. Each skeleton is made up of 31 joints. For our experiments, we resample all the actions such that the length of the every action sequence $N = 100$. Thus $X, Y, \hat{X} \in \mathbb{R}^{93 \times 100}$. All sequences are normalized so that the hip joint is fixed at the origin in 3D space. We perform 5-fold cross-validation. For each run, we use 4 subjects for training and the remaining subject for testing with about 1850 samples for training and the rest for testing. Note that, in all cases, as a pre-processing step, the unobserved joint trajectories are replaced with the respective mean trajectories computed over the observed joints in the training set. However, simply using zeros as the initialization for the missing joints produces nearly identical results and these are shown in the supplement.

Network architectures: The encoder consists of 4 temporal convolution layers with filter size of 4, and the number of feature maps in each layer is set to 75 (equal to the number of channels at the input layer). We use a latent space dimension of 200. The decoder consists of 4 temporal transposed convolutional layers. For our experiments, we train multiple autoencoders with actions consisting of random subsets of joint sequences sampled from the actions. Different fractions of joints included for training each autoencoder: $100\%, 75\%, 50\%$. We will use the term Observed-to-Total Percentage (OTP) = $\frac{K}{J} \times 100$, to denote this quantity. For classification, we use a TCN classifier similar [15]. It consists of 3 TCN blocks with one convolutional layer each.

Reconstruction performance and visualization: We compare the results of the proposed method with the baselines described in Section 7.1. As explained in Section 5.1, using our proposed method, we can achieve significantly better reconstructions by using an optimization procedure over the latent space of the autoencoder model in order to minimize the Euclidean distance between the reconstruction and input test sequence with only a subset of observed joint trajectories. Note that the parameters of the generative model, the decoder D in our case, are held fixed for this optimization. We use Adam optimizer for 500 iterations with an initial learning rate of 1.0. As the initialization, we use the latent representation of the incomplete action obtained by a feed-forward pass through the autoencoder. We carry out reconstruction experiments on the test set for autoencoders trained with different fractions of observed joints.

In order to better test the generalization ability, we use test-time OTPs that are different from training-time OTPs. For the reconstructed sequences thus obtained, we use a pre-trained classifier to classify the test set reconstruction into one of the pre-defined 130 classes for different variants and test-time OTPs of the autoencoders. The RMSE of the reconstructed sequences as well as the action recognition performance, averaged over the five folds, are shown in Tables 1 and 2, and a few sample reconstructed skeletal sequences are shown in Figure 2.

Results: We observe in all cases where either the training or test OTP is less than 100%, the proposed method of solving the optimization problem in Equation (1) i.e., using $D(\mathbf{z}^*)$ leads to significantly better results compared to baselines including using a single feed-forward pass through the autoencoder i.e., $D(E(Y))$, where D and E are the decoder and encoder respectively. In almost all cases, using the optimization approach yields accuracies within 5% points of the oracle action recognition performance and about 3 cm error in terms of RMSE. We also observe for the cases with training OTP = 100%, using a different test OTP causes more degradation in performance than when the train OTP = $75\%, 50\%$. **We note that we also tested two additional simpler baselines**: (1) perform reconstruction of skeletons per frame by replacing every unobserved joint with the closest observed joint in the skeleton, and then use the pre-trained classifier and (2) train the classifier directly on the action sequences with only a subset of joints observed. Irrespective of the train/test OTP, both these baselines fail and yield only accuracies which are close to chance.

In the supplement, we provide an additional baseline using a denoising autoencoder (DAE) framework where the only difference from the framework presented here is that, we assume that we have access to a training set with all joints trajectories available and the DAE maps from incomplete actions to complete ones. That is, rather than measuring the error only in the subset of joints (Equation 2), we measure the error for all the joints as the ground-truth training actions are assumed to be perfect. The results show this extra supervision actually hurts the final test performance when the Train and Test OTPs are different, compared to the proposed method. As in the other cases, latent space optimization helps improve the performance. We also performed an experiment where, different joints are missing for different frames, rather than joints missing for the entire sequence. The proposed method can be applied without any modifications, and the results show that the performance and trends remains the same as in the case of completely missing joints (which is the focus of this paper). These results are also provided in the supplement.

Reconstruction using structured masks: In the above, we trained and tested autoencoders with random subsets of joints dropped. In this experiment, we drop joints in a structured fashion during test time. We carry out four sets of experiments with the joints corresponding to the following body parts dropped: right arm (6 joints), left arm (6), right leg (4) and left leg (4). This demonstrates how occlusion
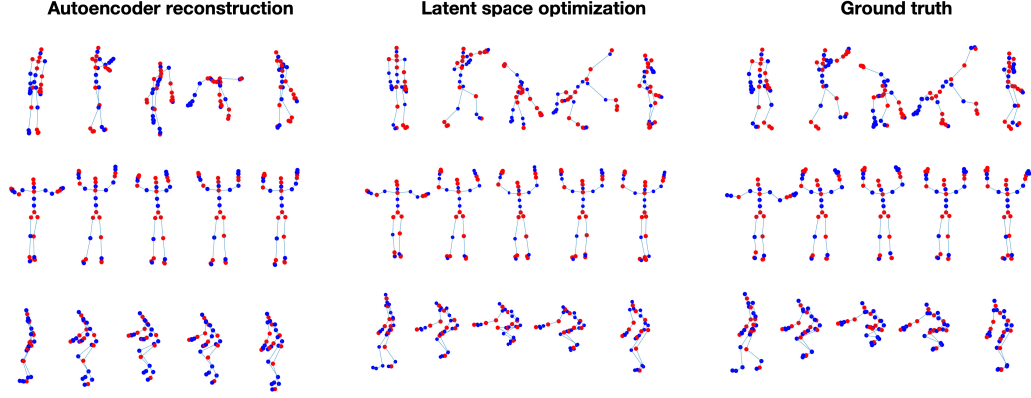
Autoencoder reconstruction　　　　**Latent space optimization**　　　　**Ground truth**

Figure 2: Reconstructed actions for the HDM05 database for train OTP / test OTP = 75/50. From the top row, the actions shown are "Cartwheel" ,"Hand waving" and "Grab low". The first column shows the reconstructions obtained by a simple feedforward pass through the trained autoencoder: $D(E(Y))$. The middle column shows the output of the proposed approach which solves the optimization problem in Equation 1: $D(\mathbf{z}^*)$. Blue dots represent the observed joints and red dots represent the unobserved. We clearly observe that the optimization approach produces superior reconstructions.

Train OTP / Test OTP	Method	HDM		NTU	
		RMSE (cm)	Acc (%)	RMSE (cm)	Acc (%)
100/100	$D(E(Y))$	3.48	79.23	2.67	74.77
100/75	$D(E(Y))$	6.21	61.21	6.18	55.90
	$D(\mathbf{z}^*)$	**2.18**	**78.04**	**3.33**	**72.26**
100/50	$D(E(Y))$	9.87	28.99	9.39	31.89
	$D(\mathbf{z}^*)$	**2.99**	**73.47**	**5.19**	**65.15**
75/75	$D(E(Y))$	6.07	68.72	6.08	59.15
	$D(\mathbf{z}^*)$	**2.27**	**78.61**	**3.49**	**71.64**
75/50	$D(E(Y))$	8.52	44.81	9.11	37.38
	$D(\mathbf{z}^*)$	**2.98**	**74.71**	**5.30**	**64.81**
50/50	$D(E(Y))$	8.77	43.55	9.37	34.93
	$D(\mathbf{z}^*)$	**3.05**	**74.37**	**5.29**	**64.59**

Table 1: Experimental results for HDM05 (averaged over 5 folds) and NTU datasets for varying train/test OTPs in terms of RMSE and action recognition accuracy (Acc). We observe easily that the proposed optimization-based reconstruction is far superior to a feedforward pass through the autoencoder in terms of all the metrics considered. As the train OTP is reduced, performance degrades more gracefully in the case of the optimization-based approach. In all cases, we can get to within 5% points of the oracle action recognition performance (train /test OTP = 100/100).

of different limbs can affect the performance of our algorithm. Note that the autoencoders were trained on random subsets of joints, as before. The results are shown in Table 3. We see once again that the proposed algorithm yields good recognition performance compared to the baselines considered. These results also illustrate another important aspect of the work—even though the train and test OTP are nearly the same as in the case of occluding right/left leg, the baseline method's performance suffers significantly as the distribution of the joints being dropped is very different from the training scenario (random subsets of joints are dropped), and the proposed latent space optimization technique allows better exploration of the decoder space in order to find better reconstructions.

Generalizing to unseen actions:　Here, instead of having a test set having actions from an unseen subject, the test

set actions also belong to *classes* which are unseen during training time. For our experiment, we use a random split of 100 and 30 action classes for the train and test set respectively. The joint RMSE results shown in Table 4 show that our method is indeed able to yield good results in this challenging setting, showing that the proposed method is able to generalize to both unseen subjects and unseen classes.

7.3. NTU RGB-D dataset [32]
Dataset details:　This is a large database of about 56000 3D skeletal action sequences obtained from Kinect of actions belonging to 60 classes and performed by 45 subjects. For each skeleton, 25 joint locations are provided. We resample all the sequences to have $N = 50$ frames. Thus $X, Y, \hat{X} \in \mathbb{R}^{75 \times 50}$. We perform the experiments in the cross-subject setting and use the train-test split as suggested

Method	HDM		NTU	
	RMSE (cm)	Acc (%)	RMSE (cm)	Acc (%)
Sparse Coding [24]	20.27	11.30	19.32	7.22
Frame-wise $D_F(E_F(Y_F))$ similar to [13]	21.47	12.30	20.40	9.69
Frame-wise $D_F(\mathbf{z_F}^*)$	21.28	13.44	20.19	8.62
Action $D(E(Y))$	8.52	44.81	9.11	37.38
Action $D(\mathbf{z}^*)$ (Proposed)	**2.98**	**74.71**	**5.30**	**64.81**

Table 2: Experimental results for HDM05 (averaged over 5 folds) and NTU datasets compared to different baselines for train/test OTP = 75/50. We observe easily that the proposed optimization-based reconstruction is superior to all the baselines considered. D_F and E_F refer to the fact that the encoder and decoder operate on a single frame at a time, rather than an action sequence.

Occluded limb/ Frac. joints dropped	Train OTP	Method	RMSE (cm)	Recog. Acc. (%)
Right arm/ $6/31 \approx 20\%$	90	$D(E(Y))$	9.66	48.97
		$D(\mathbf{z}^*)$	**7.03**	**61.55**
	75	$D(E(Y))$	9.43	51.62
		$D(\mathbf{z}^*)$	**6.54**	**64.89**
Left arm/ $6/31 \approx 20\%$	90	$D(E(Y))$	9.51	51.53
		$D(\mathbf{z}^*)$	**6.88**	**65.29**
	75	$D(E(Y))$	9.38	52.29
		$D(\mathbf{z}^*)$	**6.55**	**68.49**
Right leg/ $4/31 \approx 13\%$	90	$D(E(Y))$	6.32	58.61
		$D(\mathbf{z}^*)$	**4.25**	**67.95**
Left leg/ $4/31 \approx 13\%$	90	$D(E(Y))$	6.50	53.30
		$D(\mathbf{z}^*)$	**4.24**	**64.20**

Table 3: Average classification results (over 5 folds) of reconstructed actions on the HDM05 database. The inputs are actions with contiguous body parts that are hidden or unobserved. Here, $D(E(Y))$ is the baseline and $D(\mathbf{z}^*)$ is the proposed optimization strategy.

by the authors of the dataset. The training set consists of about 40000 examples and the remaining are in the test set. All sequences are normalized so that the hip is fixed at the origin in 3D space.

Network architectures: The generative mode is a temporal convolutional autoencoder. The encoder consists of 3 temporal convolution layers with filter size of 8, and the number of feature maps in each layer is set to 75 (equal to the number of channels at the input layer). We use a latent space dimension of 200. The decoder consists of 3 temporal transposed convolutional layers. As the action classifier, we use a TCN classifier identical to that proposed by Kim and Reiter [15]. It consists of 3 TCN blocks with 3 convolutional layers each.

Reconstruction performance: We conduct an identical set of experiments as in the case of the HDM05 dataset.

Train OTP / Test OTP	Method	RMSE (cm)
100/100	$D(E(Y))$	1.79
100/75	$D(E(Y))$	5.34
	$D(\mathbf{z}^*)$	**1.43**
100/50	$D(E(Y))$	9.42
	$D(\mathbf{z}^*)$	**2.14**
75/75	$D(E(Y))$	5.55
	$D(\mathbf{z}^*)$	**1.99**
75/50	$D(E(Y))$	6.92
	$D(\mathbf{z}^*)$	**2.68**
50/50	$D(E(Y))$	6.67
	$D(\mathbf{z}^*)$	**2.76**

Table 4: Experimental results for HDM05 when training and test action classes are different, for varying train/test OTPs in terms of RMSE. The results are averaged over 5 runs where for each run, we randomly select the action classes for train and test. We observe easily that the proposed optimization-based reconstruction is far superior to a feedforward pass through the autoencoder.

The action classfication accuracies and RMSE are shown in Tables 1 and 2. Skeletal visualizations are provided in the supplement. The trends observed are the same as those in HDM05. Compared to all the baselines considered including (1) sparse coding (2) frame-wise completion and (3) using the autoencoder reconstruction, the proposed method of latent space optimization achieves far superior results especially when train and test OTPs are considerably different, and gets close to oracle classification performance and RMSE even with just 75% of observed joints. Results for other train/test OTPs are provided in the supplement.

8. Conclusions

We consider the problem of reconstructing completely unobserved dimensions of a multi-variate time series. The problem is traditionally studied in the framework of system identification and non-linear dynamics. However, for tractability, such methods make strong assumptions on the data such as linearity of the underlying dynamical system, sparsity of observations in transform domains etc. In this paper, we study the specific example of reconstructing dynamics of unobserved joints from 3D human actions, for which we cannot easily construct hand-crafted priors. We propose to first construct a deep prior of complete actions, even when the training data has up to 50% of the joints missing. The reconstruction problem then can be solved by projecting the observed action with missing joints onto the action manifold, which is done via optimization in the latent space. Through extensive experiments and different metrics, we show that the proposed approach can effectively recover the dynamics of unobserved joints. Interesting extension of this ideas for human actions include designing stronger priors using spatio-temporal graph convolutional autoencoders and different improved loss functions for latent space optimization.

References

[1] Kinect'in therapy: Disabled access to the Xbox Kinect - hardware and games. `http://kinect-therapy.blogspot.uk`. Accessed: 2020-03-03.

[2] Andreas Aristidou, Daniel Cohen-Or, Jessica K Hodgins, and Ariel Shamir. Self-similarity analysis for motion capture cleaning. In *Computer Graphics Forum*. Wiley Online Library, 2018.

[3] Emad Barsoum, John Kender, and Zicheng Liu. HP-GAN: Probabilistic 3D human motion prediction via GAN. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1418–1427, 2018.

[4] Alessandro Bissacco. Modeling and learning contact dynamics in human motion. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 421–428. IEEE, 2005.

[5] Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G Dimakis. Compressed sensing using generative models. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 537–546. JMLR. org, 2017.

[6] Ashish Bora, Eric Price, and Alexandros G Dimakis. AmbientGAN: Generative models from lossy measurements. *ICLR*, 2:5, 2018.

[7] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2015.

[8] Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik. Recurrent network models for human dynamics. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4346–4354, 2015.

[9] Partha Ghosh, Jie Song, Emre Aksan, and Otmar Hilliges. Learning human motion models for long-term predictions. In *2017 International Conference on 3D Vision (3DV)*, pages 458–466. IEEE, 2017.

[10] Øyvind Gløersen and Peter Federolf. Predicting missing marker trajectories in human motion data using marker intercorrelations. *PloS one*, 11(3):e0152616, 2016.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[12] Alejandro Hernandez, Jurgen Gall, and Francesc Moreno-Noguer. Human motion prediction via spatio-temporal inpainting. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[13] Daniel Holden. Robust solving of optical motion capture data by denoising. *ACM Transactions on Graphics (TOG)*, 37(4):1–12, 2018.

[14] Daniel Holden, Jun Saito, and Taku Komura. A deep learning framework for character motion synthesis and editing. *ACM Transactions on Graphics (TOG)*, 35(4):138, 2016.

[15] Tae Soo Kim and Austin Reiter. Interpretable 3D human action analysis with temporal convolutional networks. In *2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW)*, pages 1623–1631. IEEE, 2017.

[16] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[17] Taras Kucherenko, Jonas Beskow, and Hedvig Kjellström. A neural network approach to missing marker reconstruction in human motion capture. *arXiv preprint arXiv:1803.02665*, 2018.

[18] Kuldeep Kulkarni, Suhas Lohit, Pavan Turaga, Ronan Kerviche, and Amit Ashok. Reconnet: Non-iterative reconstruction of images from compressively sensed measurements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 449–458, 2016.

[19] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017.

[20] Or Litany, Alex Bronstein, Michael Bronstein, and Ameesh Makadia. Deformable shape completion with graph convolutional autoencoders. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1886–1895, 2018.

[21] Lennart Ljung. System identification. *Wiley Encyclopedia of Electrical and Electronics Engineering*, 2001.

[22] Yonghong Luo, Xiangrui Cai, Ying Zhang, Jun Xu, et al. Multivariate time series imputation with generative adversarial networks. In *Advances in Neural Information Processing Systems*, pages 1596–1607, 2018.

[23] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[24] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the 26th annual international conference on machine learning*, pages 689–696, 2009.

[25] Chris Metzler, Ali Mousavi, and Richard Baraniuk. Learned D-AMP: Principled neural network based compressive image recovery. In *Advances in Neural Information Processing Systems*, pages 1772–1783, 2017.

[26] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber. Documentation mocap database HDM05. Technical Report CG-2007-2, Universität Bonn, June 2007.

[27] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016.

[28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[29] Daniele Regazzoni, Giordano de Vecchi, and Caterina Rizzi. Rgb cams vs rgb-d sensors: Low cost motion capture technologies performances and limitations. *Journal of Manufacturing Systems*, 33(4):719–728, 2014.

[30] JH Rick Chang, Chun-Liang Li, Barnabas Poczos, BVK Vijaya Kumar, and Aswin C Sankaranarayanan. One network to solve them all–solving linear inverse problems using deep projection models. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5888–5897, 2017.

[31] Viraj Shah and Chinmay Hegde. Solving linear inverse problems using GAN priors: An algorithm with provable guarantees. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4609–4613. IEEE, 2018.

[32] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. NTU RGB+D: A large scale dataset for 3D human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1010–1019, 2016.

[33] Yansong Tang, Yi Tian, Jiwen Lu, Peiyang Li, and Jie Zhou. Deep progressive reinforcement learning for skeleton-based action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5323–5332, 2018.

[34] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9446–9454, 2018.

[35] Raviteja Vemulapalli, Felipe Arrate, and Rama Chellappa. Human action recognition by representing 3D skeletons as points in a lie group. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 588–595, 2014.

[36] Vinay Venkataraman and Pavan Turaga. Shape distributions of nonlinear dynamical systems for video-based inference. *IEEE transactions on pattern analysis and machine intelligence*, 38(12):2531–2543, 2016.

[37] Chunyu Wang, Haibo Qiu, Alan L Yuille, and Wenjun Zeng. Learning basis representation to refine 3d human pose estimations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8925–8932, 2019.

[38] He Wang, Edmond SL Ho, Hubert PH Shum, and Zhanxing Zhu. Spatio-temporal manifold learning for human motions via long-horizon modeling. *IEEE transactions on visualization and computer graphics*, 2019.

[39] David Webster and Ozkan Celik. Systematic review of kinect applications in elderly care and stroke rehabilitation. *Journal of neuroengineering and rehabilitation*, 11(1):108, 2014.

[40] Alan Wolf, Jack B Swift, Harry L Swinney, and John A Vastano. Determining lyapunov exponents from a time series. *Physica D: Nonlinear Phenomena*, 16(3):285–317, 1985.

[41] Guiyu Xia, Huaijiang Sun, Beijia Chen, Qingshan Liu, Lei Feng, Guoqing Zhang, and Renlong Hang. Nonlinear low-rank matrix completion for human motion recovery. *IEEE Transactions on Image Processing*, 27(6):3011–3024, 2018.

[42] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[43] Jingyu Yang, Xin Guo, Kun Li, Meiyuan Wang, Yu-Kun Lai, and Feng Wu. Spatio-temporal reconstruction for 3D motion recovery. *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[44] Jinsung Yoon, James Jordon, and Mihaela Van Der Schaar. Gain: Missing data imputation using generative adversarial nets. *International Conference on Machine Learning*, 2018.

[45] Jian Zhang and Bernard Ghanem. ISTA-net: Interpretable optimization-inspired deep network for image compressive sensing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1828–1837, 2018.

[46] Pengfei Zhang, Cuiling Lan, Junliang Xing, Wenjun Zeng, Jianru Xue, and Nanning Zheng. View adaptive neural networks for high performance skeleton-based human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2019.