Object Agnostic 3D Lifting in Space and Time

Christopher Fusco[1] Shin-Fang Ch'ng[1] Mosam Dabhi[2] Simon Lucey[1]

[1]The University of Adelaide [2]Carnegie Mellon University

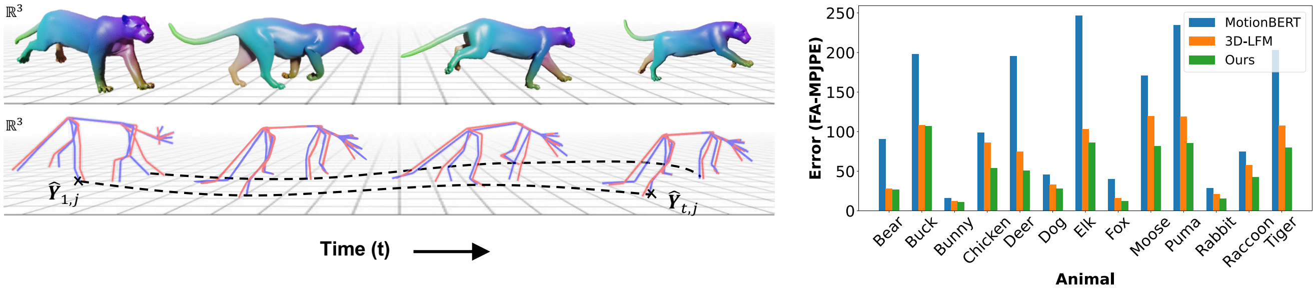


Figure 1. **Left:** Bottom row shows the 3D skeletons of a puma animal in motion. The blue lines represent our model's predictions, closely tracking the red ground-truth lines, demonstrating our model's ability to generate smooth and precise motion over time. The dashed line highlights the trajectory of a specific joint $\hat{\mathbf{Y}}_{t,j}$, emphasizing the temporal consistency and accuracy of our approach. **Right:** Quantitative FA-MPJPE comparison across 13 animal categories, where our method consistently outperforms competing models.

Abstract

We present a spatio-temporal perspective on category-agnostic 3D lifting of 2D keypoints over a temporal sequence. Our approach differs from existing state-of-the-art methods that are either: (i) object-agnostic, but can only operate on individual frames, or (ii) can model space-time dependencies, but are only designed to work with a single object category. Our approach is grounded in two core principles. First, general information about similar objects can be leveraged to achieve better performance when there is little object-specific training data. Second, a temporally-proximate context window is advantageous for achieving consistency throughout a sequence. These two principles allow us to outperform current state-of-the-art methods on per-frame and per-sequence metrics for a variety of animal categories. Lastly, we release a new synthetic dataset containing 3D skeletons and motion sequences for a variety of animal categories.

1. Introduction

Reconstructing 3D deforming objects from 2D landmarks obtained by a single camera is a long-standing challenge in computer vision. Traditional non-rigid structure-from-motion (NRSfM) approaches relied on clever but straightforward factorisation methods that are sensitive to noise and occlusions [3, 29]. In some cases, ambiguities can be resolved with multiple camera views, but come at the cost of expensive equipment and limited practicability to natural scenes. Recent learning-based methods are capable of robustly recovering 3D object structure from a single camera in the presence of noise and occlusions. Coupled with an abundance of publicly available 3D human pose data, human-specific lifting models like MotionBERT [36] have become increasingly capable. However, their reliance on human-specific information and vast amounts of training data make it problematic if they are to be used for other objects. In particular, animals pose a significant challenge due to the limited amount of publicly available 3D animal data.

This has motivated recent developments around object-agnostic lifting, where a single model is capable of lifting various object categories without category-specific fine-tuning. Most notably, 3D-LFM [8] achieved state-of-the-art performance on a combined dataset of various object categories. The permutation equivariant property of transformers and additional skeletal information were leveraged to robustly handle category imbalances and within-category object rig/skeleton variations. However, its inability to utilize temporal information results in poor performance when applied to a sequence of 2D poses obtained from a video. We observe frequent jitter and poor recovery of occluded points, particularly in dynamic sequences.

Our work tackles these challenges by introducing the first object-agnostic 3D lifting framework that is both data-efficient and temporally aware. Our approach leverages the power of transformers with a strategic inductive bias that focuses attention on temporally proximate frames, enabling it to effectively capture motion dynamics. Our approach improves the accuracy of 3D reconstructions across various object categories, particularly in challenging scenarios that contain occlusions, fast movement, limited data, and previously unseen categories. Furthermore, we address the lack of publicly available datasets for lifting diverse animal skeletons and motion sequences by creating a new synthetic dataset. Our dataset, **AnimalSyn3D**, includes 4D labels for 13 animal categories, encompassing 678 animation sequences with temporal consistency, designed to enable further research in class-agnostic lifting.

The contributions of this paper are:

- We propose a class-agnostic lifting model with a strategic inductive bias directly embedded in the architecture. We validate the state-of-the-art 3D lifting performance of our approach across challenging scenarios involving noise, occlusions, and unseen objects.
- We contribute a new synthetic dataset containing 4D skeletons for a variety of animals with animated behavior sequences, where temporal consistency is prioritized through a non-linear refinement procedure.

We empirically validate the effectiveness of our approach on our synthetic dataset. We achieve state-of-the-art results with existing metrics and provide an additional metric for a more complete analysis.

2. Related Works

2.1. 3D Pose Estimation

Obtaining the 3D pose of an object from a single monocular camera generally follows one of two paths. The first directly predicts the 3D pose from RGB images [2, 23, 26, 27], often struggling to generalize to distribution shifts such as lighting and background information. Alternatively, two-stage methods divide the task between two specialized models [4, 5, 12, 21]: a pose detector first extracts a 2D pose which is then lifted into 3D by a separate model. Our work aligns with a two-stage approach, particularly focusing on improving the robustness and generalization of the 3D lifting stage.

2.2. Object-Specific Lifting

Traditional NRSfM algorithms have been effective in modeling simple and targeted objects, such as human bodies and hands [6, 10]. These methods largely rely on the availability of 2D keypoints and specific 3D supervision for the object in question. However, recent deep learning approaches have demonstrated superior performance in handling the complexity of various object rigs [14, 18, 25, 30]. Despite these advancements, they still require the 2D keypoints to have consistent semantic correspondence across all instances of the object, where a specific landmark, such as an elbow, must have the same semantic meaning across different poses.

This limitation persists even in state-of-the-art deep lifting models like MotionBERT [36] and others [4, 5], which are tailored specifically for the human body. The specialized nature of these models and their dependence on large datasets make them unsuitable for objects with limited available data, such as animals. Existing animal-specific lifting models suffer from similar issues, often being restricted to a single animal category and demonstrating poor generalization due to data scarcity [11, 16, 22].

2.3. Object-Agnostic Lifting

The paradigm of object-agnostic lifting has recently been pioneered by 3D-LFM [8], which can handle a wide range of object categories by leveraging large-scale data to enhance performance for underrepresented or unseen objects. However, unlike object-specific models such as Motionbert [36], 3D-LFM does not incorporate temporal information, which is important for accurate 3D reconstruction of sequences. Our framework builds upon this insight, integrating the strengths of both object-agnostic lifting and temporal modeling.

2.4. Animal Datasets

To effectively benchmark category-agnostic lifting over videos of animals, a diverse set of animal categories with accurately labeled 3D skeletons is required. Recent animal datasets have made strides in this area but are often limited to single atemporal images [31, 33], with few publicly available datasets offering 3D poses of animals in video sequences [7, 19]. Moreover, methods to collect data for a broad set of animals are typically impractical, expensive, or yield noisy results.

Inspired by the tracking community, we turn to synthetic data [9, 15, 35]. Existing synthetic datasets are restricted to a single animal, such as pigs [1] or ants [28], and generally contain only simplistic motion sequences. The DeformableThings4D [20] dataset offers more complex and diverse motion sequences animated by artists, but it was created for dense mesh recovery and does not include 3D skeletons. We build upon these models and animations to create a new dataset specifically designed for the task of class-agnostic 3D lifting.

3. Method

In this section we explain our data collection pipeline and class-agnostic lifting model.

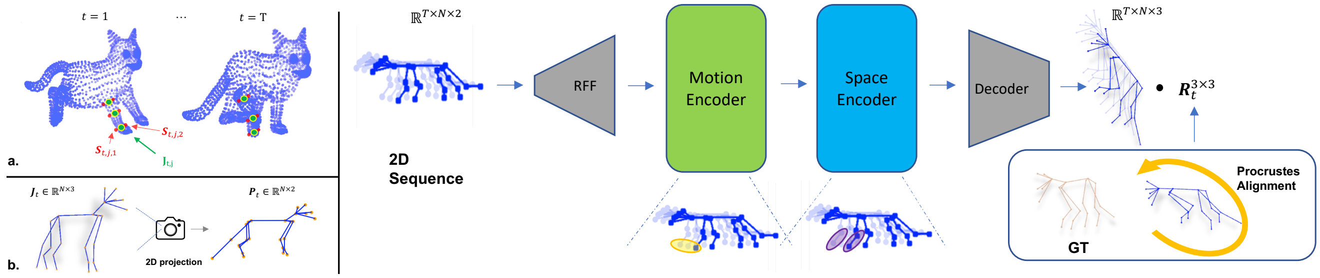


Figure 2. **Overview of our data pipeline and 3D lifting model.** The left side of the figure demonstrates (**a**) the process of calculating skeleton joints from animal mesh vertices, and (**b**) the projection of the those joints into 2D keypoints. The right side of the figure illustrates our lifting model at a high-level. The sequence of 2D input and temporal index is projected and passed through our motion encoder and space encoder layers. The spatio-temporal latent features are decoded into canonical 3D structures. The canonical structures are then aligned to the ground truth (GT) via procrustes-alignment for calculating the loss.

3.1. Dataset

We find a noticeable gap in available public datasets containing diverse 3D animal skeletons with realistic motion sequences. We aim to fill this gap by creating a new synthetic dataset, **AnimalSyn3D**, that builds on the mesh vertices and animation sequences provided by the DeformableThings4D [20] dataset. We provide 3D skeleton labels for 13 animal categories, totalling 678 animation sequences with temporal correspondence across 48,384 frames. We provide statistics and examples of our dataset in the Tab. 7. We detail our data collection pipeline in following sections.

3.1.1 Animal joints

Given the vertices of a skin-tight mesh of an animal and an associated sequence of movement, our aim is to find the 3D locations of the anatomically-accurate skeleton joints of the animal. We define the locations of K vertices and N joints in 3D space throughout a sequence of T frames as $\mathbf{V} \in \mathbb{R}^{T\times K\times 3}$ and $\mathbf{J} \in \mathbb{R}^{T\times N\times 3}$, respectively. Note that the number of mesh vertices, joints, and frames may vary across animals and sequences. We take inspiration from motion capture, where markers are attached to an object and used to estimate joint positions via triangulation. We strategically select a subset of M vertices $\mathbf{S}_{t,j} = \left\{\mathbf{S}_{t,j,1}, ..., \mathbf{S}_{t,j,M}\right\} \subset \mathbf{V}_t$ to be virtual markers, such that the mean of the markers will provide the location of a joint j in frame t:

$$\mathbf{J}_{t,j} = \frac{1}{m}\sum_{i=1}^{m}\mathbf{S}_{t,j,i}. \tag{1}$$

The selection of vertices for a subset $\mathbf{S}_{t,j}$ is guided by a visual inspection of $\mathbf{J}_{t,j}$ and the trajectory similarity of the chosen vertices. The bones of the animal are defined as an adjacency matrix $\mathbf{A}^{N\times N}$ containing the connections between joints. We decide the number of joints, their approximate locations, and their connections by reviewing the anatomical structure of the target animal.

3.1.2 Non-linear optimisation

The noise that is inherently present in our human annotation process occasionally results in the length of animal bones to change over time. We adopt an additional inverse-kinematics optimization procedure to refine the position of joints so that the bone lengths are consistent across time. We formulate the problem as solving for the pose angles $\theta \in \mathbb{R}^{T\times N\times 3}$ of forward kinematics for each joint j in frame t:

$$\widehat{\mathbf{J}}_{t,j} = f(\theta_{t,j}) = f(\theta_{t,p}) \cdot \begin{bmatrix} \mathcal{R}(\theta_{t,j}) & \mathbf{L}_{j,p} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}, \tag{2}$$

where p is the parent of joint j in the kinematic chain, $\mathcal{R}$ transforms $\theta_{t,j}$ into a valid $\mathbf{R} \in SO(3)$ rotation matrix using Rodrigues' rotation formula, and $\mathbf{L}_{j,p}$ is the bone length between joints j and p in the first frame. Forcing the $\mathbf{L}$ translation vector to be from a single frame forces bone lengths to be the same for every frame.

We use gradient descent with the Adam [17] optimizer to optimise the objective function

$$\underset{\theta}{\text{minimize}} \sum_{t=1}^{T}\sum_{j=1}^{N}\left\|\widehat{\mathbf{J}}_{t,j} - \mathbf{J}_{t,j}\right\|_2 + \lambda L_S, \tag{3}$$

where L_S is an additional smoothness regulariser:

$$L_S = \left\|\widehat{\mathbf{J}}_{t,j} - \widehat{\mathbf{J}}_{t-1,j}\right\|_2. \tag{4}$$

The inverse-kinematics optimization provides a new set of joints $\widehat{\mathbf{J}}_{t,j}$ that ensures consistent bone lengths throughout any complex sequence of movement.

3.1.3 Perspective projection

We define an initial camera pose to satisfy two conditions. The mean location of the animal throughout the sequence is at the center of the camera view, and all joints are within view for the entirety of the sequence. We randomly rotate the camera around the y-axis and project the points to the 2D camera plane. For our purpose we do this only once for each animation sequence, however this process can be used to obtain many different views of the animal throughout the sequence.

3.2. Lifting model

Given an input sequence of 2D skeletons $\mathbf{X} \in \mathbb{R}^{T\times J\times 2}$, where T is the number of frames in the video and J is the number of joints, our goal is to reconstruct the 3D skeletons $\widehat{\mathbf{Y}} \in \mathbb{R}^{T\times J\times 3}$ of the object.

3.2.1 Keypoint features

The attention mechanism of transformers is inherently permutation equivariant, such that inputs can be randomly permuted and the corresponding outputs will remain the same. We leverage this property to handle objects with different joint configurations. We utilise the masking mechanism of [8] to overcome the technical challenge of training with different numbers of joints. Inputs are zero-padded up to the maximum number of joints in a mini-batch and a mask $\mathbf{M} \in \{0,1\}^J$ is used to ignore padded joints. Each element in $\mathbf{M}$ is defined as:

$$\mathbf{M}_i = \begin{cases} 1 & \text{if joint } i \text{ is present} \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

We encode the 2D skeletons into D-dimensional features $\mathbf{F} \in \mathbb{R}^{T\times J\times D}$ using Random Fourier Features (RFF) [34]. We additionally encode each 2D joint (x,y) with its temporal location t in the sequence. We thus compute the feature of an input $\mathbf{p} = [x,y,t]^T$ as:

$$\phi(\mathbf{p}) = \sqrt{\frac{2}{D}}\Big[\sin(\mathbf{W}\cdot\mathbf{p}+\mathbf{b}); \cos(\mathbf{W}\cdot\mathbf{p}+\mathbf{b})\Big], \tag{6}$$

where $\mathbf{W} \in \mathbb{R}^{\frac{D}{2}\times 3}$ is sampled from a normal distribution $\mathcal{N}(0,I)$ and $\mathbf{b} \in \mathbb{R}^{\frac{D}{2}}$ is sampled from a uniform distribution $\mathcal{U}(0,\frac{1}{2\pi})$. We choose analytical RFF for its success in low-data and out-of-distribution (OOD) scenarios [8, 34]. We find that encoding the temporal position is beneficial for the motion encoder in capturing temporal dependencies.

3.2.2 Motion encoder

The motion encoder leverages multi-head self-attention (MHSA) to embed temporal context into the keypoint features. Although we choose MHSA for its ability to use information from all frames, its lack of an inductive bias makes it unsuitable for tasks with limited data. We argue that it is not necessary to consider all frames in a sequence because most of the useful information can be found in nearby frames. We explicitly impose this inductive bias into the MHSA blocks by applying a binary mask to the intermediate attention maps. We define the mask $\mathbf{Z} \in \mathbb{R}^{T\times T}$ for a joint at time t as:

$$\mathbf{Z}_{t,i} = \begin{cases} 1 & \text{for } t-\alpha \le i \le t+\alpha \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where α is a hyper-parameter controlling the number of frames before and after t that can contribute information during attention.

Given a set of spatial features $\mathbf{F} \in \mathbb{R}^{T\times J\times D}$ as input to our motion encoder, we matrix transpose to get $\mathbf{F}_M \in \mathbb{R}^{J\times T\times D}$. We first apply linear projections to obtain queries $\mathbf{Q}$, keys $\mathbf{K}$, and values $\mathbf{V}$ for each head h:

$$\mathbf{Q}^{(h)} = \mathbf{F}\mathbf{W}_Q^{(h)}, \mathbf{K}^{(h)} = \mathbf{F}\mathbf{W}_K^{(h)}, \mathbf{V}^{(h)} = \mathbf{F}\mathbf{W}_V^{(h)}, \tag{8}$$

where $\mathbf{Q}^{(h)}, \mathbf{K}^{(h)}, \mathbf{V}^{(h)} \in \mathbb{R}^{N,T,\frac{D}{H}}$ for a total of H heads. We apply our temporal mask $\mathbf{Z}$ before the non-linear softmax operation in each head:

$$\text{head}_h = \text{softmax}(\frac{\mathbf{Q}^{(h)}(\mathbf{K}^{(h)})^{\mathbf{T}}}{\sigma} \times \mathbf{Z}) \cdot \mathbf{V}^{(h)}, \tag{9}$$

where σ is a scaling factor. Finally, each head is concatenated and projected:

$$\mathbf{F}_M = \text{MHSA}_\alpha(\mathbf{F}_M) = [head_1; ...; head_h]\mathbf{W}_{\mathbf{P}}. \tag{10}$$

We follow the standard procedure of applying a residual connection and layer normalisation before obtaining the final output of one windowed-MHSA layer. We stack P of these layers with residual connections to create our motion encoder.

3.2.3 Space encoder

The space encoder uses the features from our motion encoder to model the relationships among joints in a single frame. We found the hybrid graph-based approach of [8] to perform favorably for our task. Given the transposed motion features $\mathbf{F}_M \in \mathbb{R}^{T\times J\times D}$, a single space-layer has two simultaneous processing streams, one for capturing the local connectivity between joints G_{local}, and another for capturing global connectivity G_{global}. These two streams are concatenated and projected to provide an output of spatial features containing a combination of both streams:

$$\mathbf{F}_S = \text{MLP}([G_{\text{local}}(\mathbf{F}_M, \mathbf{A}); G_{\text{global}}(\mathbf{F}_M)]). \tag{11}$$

The local graph-attention G_{local} utilises an adjacency matrix of the joint connections $\mathbf{A} \in \mathbb{R}^{J \times J}$ to model the object connectivity. A layer normalisation and skip connection is applied to produce the final output of a layer. As with our motion encoder, we stack O layers with residual connections between them.

3.2.4 Decoder and Procrustes-based loss

Lastly, given the latent spatial features $\mathbf{F}_S$, we use an MLP to decode the predicted 3D structures of the object in a canonical 3D space

$$\widehat{\mathbf{Y}}^{\text{canon}} = \text{MLP}(\mathbf{F}_S). \tag{12}$$

We align each canonical prediction $\widehat{\mathbf{Y}}^{\text{canon}} \in \mathbb{R}^{T \times J \times 3}$ with the ground truth $\mathbf{Y}$ via a Procrustes alignment method which solves for the optimal rotation $\tilde{\mathbf{R}}_t$ individually for each frame t as:

$$\underset{\mathbf{R}_t \in SO(3)}{\text{minimize}} \left\| \mathbf{Y}_t - \widehat{\mathbf{Y}}_t^{\text{canon}} \mathbf{R}_t \right\|_2. \tag{13}$$

In practice, we use Singular Value Decomposition (SVD) to solve this optimisation problem. To ensure that $\tilde{\mathbf{R}}_t$ belongs to the special orthogonal group $SO(3)$, we enforce $\det(\tilde{\mathbf{R}}_t)= +1$. This step is crucial for mitigating reflection ambiguity in our predictions.

The resulting $\tilde{\mathbf{R}}_t$ is used to align our canonical predictions with the ground truth. We additionally scale the predictions relative to the ground truth using a scaling factor $\mathbf{s} \in \mathbb{R}^T$:

$$\widehat{\mathbf{Y}}_t = \mathbf{s}_t \cdot (\widehat{\mathbf{Y}}_t^{\text{canon}} \tilde{\mathbf{R}}_t) \tag{14}$$

3.2.5 Loss function

With our predictions now aligned with the ground truth, we can compute our loss. We compute the Mean Squared Error (MSE) of the 3D points along with an additional velocity error $\mathcal{L}_{\text{vel}}$:

$$\mathcal{L}_{\text{total}} = \sum_{t=1}^{T} \sum_{j=1}^{J} \left\| \mathbf{Y}_{t,j} - \widehat{\mathbf{Y}}_{t,j} \right\|_2 + \lambda \mathcal{L}_{\text{vel}}, \tag{15}$$

$$\mathcal{L}_{\text{vel}} = \sum_{t=2}^{T} \sum_{j=1}^{J} \left\| (\mathbf{Y}_{t,j} - \mathbf{Y}_{t-1,j}) - (\widehat{\mathbf{Y}}_{t,j} - \widehat{\mathbf{Y}}_{t-1,j}) \right\|_2. \tag{16}$$

We use the scalar λ to weight the velocity loss.

4. Experiments

We evaluate our method on various animal categories to assess its performance and generalisation properties. Comparative analyses are with recent state-of-the-art video (MotionBERT [36]) and single-frame (3D-LFM [8]) lifting models on various animal categories and motion sequences.

Datasets We use the AnimalSyn3D dataset as described in Sec. 3.1. The 2D keypoints provided by off-the-shelf pose detectors [32] are inherently noisy due to factors such as lighting conditions and image quality. To simulate these conditions, we synthetically perturb the 2D keypoints with an additive Gaussian noise, which corresponds to a 3-pixel error on average. We present results for non-noisy data (Tab. 8) and also provide comparisons on 3D human pose estimation in the supplementary material (Tab. 11), although human-specific lifting is not the focus of our work. We normalise 2D keypoints and 3D labels to $[-1, 1]$, after scaling the 3D labels following existing works on 3D human pose estimation [36]. We split the data for training by randomly selecting 80% of the animation sequences for each animal, with the remaining animations withheld for testing.

Evaluation protocols Previous video lifting methods [5, 12, 21, 36] evaluate the non-rigid structure and motion of their approach by calculating the mean per-joint position error (MPJPE) directly with the ground truth in the *camera space*. However, our model and 3D-LFM make predictions in a *canonical space* that requires the alignment of each frame to the ground truth. As such, we instead use the standard *per-frame* Procrustes-aligned MPJPE metric and refer to it as the frame-aligned MPJPE **(FA-MPJPE)** for brevity. Additionally, we compose a metric to measure the relative motion error in a video sequence.

Sequence-Aligned MPJPE We formulate the sequence-aligned MPJPE **(SA-MPJPE)** as solving for a *single* rotation matrix $\mathbf{R} \in SO(3)$ to align the 3D predictions $\widehat{\mathbf{Y}}$ and ground truth $\mathbf{Y}$ for all T frames in a sequence. This is in contrast to the SA-MPJPE that aligns each individual frame in a sequence by solving for T rotation matrices. After alignment, we compute the MSE to produce the final error value. Let us specify $\widehat{\mathbf{Y}}, \mathbf{Y} \in \mathbb{R}^{T \times J \times 3}$, for J joints, to define our metric as

$$\underset{\mathbf{R}}{\text{minimize}} \sum_{t=1}^{T} \left\| \mathbf{Y}_t - \widehat{\mathbf{Y}}_t \mathbf{R} \right\|_2. \tag{17}$$

Compared to Eq. (13), we are instead solving for a single, global $\mathbf{R}$. Our metric thus captures any error in the motion that occurs between subsequent frames of a sequence.

Method	Bear	Buck	Bunny	Chicken	Deer	Dog	Elk	Fox	Moose	Puma	Rabbit	Raccoon	Tiger	Avg
MotionBERT	94.5	208.1	**16.7**	108.2	200.7	50.1	267.4	40.6	189.2	254.4	30.7	77.4	211.8	134.6
3D-LFM	47.6	158.2	23.2	92.3	156.8	53.9	147.8	22.2	274.7	163.4	37.8	70.0	165.4	108.7
Ours	**29.2**	**128.4**	17.1	**60.8**	**57.3**	**32.8**	**103.1**	**14.2**	**97.9**	**93.2**	**19.0**	**44.5**	**90.8**	**60.6**
MotionBERT	90.7	198.1	16.0	99.0	195.5	45.8	246.8	39.9	170.9	235.0	28.6	74.9	203.2	126.5
3D-LFM	27.9	108.3	12.2	86.3	75.0	33.3	103.0	16.2	119.7	119.3	21.2	57.6	107.7	68.3
Ours	**26.7**	**107.3**	**11.2**	**54.2**	**50.9**	**27.9**	**86.1**	**12.4**	**81.6**	**85.9**	**15.4**	**42.8**	**79.8**	**52.5**
MotionBERT	3.2	**11.0**	**1.1**	3.9	6.9	**2.6**	10.4	1.3	17.8	9.8	2.1	4.0	9.8	6.5
3D-LFM	7.6	29.0	3.4	8.4	26.3	8.4	26.5	3.8	43.4	27.3	6.7	12.3	30.4	18.0
Ours	**2.5**	12.1	1.3	**3.4**	**5.9**	2.9	**9.3**	**1.2**	**12.5**	**8.9**	**2.0**	**3.6**	**9.1**	**5.7**

Table 1. **Quantitative comparison of 2D to 3D lifting with 13 animals.** We report, in millimeters, the Sequence-Aligned MPJPE (top), Frame-Aligned MPJPE (middle), and Sequence-Aligned MPVE (bottom), see Sec. 4 for details of these evaluation metrics. Our approach (Ours) outperforms existing state-of-the-arts with significant gap across multiple animal categories.

Method	MC	FA-MPJPE↓	SA-MPJPE↓	SA-MPVE↓
MotionBERT	-	176.0	199.9	9.78
	✓	**126.5**	**134.6**	**6.5**
3D-LFM	-	89.2	126.4	20.5
	✓	**68.3**	**108.7**	**18.0**
Ours	-	105.4	128.4	11.0
	✓	**52.5**	**60.6**	**5.7**

Table 2. **Quantitative comparison between multi-category (MC) and single-category training.** We use a ✓for models trained with multi-category training. Each method benefits from multi-category training. See supplementary material for a breakdown of per-animal results.

Lastly, we are able to report the commonly used mean per-joint velocity error (MPVE) after performing the global sequence alignment. For clarity in our comparisons, we refer to this as **SA-MPVE**.

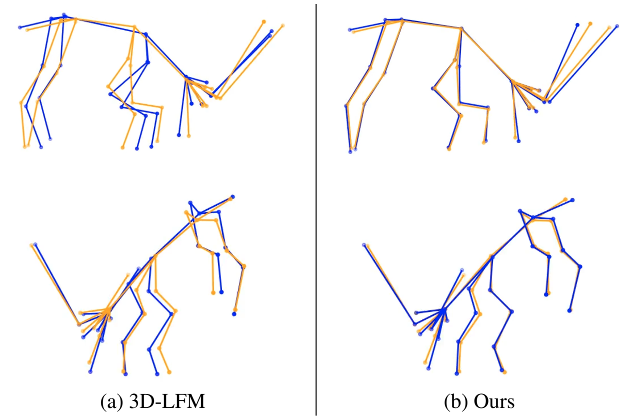
Implementation details Here we provide important implementation details of our method and refer the reader to the supplementary material for further details. We construct batches of 32 sequences with 48 frames per sequence. We train on a total of 871 video sequences and evaluate on a separate set of 199 unseen sequences across all 13 animal categories. Inputs are zero-padded up to the maximum of 29 joints that occur in dataset. A layer size of $P = 4$ is chosen for the motion encoder and $O = 12$ for the space encoder. The hidden-dimension size D is 256. Experiments were conducted on a single NVIDIA A100 GPU.

MotionBERT uses human-specific semantic knowledge, making it unsuited for object-agnostic lifting. To provide a fair comparison, we do not modify the proposed architecture and instead apply two alternative alterations. We first set the number of learned positional embeddings to the maximum amount of joints seen in the dataset, allowing it to handle all animal rigs. We also randomly permute the 2D inputs during training and testing to simulate a real object-agnostic scenario. This is required so that the positional embeddings are not learning dataset-specific skeleton semantics. We ensure that joints being permuted retain their temporal correspondence over a sequence.

4.1. Object-agnostic lifting

Tab. 1 demonstrates the effectiveness of our method compared to MotionBERT and 3D-LFM. We outperform both methods across all three metrics and nearly every animal. Notably, our approach achieves 45% lower SA-MPJPE and 70% lower SA-MPVE compared to 3D-LFM, demonstrating that our predicted 3D motion has more accurate and smoother movement, while also preserving high-fidelity object structure (FA-MPJPE). This substantial performance gap highlights the critical role of the motion encoder in capturing the temporal relationship of joints. While MotionBERT is a spatio-temporal approach, it has no inductive bias to assist with handling scarce data and multi-category training. Fig. 3 qualitatively demonstrates the predictions of our method compared to 3D-LFM.

Single- vs multi-category We evaluate performance on single-category training and compare it to their performance on multi-category training. For single-category, we train each approach from scratch using data specific to a single animal category. This process is repeated for all 13 animals, and the mean error is reported in Tab. 2. All methods show significant improvement when performing multi-category training as opposed to single-category, achieving at least a 25% reduction in FA-MPJPE, with our approach achieving a 50% reduction. This improvement highlights the advantage of unified learning across a vast spectrum of object categories, particularly in scenarios where the training data is small and unbalanced.


(a) 3D-LFM

(b) Ours

Figure 3. **Quantitative comparison on a *Deer* sequence from two different views**: Our method provides significantly more accurate 3D predictions. In this visualization, blue represents the predicted 3D points whereas the orange denotes the ground truth.

Robustness to occlusion To evaluate the robustness of 3D-LFM and our approach to occlusion scenarios, we trained both models by randomly masking 10% of all 2D keypoints within a frame and tasking the models with recovering the 3D locations of the missing joints. As shown in Fig. 6, our method demonstrates superior robustness while 3D-LFM struggles. Even in an extreme case of 60% occlusion we continue to see legible predictions from our method; see Fig. 8 in the supplementary material for a qualitative result.

4.2. OOD generalization

Unseen objects We perform a 13-fold analysis, where each fold involves holding out one animal category from the original dataset during training. For example, the *bunny* category is excluded from the training data and used to evaluate the generalization capability of each method. As shown in Fig. 4 (left), our approach demonstrates superior OOD generalization when handling unseen animal categories by outperforming existing methods by a significant margin. We present a qualitative reconstruction for a *bunny* instance in Fig. 5 and refer the reader to Tab. 9 in the supplementary material for tabulated results.

Rig transfer When lifting an unseen animal in the wild, we may encounter an animal with a more complex structure than seen during training. We showcase our ability to generalize to an unseen animal with an unseen number of joints. While MotionBERT is limited to rigs with the same or fewer joints as those seen during training, our method can handle *any* number of joints. We train on animal rigs with 27 or fewer joints and test on two *unseen* animals with 29 joints (deer and moose). As shown in Fig. 4 (right), while both 3D-LFM and our approach are impacted by the difficulty of the task, our approach outperforms 3D-LFM by at least 40% in all metrics. We show the tabulated results in the supplementary material (Tab. 10).

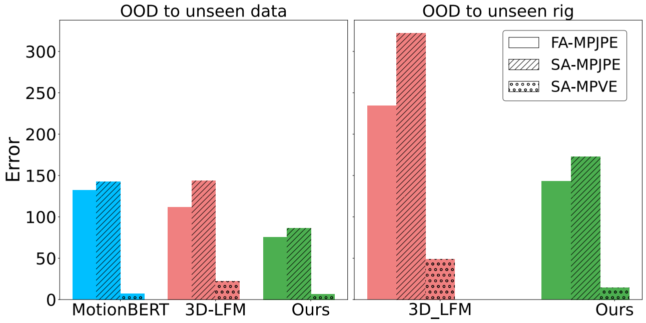
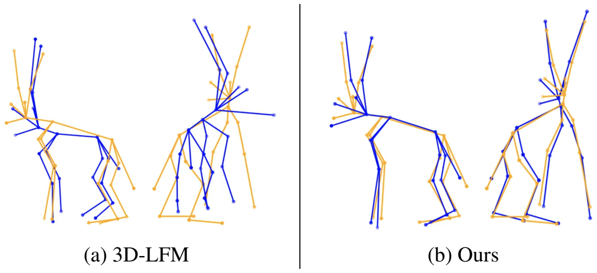
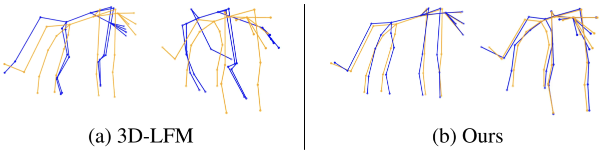


Figure 4. **OOD generalization**. *OOD to unseen data* (left): We perform a 13-fold evaluation to assess each method's ability to handle unseen animal categories. *OOD to an unseen category and rig* (right): Note that MotionBert is constrained to rigs with the same or fewer joints as those seen during training and hence cannot handle unseen rigs with more joints. Our method can handle generalization to both unseen category and unseen rig more effectively.


(a) 3D-LFM

(b) Ours

Figure 5. **OOD generalization on an unseen *Bunny* category from two different views**: Our method provides significantly more accurate 3D predictions compared to 3D-LFM. In this visualization, blue represents the predicted 3D points whereas the orange denotes the ground truth.


(a) 3D-LFM

(b) Ours

Figure 6. **A comparison of robustness when 10% of a tiger is occluded.** We include the average frame-aligned MPJPE across all animals for each method. Two views of the object are shown.

4.3. Ablations

In this section we ablate the important building blocks of our approach. We first highlight the importance of our spatio-temporal approach as opposed to a spatial-only approach. Then, we demonstrate the improvement gained from our temporal-proximity inductive bias. We go on to

Model	FA-MPJPE↓	SA-MPJPE↓	SA-MPVE↓
Space	67.8	99.9	13.1
Time	**57.9**	**66.5**	**5.6**

Table 3. **Ablation of time vs. space.**

α	FA-MPJPE ↓	SA-MPJPE ↓	SA-MPVE ↓
2	$53.9_{\pm 1.12}$	$65.5_{\pm 0.96}$	$7.0_{\pm 0.02}$
4	$54.4_{\pm 1.13}$	$64.9_{\pm 2.78}$	$6.4_{\pm 0.01}$
8	$\mathbf{52.7}_{\pm 0.19}$	$\mathbf{61.4}_{\pm 0.40}$	$5.8_{\pm 0.00}$
16	$57.1_{\pm 0.93}$	$65.6_{\pm 1.70}$	$5.7_{\pm 0.02}$
-	$57.9_{\pm 0.33}$	$66.5_{\pm 0.88}$	$\mathbf{5.6}_{\pm 0.01}$

Table 4. **Ablation of our inductive bias**. We use $-$ to denote no constraint being applied. We report the averages over 5 independent runs.

compare strategies for encoding the position of a joint in time. Lastly, we ablate the importance of our procrustes-based training.

Time vs. space We evaluate the significance of information sharing across time facilitated by our proposed motion encoder. To isolate the impact of time, we create a space-only variant of our model by replacing the motion encoder with additional space blocks such that the number of parameters remains similar. As shown in Tab. 3, modelling temporal dependencies is crucial for enhancing 2D-3D lifting performance, particularly in terms of per-sequence reconstruction accuracy (SA-MPJPE) and smoothness (SA-MPVE).

Constrained temporal attention We observe that applying our inductive bias to restrict information sharing enhances the optimization of our model. We conduct our ablation by progressively increasing the window size α. We apply α according to Eq. (7). Tab. 4 shows the impact of different α on performance, averaged over five statistical runs. Overall, we observe a trade-off between the accuracy of the 3D structure and the accuracy of 3D motion. In our case, we identify $\alpha = 8$ as optimal, which we used for all of our other experiments.

Analytical temporal embedding Here we ablate our use of analytical RFF to encode temporal information. In Tab. 5, we show that it is beneficial to use an analytical embedding over a learned embedding. Analytical RFF provides a significant performance increase over a learned embedding, in our case it seems to result from a scarcity of data.

Temporal Embedding	FA-MPJPE↓	SA-MPJPE↓	SA-MPVE↓
-	$55.5_{\pm 2.29}$	$64.8_{\pm 3.04}$	$5.9_{\pm 0.03}$
Learned	$54.0_{\pm 1.13}$	$64.1_{\pm 1.24}$	$6.1_{\pm 0.02}$
Analytical	$\mathbf{52.7}_{\pm 0.25}$	$\mathbf{61.4}_{\pm 0.27}$	$\mathbf{5.8}_{\pm 0.01}$

Table 5. **Ablation of temporal embedding strategies.** We report the averages over 5 independent runs.

Model	Procrustes	FA-MPJPE↓	SA-MPJPE↓	SA-MPVE↓
3D-LFM	-	83.2	97.3	25.0
3D-LFM	✓	68.3	108.7	18.0
Ours	-	62.2	74.9	7.7
Ours	✓	**52.7**	**61.4**	**5.6**

Table 6. **Procrustes vs. non-Procrustes training**. Our spatio-temporal 2D-3D lifting approach with Procrustean alignment surpasses 3D-LFM.

Procrustes-based training Similar to 3D-LFM, we find it useful to train our model with a Procrustes-based loss, as shown in Tab. 6. Allowing the model to focus solely on learning object structure significantly enhances the predicted 3D structure and benefits overall 3D motion accuracy. Interestingly, we observe that we are able to perform well without the Procrustes-based loss, even outperforming 3D-LFM across all metrics. This offers an interesting insight into future work involving our method for practical applications that require implicitly predicted camera rotation.

5. Conclusion

In this work, we introduced an object-agnostic 3D lifting model that leverages a temporal inductive bias for temporal sequences. Our approach sets a new benchmark in lifting performance for object categories with limited available data. The model's ability to generalize across unseen object categories and rigs shows its versatility and robustness, even in challenging scenarios involving noise and occlusions. In addition to these contributions, we have introduced a new synthetic dataset, AnimalSyn3D, designed to stimulate further research in class-agnostic 3D lifting models. This work aims to pave the way for more generalized and efficient 3D reconstruction methods that can be applied across diverse real-world applications.

6. Acknowledgements

We would like to thank Angel Allen for contributing the dog and raccoon to the dataset, Hemanth Saratchandran for polishing the mathematical notations, Lachlan Mares and Stefan Podgorski for notes during drafting, and Chee-Keng Ch'ng for helpful discussions throughout the project.

References

[1] Liang An, Jilong Ren, Tao Yu, Tang Hai, Yichang Jia, and Yebin Liu. Three-dimensional surface motion capture of multiple freely moving pigs using MAMMAL. *Nature Communications*, 14(1):7727, 2023. 2

[2] Ernesto Brau and Hao Jiang. 3D Human Pose Estimation via Deep Learning from 2D Annotations. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 582–591, 2016. 2, 1

[3] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3D shape from image streams. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No.PR00662)*, pages 690–696 vol.2, 2000. 1

[4] Ju Yong Chang, Gyeongsik Moon, and Kyoung Mu Lee. PoseLifter: Absolute 3D human pose lifting network from a single noisy 2D human pose, 2020. 2

[5] Tianlang Chen, Chen Fang, Xiaohui Shen, Yiheng Zhu, Zhili Chen, and Jiebo Luo. Anatomy-aware 3D Human Pose Estimation with Bone-based Pose Decomposition, 2021. 2, 5

[6] Zheng Chen and Yi Sun. Joint-wise 2D to 3D lifting for hand pose estimation from a single RGB image. *Applied Intelligence*, 53(6):6421–6431, 2023. 2

[7] Chaoqun Cheng, Zijian Huang, Ruiming Zhang, Guozheng Huang, Han Wang, Likai Tang, and Xiaoqin Wang. MarmoPose: A Deep Learning-Based System for Real-time Multi-Marmoset 3D Pose Tracking, 2024. 2

[8] Mosam Dabhi, Laszlo A. Jeni, and Simon Lucey. 3D-LFM: Lifting Foundation Model, 2024. 1, 2, 4, 5

[9] Carl Doersch, Ankush Gupta, Larisa Markeeva, Adrià Recasens, Lucas Smaira, Yusuf Aytar, João Carreira, Andrew Zisserman, and Yi Yang. TAP-Vid: A Benchmark for Tracking Any Point in a Video, 2023. 2

[10] Liuhao Ge, Zhou Ren, Yuncheng Li, Zehao Xue, Yingying Wang, Jianfei Cai, and Junsong Yuan. 3D Hand Shape and Pose Estimation from a Single RGB Image, 2019. 2

[11] Adam Gosztolai, Semih Günel, Marco Pietro Abrate, Daniel Morales, Victor Lobato Ríos, Helge Rhodin, Pascal Fua, and Pavan Ramdya. LiftPose3D, a deep learning-based approach for transforming 2D to 3D pose in laboratory animals, 2020. 2

[12] Mir Rayat Imtiaz Hossain and James J. Little. Exploiting Temporal Information for 3D Human Pose Estimation. In *Computer Vision – ECCV 2018*, pages 69–86. Springer International Publishing, Cham, 2018. 2, 5

[13] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, 2014. 1

[14] Haorui Ji, Hui Deng, Yuchao Dai, and Hongdong Li. Unsupervised 3D Pose Estimation with Non-Rigid Structure-from-Motion Modeling, 2023. 2

[15] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. DynamicStereo: Consistent Dynamic Depth from Stereo Videos. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13229–13239, Vancouver, BC, Canada, 2023. IEEE. 2

[16] Pierre Karashchuk, Katie L. Rupp, Evyn S. Dickinson, Sarah Walling-Bell, Elischa Sanders, Eiman Azim, Bingni W. Brunton, and John C. Tuthill. Anipose: A toolkit for robust markerless 3D pose estimation. *Cell Reports*, 36(13), 2021. 2

[17] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, 2017. 3, 1

[18] Chen Kong and Simon Lucey. Deep Non-Rigid Structure From Motion. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1558–1567, Seoul, Korea (South), 2019. IEEE. 2

[19] Ci Li, Ylva Mellbin, Johanna Krogager, Senya Polikovsky, Martin Holmberg, Nima Ghorbani, Michael J. Black, Hedvig Kjellström, Silvia Zuffi, and Elin Hernlund. The Poses for Equine Research Dataset (PFERD). *Scientific Data*, 11(1): 497, 2024. 2

[20] Yang Li, Hikari Takehara, Takafumi Taketomi, Bo Zheng, and Matthias Nießner. 4DComplete: Non-Rigid Motion Estimation Beyond the Observable Surface. https://arxiv.org/abs/2105.01905v1, 2021. 2, 3

[21] Ruixu Liu, Ju Shen, He Wang, Chen Chen, Sen-ching Cheung, and Vijayan Asari. Attention Mechanism Exploits Temporal Contexts: Real-Time 3D Human Pose Reconstruction. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5063–5072, Seattle, WA, USA, 2020. IEEE. 2, 5

[22] Alexander Mathis, Pranav Mamidanna, Taiga Abe, Kevin M. Cury, Venkatesh N. Murthy, Mackenzie W. Mathis, and Matthias Bethge. Markerless tracking of user-defined features with deep learning, 2018. 2

[23] Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Camera Distance-aware Top-down Approach for 3D Multi-person Pose Estimation from a Single RGB Image, 2019. 2

[24] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked Hourglass Networks for Human Pose Estimation, 2016. 1

[25] David Novotny, Nikhila Ravi, Benjamin Graham, Natalia Neverova, and Andrea Vedaldi. C3DPO: Canonical 3D Pose Networks for Non-Rigid Structure From Motion, 2019. 2

[26] Sungheon Park, Jihye Hwang, and Nojun Kwak. 3D Human Pose Estimation Using Convolutional Neural Networks with 2D Pose Information, 2016. 2

[27] Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. Ordinal Depth Supervision for 3D Human Pose Estimation, 2018. 2

[28] Fabian Plum, René Bulla, Hendrik K. Beck, Natalie Imirzian, and David Labonte. replicAnt: A pipeline for generating annotated images of animals in complex environments using Unreal Engine. *Nature Communications*, 14(1): 7195, 2023. 2

[29] Carlo Tomasi and Takeo Kanade. Shape and motion from image streams under orthography: A factorization method. *International Journal of Computer Vision*, 9(2):137–154, 1992. 1

[30] Chaoyang Wang and Simon Lucey. PAUL: Procrustean Autoencoder for Unsupervised Lifting. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 434–443, Nashville, TN, USA, 2021. IEEE. 2

[31] Jiacong Xu, Yi Zhang, Jiawei Peng, Wufei Ma, Artur Jesslen, Pengliang Ji, Qixin Hu, Jiehua Zhang, Qihao Liu, Jiahao Wang, Wei Ji, Chen Wang, Xiaoding Yuan, Prakhar Kaushik, Guofeng Zhang, Jie Liu, Yushan Xie, Yawen Cui, Alan Yuille, and Adam Kortylewski. Animal3D: A Comprehensive Dataset of 3D Animal Pose and Shape, 2024. 2

[32] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. ViTPose: Simple Vision Transformer Baselines for Human Pose Estimation, 2022. 5

[33] Yuan Yao, Praneet Bala, Abhiraj Mohan, Eliza Bliss-Moreau, Kristine Coleman, Sienna M. Freeman, Christopher J. Machado, Jessica Raper, Jan Zimmermann, Benjamin Y. Hayden, and Hyun Soo Park. OpenMonkeyChallenge: Dataset and Benchmark Challenges for Pose Estimation of Non-human Primates. *International journal of computer vision*, 131(1):243–258, 2023. 2

[34] Jianqiao Zheng, Xueqian Li, Sameera Ramasinghe, and Simon Lucey. Robust Point Cloud Processing through Positional Embedding, 2023. 4

[35] Yang Zheng, Adam W. Harley, Bokui Shen, Gordon Wetzstein, and Leonidas J. Guibas. PointOdyssey: A Large-Scale Synthetic Dataset for Long-Term Point Tracking, 2023. 2

[36] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. MotionBERT: A Unified Perspective on Learning Human Motion Representations, 2023. 1, 2, 5