2022

Learned structural and temporal context for dynamic 3D pose optimization and tracking

BOSTON UNIVERSITY

GRADUATE SCHOOL OF ARTS AND SCIENCES

Thesis

LEARNED STRUCTURAL AND TEMPORAL CONTEXT FOR DYNAMIC 3D POSE OPTIMIZATION AND TRACKING

by

MAHIR PATEL

B.E., Gujarat Technological University, 2018

Submitted in partial fulfillment of the

requirements for the degree of

Master of Science

2022

Approved by

First Reader ____

Margrit Betke, Ph.D.
Professor of Computer Science

Second Reader ____

Michael E. Hasselmo, D.Phil.
William Fairfield Warren Distinguished Professor
Professor of Psychological and Brain Sciences

Third Reader ____

Sarah A. Bargal, Ph.D.
Research Assistant Professor of Computer Science

Errors using inadequate data are much less than those using no data at all

— Charles Babbage

Acknowledgments

I want to thank my advisors Prof. Margrit Betke and Prof. Michael Hasselmo, for their invaluable guidance. They provided immense support and advice during an unprecedented academic environment. They instilled the confidence required to complete this work. I would also like to thank them for the opportunity to work on the Neuro-Autonomous MURI. I would also like to thank Prof. Sarah Adel Bargal, her machine learning course was the most engaging and helpful course I have ever taken. It was the foundation that bolstered my academic journey. I sincerely hope to walk in their footsteps someday.

I would also like to thank Yiwen Gu and Lucas Carstensen, Ph.D. students in Computer Science and Neuroscience departments. In addition to their crucial support in research, I also appreciate their perspective and insights on what a doctoral program entails. It has prepared me better for my future endeavors.

Finally, I would like to thank my parents, brother, and friends for their unwavering love and support. I lack the words to describe their importance in my life.

This work was supported by the Center for Systems Neuroscience at Boston University under ONR MURI grant N00014-19-1-2571.

LEARNED STRUCTURAL AND TEMPORAL CONTEXT FOR DYNAMIC 3D POSE OPTIMIZATION AND TRACKING

MAHIR PATEL

Boston University, Graduate School of Arts and Sciences, 2022

Major Professor: Margrit Betke, PhD
Professor of Computer Science

ABSTRACT

Accurate 3D tracking of animals from video recordings is critical for many behavioral studies. However, other than for humans, there is a lack of publicly available datasets of videos of animals that the computer vision community could use for model development. Furthermore, due to occlusion and the uncontrollable nature of the animals, existing pose estimation models suffer from inadequate precision. People rely on biomechanical expertise to design mathematical models to optimize poses to mitigate this issue at the cost of generalization. We propose OptiPose, a generalizable attention-based deep learning pose optimization model, as a part of a post-processing pipeline for refining 3D poses estimated by pre-existing systems. Our experiments show how OptiPose is highly robust to noise and occlusion and can be used to optimize pose sequences provided by state-of-the-art models for animal pose estimation. Furthermore, we will make Rodent3D, a multimodal (RGB, Thermal, and Depth) dataset for rats, publicly available.

Contents

List of Tables

List of Figures

List of Abbreviations

CM	....	Context Models
DLT	....	Direct Linear Transformation
FTE	....	Full Trajectory Estimation
HSDR	....	High Speed Data Recorder
NLP	....	Natural Language Processing
PCK	....	Percentage of Correct Keypoint
PCM	....	Parallel Context Models
PreLU	....	Parametric Rectified Linear Unit
Sub-CM	....	Sub-Context Models
TTL	....	Transitor-Transistor Logic

Chapter 1

Introduction

The precise quantification of animal behaviors, based on their position, orientation, and movement of their head, body, or limbs, is critical for neuroscience, ecology, and psychology studies. Computer vision has long been a facilitator of such studies, initially limited to two-dimensional analysis of single videos of rodents and flies in laboratory settings [2, 27, 20, 11], then providing three-dimensional (3D) behavior analysis of wild animals in the field, for example, 3D flight paths of bats and birds [31, 33, 34], and eventually using deep learning models to estimate the 3D pose of animals, small and large, caged and wild [10, 15, 23]. While DeepLabCut [23] is likely the most widely used state-of-the-art software for estimating and tracking 3D keypoints on animals in laboratory settings, other approaches are emerging to improve the performance and the accuracy of keypoint detection and tracking [10, 18, 28]. Recently people began to include the postural dynamics of the animal either through manual constraints from biomechanical experts [15, 35] or through deep learned geometric reasoning [8]. Using postural dynamics for pose estimation outperforms the 2D markerless systems like DeepLabCut, especially in unfavorable conditions. Furthermore, deep-learned structural reasoning has a clear advantage of generalizability over hand-crafted constraint-based methods. One could argue that these approaches lack an understanding of the posture at a temporal scale. Our work aims to address this issue.

Since the emergence of Transformer models in the field of Natural Language Pro-

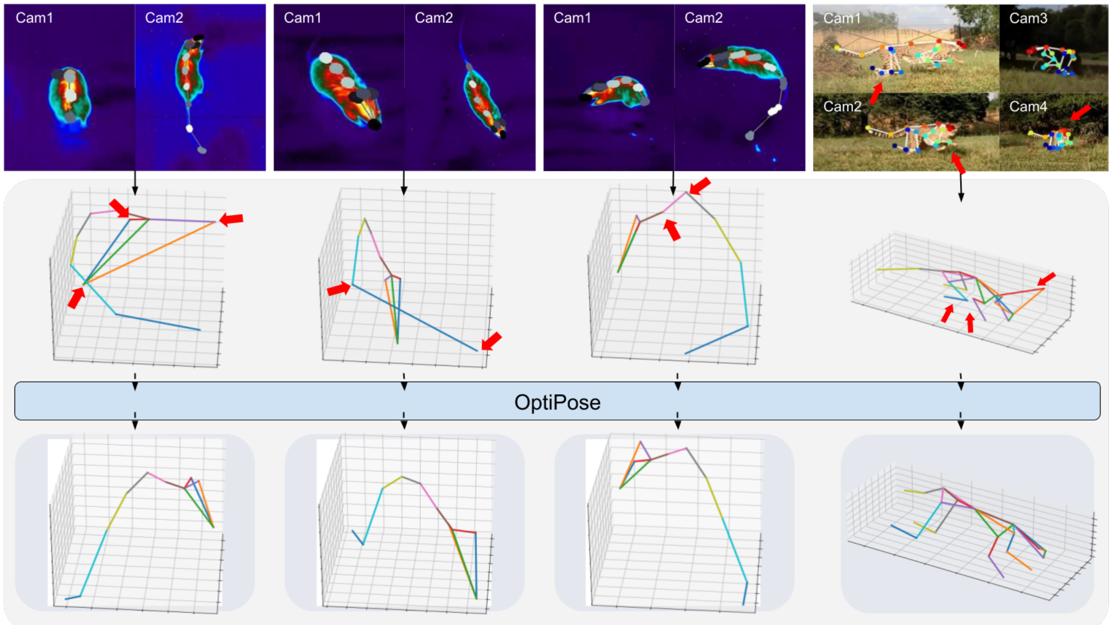


Figure 1·1: OptiPose Task: Improving the tracks of 3D keypoint estimates. Red arrows indicate inaccurate keypoint estimates due to occlusion, inaccurate detection by the 2D keypoint extractor, or errors introduced in the 3D reconstruction.

cessing, it is the consensus that attention-based architectures rapidly became state-of-the-art for learning the relations among the words and, in turn, acquiring a crude understanding of the underlying grammar. Therefore, we reframed the pose optimization problem as an NLP Task. Our model treats each pose as a token, and the keypoints collectively form its embedding in a high-dimensional space. Specifically, given T consecutive sets of N keypoints on an animal of interest, its poses over a period of time, $P_{1..T}$, can be defined by an embedding vector in $\mathbb{R}^{T\times 3N}$ space. Thus, an activity, defined as a sequence $P_{1..T}$ of poses, can be considered on par with a sentence in a natural language. Different permutations of pose tokens form different activities. These permutations are defined by the movement pattern of the animal, which could be interpreted as the grammar of the language. We designed OptiPose as a spatio-temporal self-attention model that outputs a sequence of 3D animal pose

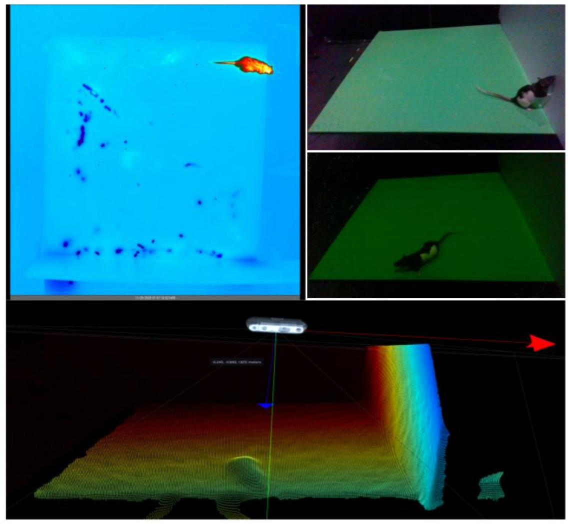


Figure 1·2: The proposed publicly available Rodent3D dataset contains annotated multimodal videos, including thermal infrared (top left), regular and low-light RGB (top right), and depth (bottom).

estimates defined by 3D keypoints located on the surface of the animal (see Fig. 1·1). Its input is a potentially highly noisy version of a 3D keypoint sequence. OptiPose can handle significant occlusion of keypoints (Fig. 1·1, left), label inaccuracies (Fig. 1·1, middle), and 3D reconstruction inaccuracies (Fig. 1·1, right). It can be employed in combination with any keypoint tracker, e.g., DeepLabCut [23], as a second stage in a two-stage 3D pose tracking system.

In our work, we developed a high refresh-rate multi-camera system to record rodents in a laboratory setting, using three modalities – RGB, depth, and thermal infrared (see Fig. 1·2) – to produce ground truth labels of the 3D pose. Thermal infrared cameras or color cameras at low light are preferred over color cameras in bright light for recording rodent behavior in neuroscience laboratories to avoid unnatural behavior due to light distraction by these nocturnal creatures. We have collected and curated an animal pose estimation dataset, called Rodent3D, that we made publicly available so that others can conduct that benchmark experiments with single or multiple modalities. Finally, we will release our highly configurable software that executes our pipeline and performs behavioral analysis.

Chapter 2

Literature Review

This chapter reviews the previous work on 3D pose estimation and attention-based sequence-to-sequence models. For 3D pose estimation, we will focus on the approaches that exploit kinematics or postural information to optimize the initial coarse predictions. Finally, we will also discuss the pose estimation datasets available for the animals.

2.1 3D Animal Pose Estimation

General approaches for 3D pose estimation can be categorized into two broad techniques: One set of approaches use classical photogrammetric methods on intermediate 2D markers obtained through deep convolutional neural networks [23, 11, 27, 28, 15, 16, 12]. The other set of approaches uses a volumetric convolutional network to predict 3D poses directly [8, 24, 14, 26]. Developing robust volumetric networks requires accurate 3D annotations, which are difficult to obtain without expensive recording hardware like motion capture systems. On the other hand, using classic multi-view reconstruction of keypoints is prone to noise due to its multi-stage reconstruction pipeline. In that, both the keypoint detection and reconstruction could add unnecessary noise. Furthermore, both techniques are inclined to have incorrect predictions due to occlusion or lighting conditions. To reduce noise in keypoint detection, DeepFly3D [12] uses pictorial patterns [9] to divide the 2D skeleton into different parts and minimize the region of predictions. Tools like Anipose [16] apply spatio-temporal filtering to alleviate marker noise and perform robust triangulation. However, such filtering cannot handle long-term occlusions. Researchers have addressed this issue by incorporating postural information in the model or as a post-processing function. Approaches such as GIMBAL [35], FTE [15] that incorporate the biomechanics of the subject can be used for improving 3D estimates of poses reconstructed with geometric triangulation while handling long term occlusions. However, these methods require biomechanical expertise and manual feature definitions. Consequently, they are not cross-species. On the other hand, a deep-learned volumetric network is bound to learn the spatial postural patterns but collecting accurate 3D ground truth is very expensive. Furthermore, learning spatial biomechanical patterns is not enough; leveraging temporal understanding is equally important.

2.2 Attention-based Sequence-To-Sequence Models

Recent advancements in Natural Language Processing are primarily attributed to the type of attention mechanism used in the Transformer model family [32, 7, 4, 3, 29]. These models are better than the previous recurrent neural networks as they are non-sequential and instead use clever positional embedding to encode sequential relations. Recent large-scale models [3, 29] utilizing these methods have exhibited a realistic understanding of the language. Therefore, by looking at the poses as words and the performed activities as sentences, we can use these methods to understand the underlying "grammar" of the subject's behavior. Recent work [36] proposes a transformer-based model that inputs raw images and generates a 3D pose that successfully outperforms state-of-the-art pose-estimation models. However, it is not a sequence-to-sequence model and generates a single 3D pose from a sequence of inputs. Furthermore, since it operates directly on images, it can not easily take advantage of the augmentation methods described in Sec.4.4.

2.3 Datasets

The pose estimation datasets for the animals are not as mature as the human datasets. Researchers have recently begun to release large-scale video datasets for various animals [8, 15, 22]. The AcinoSet (Cheetah) dataset includes sequences of a running cheetah from six views recorded with high-speed cameras. It provides approximately 7,500 hand-annotated frames with 20 keypoints. It also includes 3D reconstructed poses for 119,490 frames, computed by Full Trajectory Estimation (FTE), which can be used as ground truths [15]. The Rat7M [8] dataset includes 10.8 hours of motion capture and color video data, almost 7 million frames total. The motion capture data was obtained by attaching markers to 20 sites on a rat's head, trunk, and limbs using body piercings. The RGB videos involve up to 6 synchronized camera views, operating at 30 Hz under day-time light conditions. The PAIR-R24M [22] provides 24 million frames of multi-animal RGB videos and motion-captured 3D pose annotations. These existing datasets have a large number of frames, but in contrast to our proposed dataset Rodent3D, they do not offer multiple imaging modalities.

Chapter 3

Rodent3D: Data Acquisition

3.1 Configuration and Synchronization

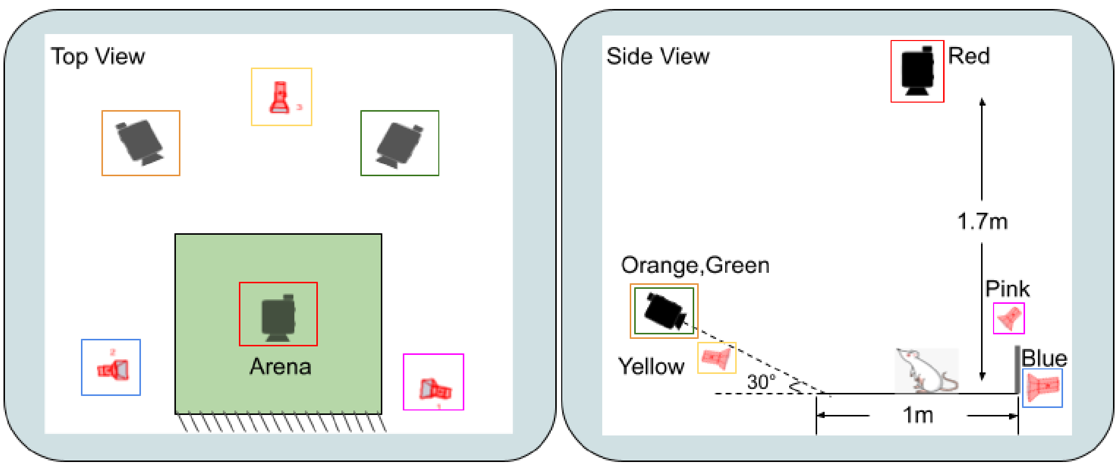


Figure 3·1: Setup of video cameras for Rodent3D dataset collection, including thermal cameras (Red, Orange, and Green) and RGB-D cameras (Blue, Yellow, and Pink). The animal arena, 1 m × 1 m, has either drop-off edges or walls on its sides.

Our recording system consists of 3 FLIR SC8000 cameras and 3 Intel Realsense D435 cameras Tab. 3.1. The SC8000 cameras can record thermal infrared videos of 1024x1024 pixel frames at 130 Hz. To ensure seamless recordings without frame drops, we used FLIR's High-Speed Data Recorders (HSDRs). Each HSDR connects to one thermal camera to act as a buffer for the recording computer. The recording capability of the Realsense D435 camera is dependent on the number of recorded streams and the resolution; therefore, the D435 cameras acted as the rate-limiting factor in our

Table 3.1: Hardware information of the recording cameras.

Model	Resolution (px)	Focal length	Pitch	Modality
SC8000	1024x1024@130Hz	25 mm	18 μm	Thermal
D435	848x480@60Hz	1.8 mm	3.3 μm	RGB, depth

configuration. Consequently, we configured the D435 cameras at 848x480 spatial resolution, 16 bits depth stream, and 24 bits RGB stream to ensure high-resolution data and selected a 60 Hz constant framerate for all cameras. Since the depth cameras do not support HSDR hardware, we used timestamps to detect frame drops. We used this information during reconstruction to filter out invalid views. In addition to the calibration accuracy, the orientation of the cameras towards the arena also affects the quality of the reconstruction. To minimize the reconstruction uncertainty, we used the EasyCamera method [31] to optimize the design of our camera configuration. The resulting overlapping views of the cameras are defined in Fig. 3·1.

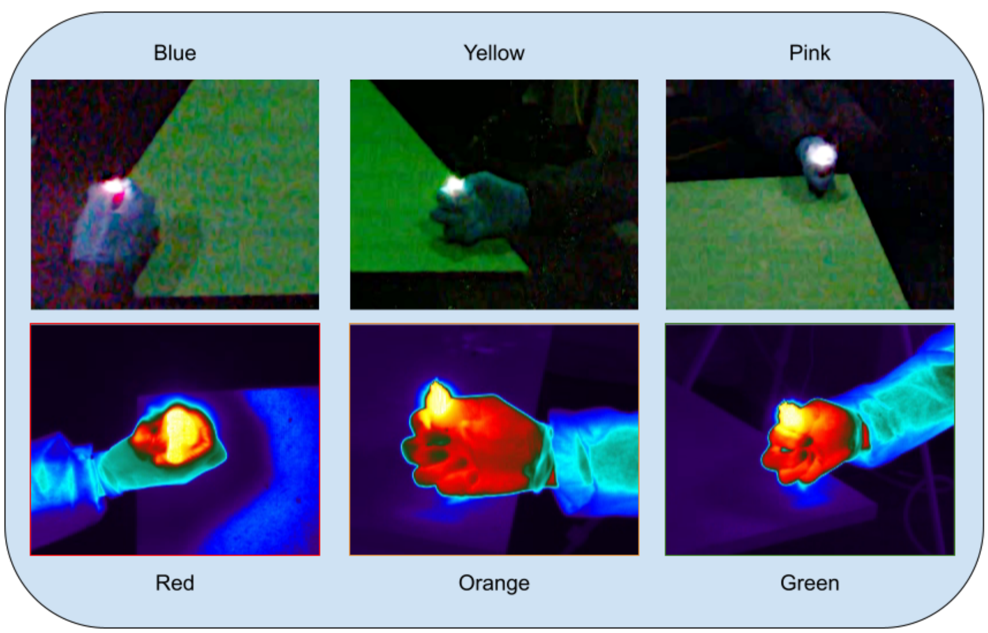
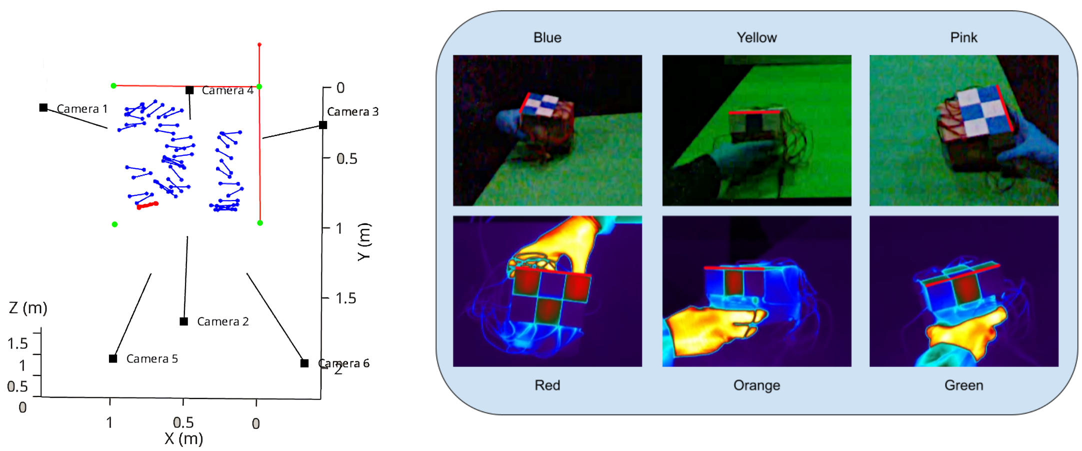


Figure 3·2: Frame synchronization event from the 6 camera views

One of the most critical steps of any multi-view computer vision task is camera synchronization. Our cameras are hardware synchronized to an external 60 Hz, 1.8 Volts, TTL (Transistor-Transistor Logic) signal generated by a fourth non-recording RealSense camera. Using such a hardware synchronization ensures that all cameras acquire frames simultaneously. Although they support a synchronization signal from specialized hardware such as a function generator through an additional circuit, using an extra camera in a master-slave configuration was simpler. However, manual video alignment is still required to overcome camera initialization delays. Therefore, we started our recording sessions with a lighter flame event, visible in all modalities, see Fig. 3·2. We used FFMPEG to align these videos based on the corresponding timestamps of such events.

3.2 Camera Calibration



(a) (b)

Figure 3·3: Camera calibration annotation: (a) Reconstructed wand points through EasyWand [31]; and (b) Visualization of synchronized wand annotation corresponding to the red wand in (a)

Although we used standard techniques to calibrate the RGB-D cameras, the traditional computer-vision calibration method for visible-light cameras that involves a flat checkerboard is not suitable in the setting of thermal cameras. We, therefore, designed a calibration cube with thermoelectric devices that uses the Peltier effect to generate hot and cold surfaces. The cube can thus be imaged by our thermal cameras, as shown in Fig 3·3-b. We annotated 45 synchronized frames.

We used the EasyWand method [31] to process the annotated calibration data with bundle adjustment and generated the direct linear transformation (DLT) coefficients. We annotated 45 synchronized frames, each containing one wand and four static corners of the arena.

Chapter 4

OptiPose: A Self-Attention Based Pose Optimizer

OptiPose is a supervised model that refines the raw 3D keypoints reconstructed from any state-of-the-art 3D pose estimation method, yielding refined pose estimates. Fig. 4·1 defines the pipeline. The coarse 3D keypoints were obtained by performing multi-view reconstruction using the DLT coefficients, obtained through the process described in the previous chapter, with the markerless 2D tracking provided by DeepLabCut [23]. We cleaned the reconstruction errors through the pre-processing techniques described in Sec. 4.3. After parsing the reconstruction data and collecting clusters of continuous data points without missing keypoints, we used our training data augmentation algorithms defined in Sec. 4.4 to generate the dataset. The training and testing of OptiPose is done in a sliding window manner, meaning that a sequence $P_{1..T}$ of T 3D poses, i.e., sets of 3D keypoints $\{x_1, x_2, \ldots, x_N\}_{t=1,..,T}$, depicting a certain motion, are processed by OptiPose one at a time. The central component of OptiPose is the "context models" (CM) that build on the idea of Self Attention encoders [32]. Furthermore, we used temporal [6] and structural [25, 5] loss functions to promote learning the inherent geometrical patterns. The architecture is discussed further in the next section.

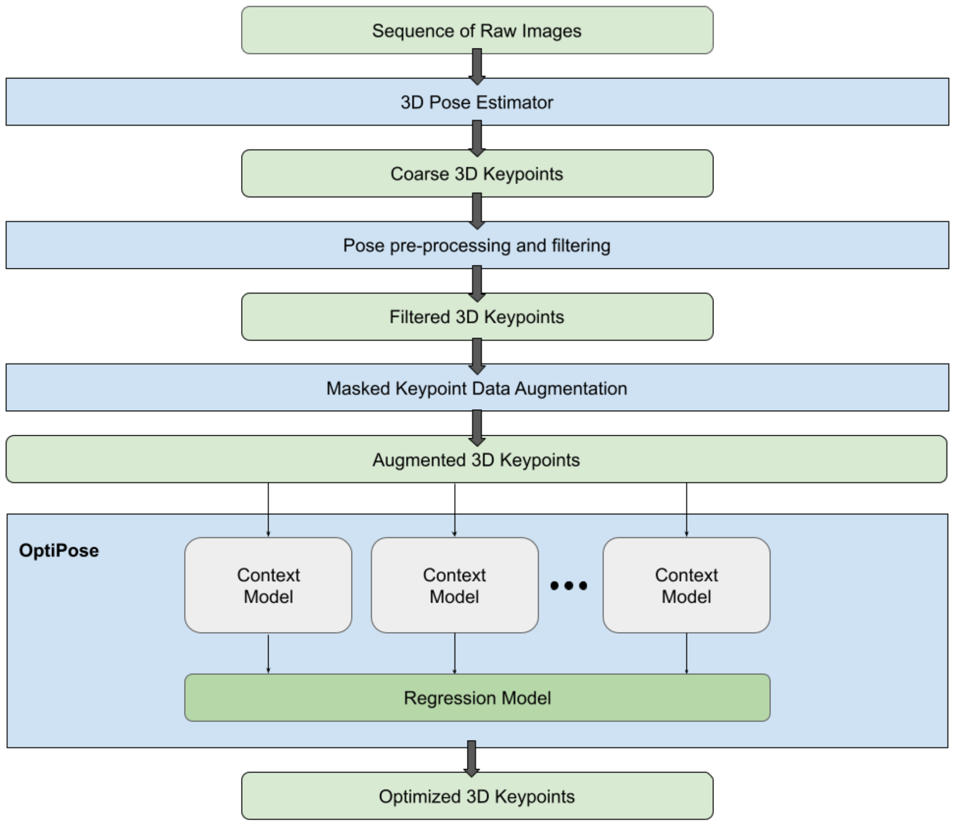


Figure 4·1: A general pipeline for training the OptiPose model.

4.1 Architecture

The core of OptiPose is a set of parallel context models (CMs), shown in Fig. 4·2. Each CM receives a sequence of 3D poses, i.e., tokens, and computes an embedding vector. We denote the token as P and a sequence of tokens as $P_{1...T} \in \mathbb{R}^{3N}$, where the length of the sequence is denoted by T and N is the number of keypoints per pose token. The output of each CM is in the dimension of $T \times E$ (e.g., we set $E = 64$ in our experiments).

These outputs are concatenated and fed into a fully connected dense linear layer (dark green box at the bottom of each CM block in Fig. 4·2) and that reduces the

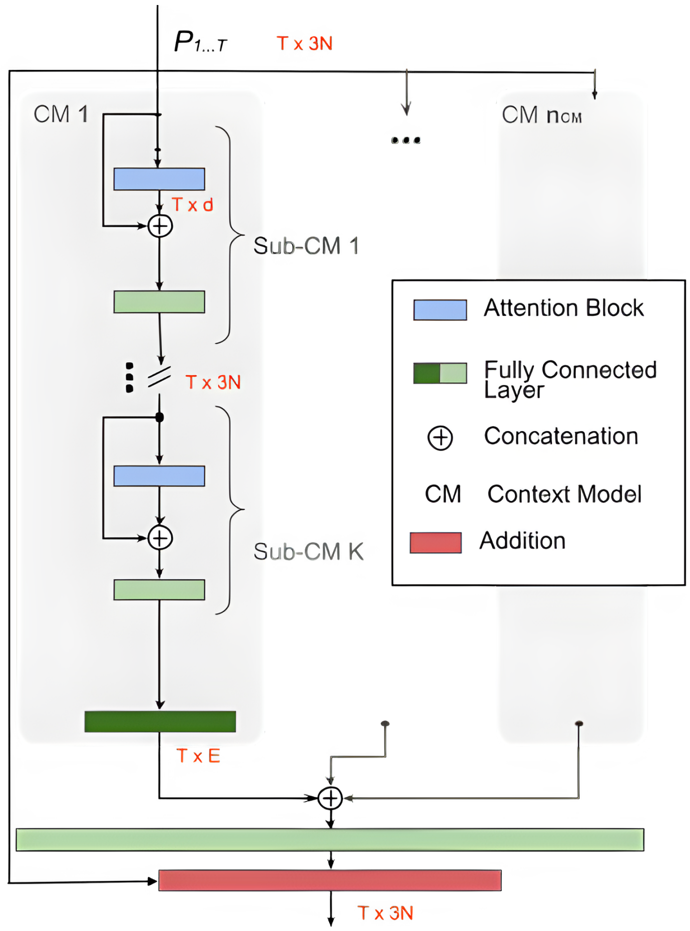


Figure 4·2: Architecture of OptiPose. It consists of parallel context models CMs (grey boxes) that share the same input $P_{1...T}$ but whose weights are initialized differently during training. Each CM contains K submodels. A submodel contains an attention block and a fully connected layer.

dimensionality back to a $T \times 3N$-dimensional output, generating the offsets required for refining the poses. Finally, these offsets are added to the masked input to obtain the optimized 3D poses. Every CM consists of a sequence of sub-CMs that take $T \times 3N$ input and generates the same dimension output. Each sub-CM has an attention block (blue) and a fully connected layer (green). The output of the attention block is concatenated with its input and fed forward to a fully connected layer (dark green), as in a standard encoder block [32]. Keeping a sub-CM's input and output dimensions the same allows us to vary the number K of sub-CMs.

The attention blocks in OptiPose is implemented as a single-headed self-attention model [32]. It provides a unique Spatio-temporal mechanism of attention. The self-attention mechanism in natural language processing is multi-headed, with each head focusing on a subset of embeddings in parallel. However, in the case of pose optimization, where the input embeddings are the pose themselves, multi-headed attention would fail to recognize patterns among the keypoints from the other heads. Therefore, to enable OptiPose to learn to pay attention to different Spatio-temporal patterns, we employ multiple CMs in parallel. We chose a single-headed attention mechanism since the keypoints all have distinct relationships.

OptiPose learns to pay attention to the spatial relationships between keypoints in a pose and temporal relationships among poses. It weighs keypoint information differently in case of occlusion. The temporal attention mechanism is essential when sets of keypoints are missing in consecutive frames, and the locations of these keypoints need to be estimated.

All fully connected layers in the CMs have a Parametric ReLU (PReLU [13]) as the nonlinear activation function. Using PReLU ensures that our model does not have an upper limit on the range of keypoint locations, making the prediction-range task-adaptive (i.e., laboratory scale for the rodent and wildlife-enclosure scale for the cheetah). The number of parallel CMs (n_{cm}) and sub-CMs (K), as well as the parameters ($d(64)$ and $E(30)$) that determine the length of attention and embedding encodings, are hyperparameters that can be adjusted according to the number of keypoints in different animal species. For our experiments, the maximum length of attention was 30 and the size of embedding encodings was 64.

4.2 Spatio-Temporal Loss Functions

Keypoints of interest in many animal pose estimation tasks are connected to one another and the distances between some keypoints are constrained by rigid bones. We capture such constraints using an undirected graph $G = (V, E)$ where the graph nodes set V represents a set of keypoints, and the edge set E represents the distance between keypoints. We incorporate this geometric constraint and define the structural loss [25, 5] as

$$\mathcal{L}_{st} = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{N} \left(\|x_i - x_j\|_2 - \|\hat{x}_i - \hat{x}_j\|_2 \right)^2, \tag{4.1}$$

where N is the number of keypoints, and $x_i, \hat{x}_i \in \mathbb{R}^3$ are the predicted and the groundtruth coordinates of the i-th keypoint, respectively. We also introduce a temporal constraint over T consecutive poses that is designed to capture the movement pattern of the animal, called temporal loss [6],

$$\mathcal{L}_{tp} = \frac{1}{T} \sum_{t=1}^{T} \sum_{i=1}^{N} \left\| (x_i^{(t)} - x_i^{(t-1)}) - (\hat{x}_i^{(t)} - \hat{x}_i^{(t-1)}) \right\|, \tag{4.2}$$

where N is the number of keypoints, $x_i^{(t)}$ is the predicted position and $\hat{x}_i^{(t)}$ the groundtruth at time t. We define our combined loss function $\mathcal{L}$ as a weighted sum of structural and temporal losses [36]:

$$\mathcal{L} = \frac{1}{T} \sum_{t=1}^{T} \left(\frac{1}{N} \sum_{i=1}^{N} \|x_i - \hat{x}_i\|_2 + \alpha \, \mathcal{L}_{st} \right) + \beta \, \mathcal{L}_{tp}, \tag{4.3}$$

where α and β are hyperparameters weighing the structural loss $\mathcal{L}_{st}$ and the temporal loss $\mathcal{L}_{tp}$.

4.3 Pre-processing and Cleaning Poses

We perform pre-processing of the poses in stages. We used DeepLabCut [23] to perform 2D keypoint detection. It provides a likelihood value $\in [0, 1]$ for each keypoint which can be thresholded to filter out noisy predictions. We used the frame drop information collected during data acquisition to set the likelihood values in corresponding frames to 0, ensuring that they will not be used in our reconstruction pipeline. We then used a likelihood threshold of 0.95 for filtering views during the reconstruction. This step is essential since we padded the dropped frames with the last recorded frame to maintain the alignment of subsequent frames with other views. However, if we record the videos at a very high frame rate, the noise added by padded duplicate frames in reconstruction would be minimum for a short period.

The outlier detection algorithm defined in Algorithm 1 takes advantage of the fact that the keypoints are grouped together in most animals. The algorithm computes distances for each keypoint to all other available keypoints. If the median of those distances is larger than the threshold, we can assume that the keypoint in question is noisy. A simple threshold value could be the average maximum distance among the keypoints. For our experiments, we used 350 mm as the threshold. The primary application of this method is to remove outliers located far away from the accurate keypoints. In such cases, the median distance for the outliers would likely be higher than the threshold, whereas the median distance for the clustered valid keypoints will be lesser. However, the downside of this method is that it assumes there always will be at least a subset of valid keypoints per pose.

4.4 Data Augmentation

For OptiPose, we cannot use the standard data augmentation techniques used in computer vision [30], given that OptiPose manipulates 3D keypoints and not images.

Algorithm 1 Outlier Detection Algorithm

Input $P, threshold$ **Output** P

for $keypoint \in P$ **do**
  $distances \leftarrow []$
  for $ref \in P$ **do**
    $add(distances, distance(keypoint, ref))$
  if $Median(distances) > threshold$ **then**
    $remove(P, keypoint)$

Algorithm 2 Masked Keypoint Data Augmentation

Input $T, N, datasets$ **Output** $datasetsAug$

Initialization: $datasetsAug \leftarrow []$
for $iteration = 1, 2, \ldots$ **do**
  $rotations \leftarrow$ uniformly sample from $\{0, \ldots, 360\}$
  for $r \in rotations$ **do**
    for $d \in datasets$ **do**
      $P \leftarrow$ sample T consecutive poses $\in d$
      $f \leftarrow$ select augmentation function with p
      $P' \leftarrow f(P)$
      $s \leftarrow$ random scaling factor $\in [0.5, 1.5]$
      $scale(P', s)$
      $rotate(P', \text{Z-axis}, r)$
      $v \leftarrow$ random translation vector
      $translate(P', v)$
      $insert(datasetsAug, P')$

Nonetheless, we can employ standard geometric transformations such as rotation and translation, and noise addition for augmentation. In addition, we mask a subset of poses (Algorithm-2) to promote learning of geometric patterns, similar to masked language modelling for natural language processing [7]. One key difference is that we mask the embeddings partially in addition to the complete poses.

This enables us to train OptiPose to handle the occlusion of keypoints on the animal, e.g., an occluded ear or foot, as we will show in the ablation study below. Such occlusions may not appear in the training set otherwise, and so the purpose of the augmentation algorithm is to maximize the variety of pose sequences in the total training set to encourage the model to learn the structure of the animal. Relying on random decisions at every step ensures that the augmented data, while still physically valid, is substantially different from the original data, and thus enables OptiPose to generalize and correctly interpret unseen data.

We designed three augmentation functions (f) to mimic real world scenarios or to enforce the learning of specific spatio-temporal features. They are selected at random in the main algorithm for $P' \leftarrow f(P)$. The probability assigned to each function is also a hyper-parameter and can be tuned to specific real world recording conditions.

- **Function 1: RandomNoise.** This function does not mask any pose or keypoints, however, random noise is added to a uniformly selected subset of poses. The motivation behind this augmentation was to encourage the model to correct errors in visible keypoints. The probability p for this function is *0.01*.

- **Function 2: Masked Pose.** This function has three variants. The first variant uniformly masks either odd indexed poses or even indexed poses. The second variant masks a cluster of randomly selected consecutive poses. Whereas the third variant uniformly selects and masks non-consecutive poses. These functions encourage the model to understand the activity defined by P poses at temporal scale. The total probability for all three variants is *0.38*.

- **Function 3: Masked Keypoints.** This function has two variants. The first variant uniformly masks at most $\frac{N}{2}$ keypoints in every pose in P. The second variant masks the keypoints selected at random for each pose. The first variant promotes spatial reasoning whereas the second variant promotes both spatial and temporal reasoning. The total probability for this function is *0.61*.

For the dataset of at most T consecutive poses with N keypoints, we uniformly sample a set $Rotations$ such that $Rotations \subset \{0, \ldots, 360\}$. For each $r \in Rotations$, we

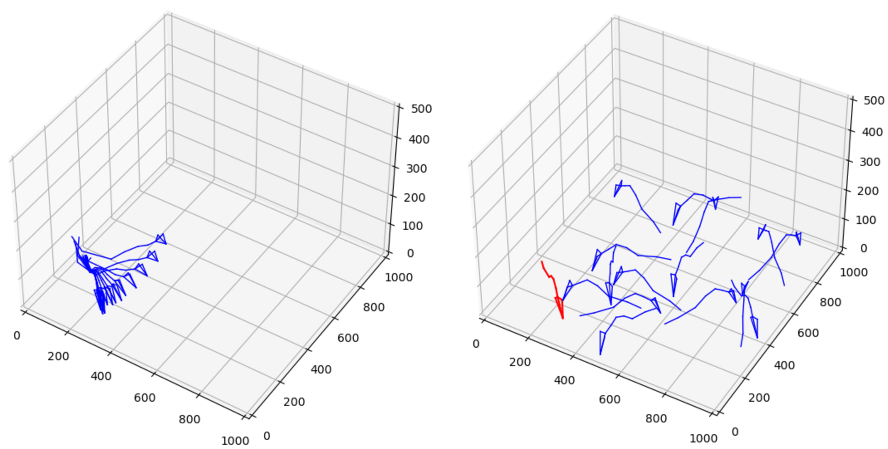


Figure 4·3: Left: Plot of the activity defined by the ground-truth. Right: Plot of first pose of the original data (red) and the augmented data(blue)

sample P consecutive poses such that $|P| \leq T$. We apply one of three augmentation functions, chosen randomly. The resulting partial poses P' are then scaled, rotated in the XY (horizontal) plane by r degrees followed by a 3D translation. Fig. 4·3(left) shows a sequence of unmasked poses that defines "sweeping head" activity. The Fig. 4·3(right) shows the plot of the first pose of the sequence after applying the rotation and translation. Due to random masking, these sets of partial poses will each act as an independent input to the model.

Chapter 5

Experiments and Results

Table 5.1: Configuration of OptiPose used for experiments (Fig.4·2). PCM and Sub-CM will vary for ablation study.

Parameters	Values
PCM (n_{cm})	10
Sub-CM (K)	3
Embedding Dimension (E)	30
Attention Dimension (d)	64

We performed our experiments using three datasets - Rodent3D, AcinoSet, and Rat7M. The architecture parameters (Sec.4.1) for all experiments except the ablation studies are as defined in Tab. 5.1. For the Rodent3D dataset, we evaluated our model on data from videos with different modalities and refresh rates. We also compared our model with a publicly available tool called Anipose [16], which shares a similar pipeline as ours. Finally, we performed ablation studies to evaluate the impact of different architecture variations and the loss functions.

5.1 Evaluation Metric and Baseline Identity Model

We used the Probability of Correct Keypoint (PCK) metric at 5% and 10% thresholds. It is defined as follows:

$$\frac{1}{n}\sum_{i=1}^{n}\begin{cases}1, & \text{if } d_i < thresh \times \max(d)\\ 0, & \text{otherwise}\end{cases}$$

Table 5.2: Average PCK accuracy of OptiPose per keypoint over 6,500 sets of $T \leq F$ consecutive poses on the Rodent3D dataset (Top: color input, $F = 30$. Bottom: thermal input $F = 60$). Baseline model as described in Sec.5.1.

Rodent3D:	Snout	RightEar	LeftEar	HeadBase	Mid	TailBase	TailMid	TailTip	Avg
Baseline $H(P)$	64.50	64.52	64.73	64.86	65.00	65.68	64.97	64.90	64.90
PCK@0.05	74.08	83.61	85.04	85.82	78.51	80.59	84.16	80.66	81.56
PCK@0.10	**85.37**	**92.54**	**92.90**	**93.38**	**90.25**	**90.62**	**91.41**	**87.34**	**90.48**
Rodent3D:	Snout	RightEar	LeftEar	HeadBase	Mid	TailBase	TailMid	TailTip	Avg
Baseline $H(P)$	65.32	65.77	65.21	65.76	66.14	65.24	65.85	65.37	65.56
PCK@0.05	78.66	82.11	82.19	78.11	83.76	82.24	80.33	74.34	80.91
PCK@0.10	**89.42**	**92.69**	**92.95**	**94.22**	**93.14**	**92.05**	**90.4**	**86.35**	**91.89**

where d_i is the distance between prediction and output, $\max(d)$ is the maximum distance among any two points in the ground truth, and n is the number of keypoints.

For quantitative comparison of how much OptiPose improves the noisy data, we assume a hypothetical identity model H as our baseline such that for a sequence of noisy poses P, $H(P) = P$. Using the above evaluation metric on baseline output will give us the percentage of input keypoints that match the test dataset's ground truth. Therefore, a higher evaluation score for the OptiPose model would indicate that it successfully optimizes the poses by reducing noise and filling in the masked data.

5.2 OptiPose on Rodent3D

On average, OptiPose improves the noise-added inputs by 15 percentage points within 5% and 24 percentage points within 10% of their ground truth range. Estimates of raw keypoint locations can be highly inaccurate due to occlusion, missed detection by the 2D keypoint extractor, or errors introduced in the 3D reconstruction, as Fig. 1·1 illustrates. One may argue that such inaccuracies can be smoothed out or interpolated by a temporal filter, thus making a keypoint refiner like OptiPose less relevant. However, simple smoothing or interpolation will lead to information loss for keypoints that are missing for a lengthy period. Outlier coordinates of a keypoint cannot easily be substituted.

As shown in Fig. 5·1, we plot the trajectory of the snout of a rodent in x, y, and z as functions of time and highlight that OptiPose can estimate snout position during long periods when the accuracy of raw keypoint locations is poor.

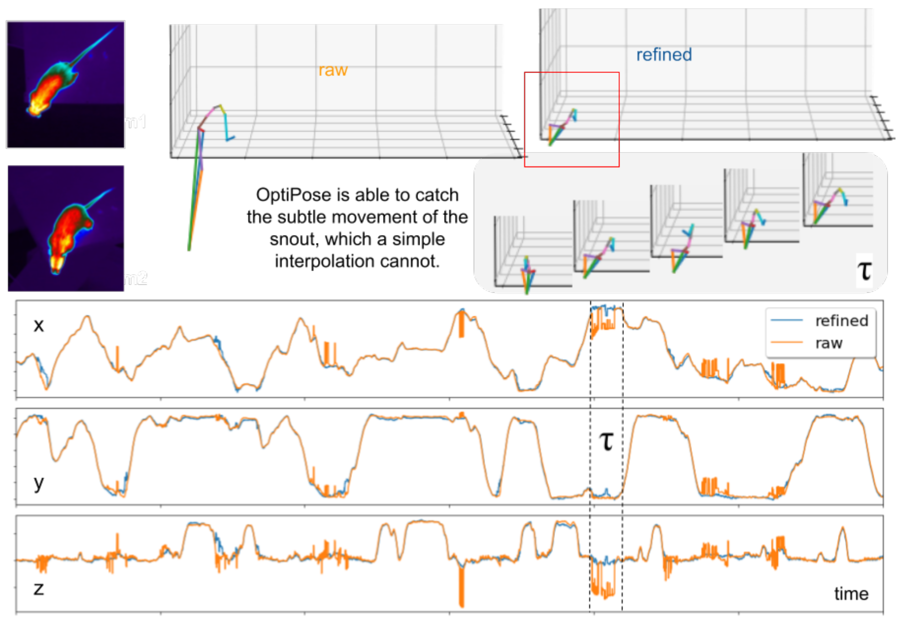


Figure 5·1: OptiPose can catch the subtle movement, which a simple interpolation is unable to. Top: The rodent's 3D pose (red box) shown in the infrared images is a refined version of the inaccurate raw 3D pose shown. Bottom: The plots show raw (orange) and refined (blue, by OptiPose) keypoint coordinates for a sequence of 12,000 frames (100 s). Interpolation will not provide accurate snout positions, particularly during challenging periods like the 360-frame long period marked τ.

We applied the OptiPose model trained on the color dataset on a thermal dataset that was recorded in a different session (Tab. 5.3). This experiment shows that OptiPose generalizes well on unseen data type and unseen animal behavior.

Table 5.3: Average PCK accuracy of OptiPose, trained on 30 Hz RGB data, over 6,000 sets of $T \leq 30$ consecutive poses, recorded at 120 Hz from thermal cameras. Baseline model as described in Sec.5.1

Rodent3D:	Snout	RightEar	LeftEar	HeadBase	Mid	TailBase	TailMid	TailTip	Avg
Baseline $H(P)$	64.86	65.51	64.87	65.41	64.55	64.62	65.30	65.12	65.03
PCK@0.05	74.48	83.45	84.86	85.50	77.86	79.93	84.80	80.44	81.41
PCK@0.10	**85.44**	**92.52**	**92.58**	**93.18**	**90.02**	**90.33**	**91.86**	**87.08**	**90.38**

5.3 OptiPose on AcinoSet

We conducted experiments to compare the performance of our OptiPose model on the AcinoSet dataset [15] with the performance of the state-of-the-art Full Trajectory Estimation (FTE) technique provided with the AcinoSet [15]. As recommended [15], we treated the 3D cheetah poses reconstructed by FTE from six camera views (FTE6) as the ground truth 3D poses.

We applied our data augmentation method to the published 6-camera FTE reconstructed 3D poses to create a training dataset for OptiPose. For testing, we compared our model performance with the FTE performance in settings where FTE uses fewer than six cameras to provide 3D pose reconstructions (ground truth is still obtained from all six views). We denote these baseline models as FTE2 and FTE4, corresponding to the settings with two or four cameras only. We treat the FTE2 and FTE4 outputs as raw 3D keypoints and aim to improve their accuracy with OptiPose. We used 14 videos (2,204 frames) for training and two videos (331 frames) for testing. The resulting augmented training dataset had 22,400 sequences of poses. The results indicate that OptiPose outperforms FTE in both settings. It improves the FTE2 by $\sim$2 percent points and FTE4 by $\sim$3 percent points (Tab. 5.4). In addition, inference with OptiPose is faster than with iterative constraint optimization [15] (seconds versus minutes).

Table 5.4: PCK@0.1 accuracy of baselines and OptiPose on the cheetah “run” data of the AcinoSet [15]. The ground truth is FTE with 6 cameras.

Method	2 cameras	4 cameras
FTE2 [15]	73.415	NA
FTE4 [15]	NA	90.09
OptiPose	**75.44**	**92.92**

Table 5.5: Average PCK accuracy of OptiPose per keypoint over 2,909 sets of $T \leq 30$ consecutive poses on the Rat7M [21] motion capture dataset.

Rat7M:	HeadF	HeadB	HeadL	SpineF	SpineM	SpineL	Offset1	Offset2	
Baseline $H(P)$	70.97	70.61	71.04	70.61	70.94	70.54	71.33	70.80	
PCK@0.05	75.75	75.51	75.66	71.30	75.66	79.16	82.76	61.07	
PCK@0.10	85.23	87.44	84.76	86.62	88.36	89.34	89.12	79.07	
	HipL	HipR	ShoulderL	ShoulderR	KneeR	KneeL	ShinL	ShinR	Avg
Baseline $H(P)$	71.05	71.44	71.11	71.43	71.13	71.08	71.15	71.93	71.07
PCK@0.05	69.82	78.59	65.94	72.32	67.25	71.80	75.90	84.47	73.94
PCK@0.10	82.19	88.80	76.11	86.00	81.72	87.48	87.29	89.44	85.56

5.4 OptiPose on Rat7M

We conducted experiments to evaluate OptiPose on the motion capture data provided in the Rat7M dataset [21]. We trained OptiPose on the subject-1-day-1 and evaluated the model performance on the subject-2-day-1 dataset. We generated our training dataset by applying our data augmentation and masking method to the 3D poses of subject-1-day-1, defined by 16 keypoints. The training set consists of 126K frames and the test set of 184K frames. We report the PCK@5 and PCK@10 values of our Rat 7M experiment in Tab. 5.5. OptiPose improves the noisy input by ∼3 percent points (PCK@5) and ∼14 percent points (PCK@10).

Table 5.6: Reprojection error in pixels onto the three RGB-D camera views.

View	Model	Snout	RightEar	LeftEar	HeadBase	Mid	TailBase	TailMid	TailTip
Blue	OptiPose	5.93	5.78	5.19	4.99	6.99	4.17	5.26	5.06
	Anipose [16]	9.26	9.89	9.21	8.63	7.76	7.52	7.62	7.98
Pink	OptiPose	4.89	4.14	4.24	3.95	4.54	3.07	4.37	3.41
	Anipose [16]	13.23	12.96	13.20	12.34	12.73	11.44	11.86	10.83
Yellow	OptiPose	57.04	47.27	46.88	45.53	33.49	28.01	28.44	27.41
	Anipose [16]	59.05	49.33	49.37	47.97	36.45	32.43	32.43	31.33

5.5 Comparision: OptiPose and Anipose

To compare the performance of Anipose versus OptiPose, we reprojected the 3D reconstructed keypoints back to their original 2D planes and compared these reprojected keypoints in 2D with the manually annotated ones. We report the 2D Euclidean distance in pixels in Tab. 5.6. OptiPose outperforms Anipose in all views in all keypoints. It is noticeable that data from the front-view camera (Yellow), see Fig. 3·1, suffers the most error in both OptiPose and Anipose. It is consistent with our observation that the occasional frame drops happened mainly in the recordings from the Yellow camera.

5.6 Ablation Study

5.6.1 Architecture Variants

To study the impact of the number of sub context models (sub-CM) and parallel context models (PCM), we trained a variety of combinations of the architecture, containing 1 and 3 sub-CMs and 4, 7 and 10 PCMs for both the Rodent3D (8 keypoints) dataset and the AcinoSet (20 keypoints) dataset respectively. It is evident from Tab. 5.7 (Left) that a higher number of parallel context models and sub-CMs, here 10 and 3 respectively, is beneficial for increasing the overall accuracy of keypoint

Table 5.7: Ablation study for OptiPose with maxium 30 pose input. Left:PCK@0.1 averaged across all keypoints for different OptiPose variations. Right: Number of trainable parameters for different OptiPose variants.

Model	Datasets	1-sub-CM	3-sub-CM
Baseline	Rodent3D	64.80	64.80
	AcinoSet	67.11	67.11
PCM-4	Rodent3D	77.49	88.54
	AcinoSet	69.83	74.24
PCM-7	Rodent3D	83.88	88.59
	AcinoSet	74.06	81.52
PCM-10	Rodent3D	80.26	90.46
	AcinoSet	77.40	84.26

Model	Datasets	1-sub-CM	3-sub-CM
PCM-4	Rodent3D	58,264	149,656
	AcinoSet	160,828	420,412
PCM-7	Rodent3D	92,440	233,368
	AcinoSet	258,076	665,692
PCM-10	Rodent3D	126,616	317,080
	AcinoSet	355,324	910,972

Table 5.8: PCK@0.05 of OptiPose-10PCM-3SubCM, trained with different loss functions, on Rodent3D dataset. The baseline accuracy (Sec. 5.1) of the test dataset was 64.14

Loss Function	PCK@0.05
MSE	79.32
$\mathcal{L}_{st}$	**87.39**
$\mathcal{L}_{tp}$	86.16
$\mathcal{L}_{st}$ $+\mathcal{L}_{tp}$	86.21

estimation. The more keypoints that are used to model animal shape and motion, the higher the number of parallel context models should be.

5.6.2 Loss Functions

In order to perform ablation study for the loss functions, we trained an OptiPose model with 10 PCMs and 3 Sub-CMs on Rodent3D dataset. For each loss functions variants, we trained our model for 1000 epochs with α and β set to 0.0001 and 0.00005 (see Sec. 4.2 for details). We used the Adam Optimizer [17] with a learning rate 0.0005 and an exponential learning rate decay. Tab. 5.8 provides accuracy on the test dataset for different loss functions. The results show that the performance of proposed

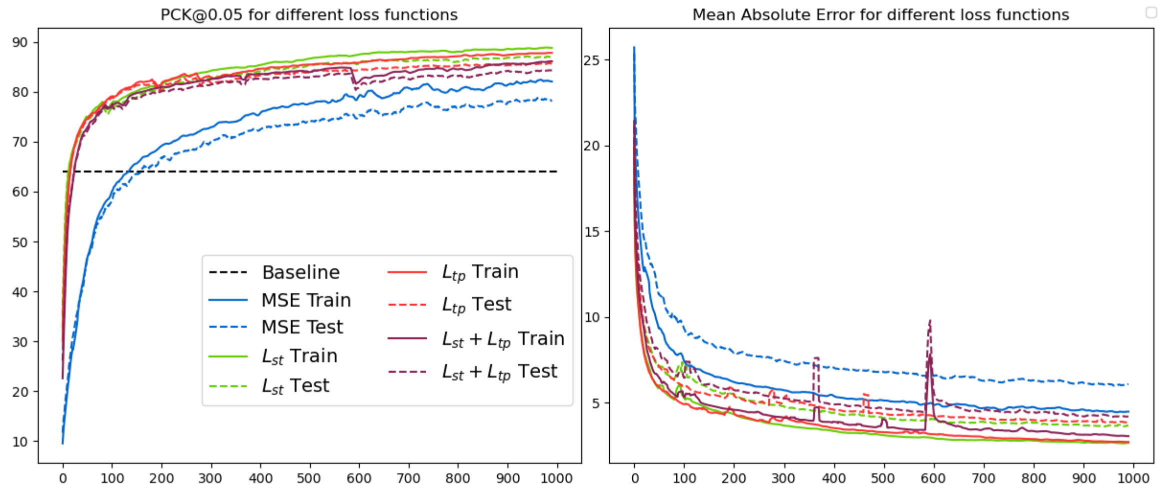


Figure 5·2: Left: Plot of PCK@0.05 for different loss function against training epoch. Right: Mean Absolute Error for different loss functions against training epochs

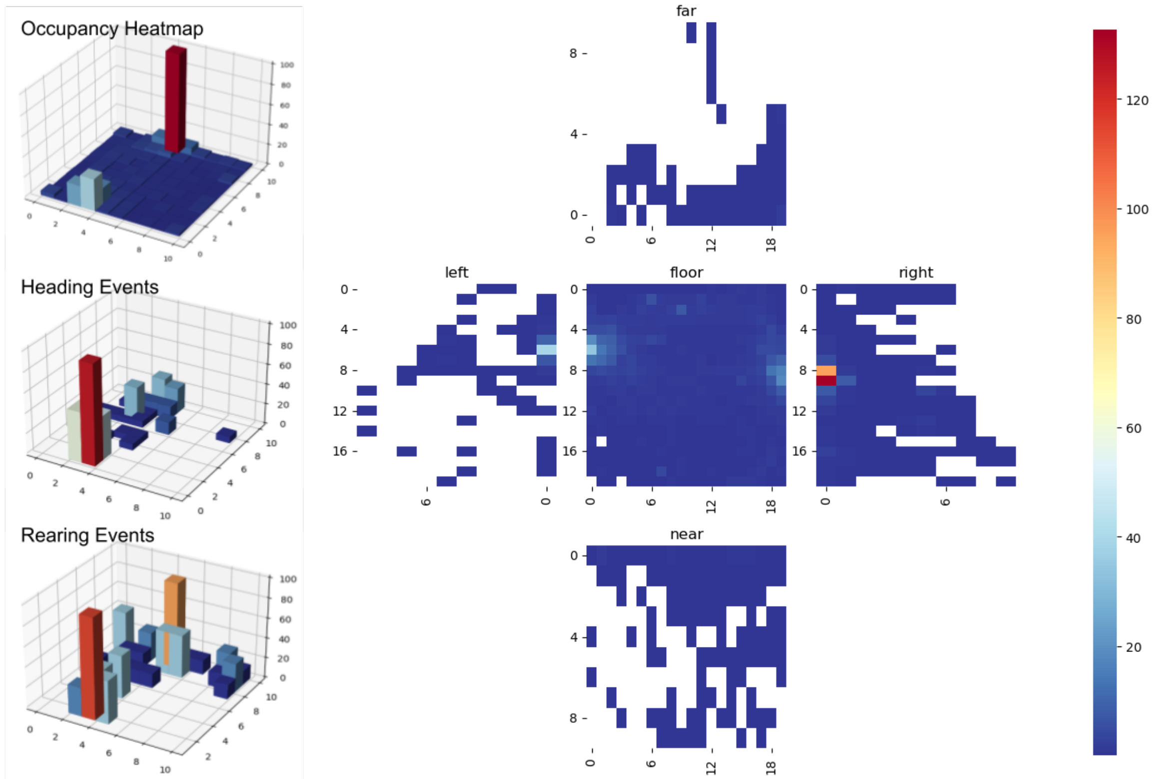
cost function was actually slightly worse than the performance of the structural loss function. However, the difference is not significant and could be attributed to the complexity of the loss function. It might need more epochs or a different scalar coefficient to converge faster.

Chapter 6

Discussion and Analysis

Accurate 3D poses allow us to perform complex analyses of animal behavior. Two components of the pose, the head direction and the body direction, are widely used for behavior analysis in rodents [1]. We defined the "head direction" as a vector from HeadBase to Snout keypoints and the "body direction" as a vector from Midpoint to HeadBase keypoints. The location of the rodent at any instance was computed by averaging the positions of the keypoints-Snout, HeadBase, and Midpoint.

For our analysis, we used a session where the arena was surrounded by three walls and a platform drop-off on the fourth side. Furthermore, this session was recorded with only top view and the front view (see Fig.3·1). Two automatic food dispensers were placed in the middle of on the left and right walls, respectively. The spatial heatmaps shown in Fig. 6·1 are examples of types of behavioral analysis that can be performed with Rodent3D and OptiPose. The occupancy plot represents the locations where rodent spent most of its time. The hot-spots near the Left and Right walls are the rough positions of the automatic food dispensers. Consequently, they are the most explored areas. The bottom plot indicates the locations and frequency of the instances of rearing behavior which is one of the exploration indicators. High rearing occupancy values were found near the region close to a food dispenser and the corners of the walls. The Heading plot indicates all locations in which the head direction led the body direction of the animal. The positions shown with higher values are near a known reward location. It is evident that the animal likely turned its head to face a



(a) (b)

Figure 6·1: Behavioral analysis (a) Spatial heatmaps measuring occurrence of rodent behaviors across a 1 m by 1 m laboratory arena. Top: Discrete locations of the rodent in bin units of 10 cm by 10 cm. Middle: Locations of "heading events" where the body direction consistently followed the head direction, notably between and at food dispensers. Bottom: Locations where rodent performed a 'rearing' action, notably near the food dispensers and arena corners. (b) Heatmaps of where the rodent is approximately looking per second. The five plots show the five surfaces of the arena.

reward location, then moved in the corresponding direction to navigate to the reward. Fig.6·1(b) plots the frequency of how much time the rodent spent, looking at 5 cm x 5 cm blocks on one of the surfaces. We assumed that the viewing direction was the same as the head direction. By simply tracing the head-direction vector to one of the surfaces, we can estimate a general position of where the rat is looking.

For accurate analysis, it is essential that the tracking is robust. Our results show that OptiPose performs and generalizes well on different input modalities. Notably, the fact that the OptiPose model, trained on low-frequency data and a different modality, can still predict 3D locations of keypoints in 120 Hz thermal recordings with competing accuracy (Tab. 5.3 compared to Tab. 5.2 Bottom) suggests that OptiPose is capable of understanding relation among keypoints at spatio-temporal scale. In addition to the color dataset computer scientists use, the thermal recordings might be preferable to neuroscientists studying the rodents for avoiding unnatural behavior since light is often an aversive stimulus for these nocturnal creatures.

In our experiments we show a trade-off for having a different number of PCMs and Sub-CMs in Tab. 5.7. It is evident from the experiment that a skeleton with a higher number of joints demands more PCMs and Sub-CMs. As it is in the general scenario, the accuracy seems directly proportional to the number of trainable parameters. However, tuning the architecture is essential for finding where the trade-off plateaus. The impact of the spatio-temporal loss function is evident from our ablation study. Fig. 5·2 shows the plot of two evaluation metrics during the training. It is clear that even using one of the loss functions, helps model to converge faster. It is also evident from Tab. 5.8 that using the loss functions improves accuracy by approximately 8 percentage points. Although the structural loss function outperforms temporal and the combined loss function, we believe that tuning α and β (see Sec. 4.2) would provide much better results.

Annotations on our dataset are limited to keypoints along the head, spine, and tail of a rodent. These keypoints were chosen to address questions from experimental neuroscience about the position, orientation, and velocity of the rodent's head and body. Furthermore, it could be viewed as a limitation of our approach that we did not include CAD models in the training process, as proposed by recent work by Chen Li et al. [19]. However, the design and verification of dynamic CAD models of animals require biomechanical expertise to ensure realistic modeling, particularly of limb movements. However, our problem statement was to avoid such expertise. Nonetheless, it should be noted that if an accurate dynamic CAD model of the animal species of interest exists, it could be used for the training of OptiPose and would likely make its results even more accurate.

OptiPose is not an end-to-end pose estimator. Therefore, the model's performance depends on the quality of the 3D raw keypoints. Future work may circumvent this issue by integrating an existing pose estimation architecture with a pre-trained OptiPose model. Such an end-to-end system may benefit from OptiPose's insights on the structure and movement patterns of the animal or human.

Chapter 7

Future Work

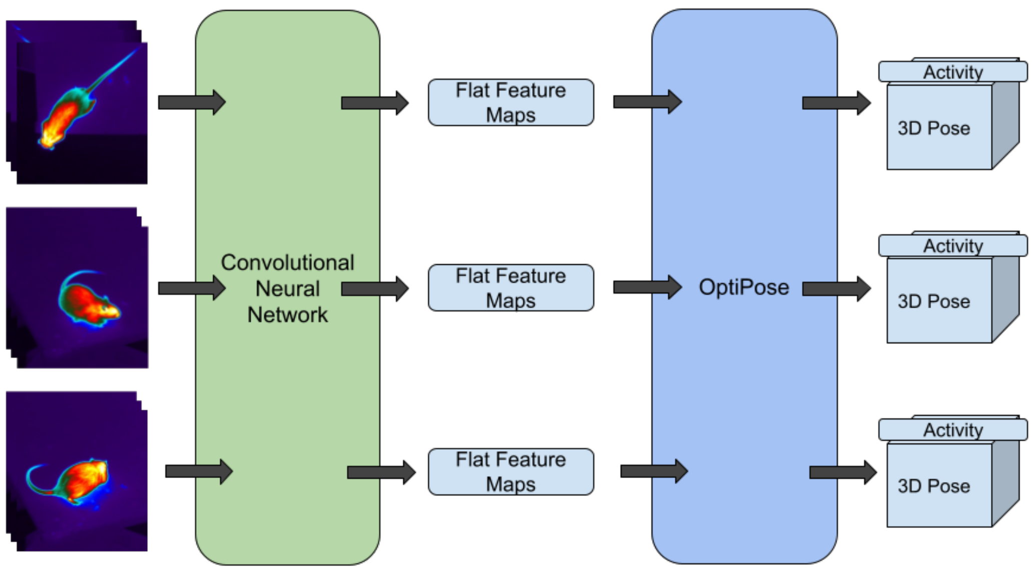


Figure 7·1: Future Work: Modification of Optipose to accept raw images as inputs and generating voxelized 3D pose and activity classification label

A notable limitation of OptiPose is that it does not use raw 2D images as its input. It relies upon the accuracy of the keypoint detection algorithm and the quality of pre-processing defined in Sec. 4.3. In future work we can address this issue by using a convolutional neural network to obtain feature vectors. However, instead of using a pose regressor or the standard hour-glass structure that generates a probability distribution over input pixels, we can pass them to the OptiPose model that outputs $n \times n \times n$ voxels Fig. 7·1. The model can also be trained to generate additional output that predicts the activity based on the spatio-temporal context. However, to make our current data augmentation method work on an end-to-end system, we have to use generative models to create synthetic views from augmented poses.

The OptiPose model does not utilize the depth information provided by the Rodent3D dataset; in the future, we can use the multi-view aligned depth maps to generate accurate point cloud reconstruction of the arena to aid the tracking and 3D ground-truth acquisition. Furthermore, for behavior analysis, a feature vector can be extracted from an earlier layer of a pre-trained OptiPose model and fed to a clustering algorithm to perform unsupervised behavioral analysis on a sequence of frames.

Bibliography

[1] Andrew S. Alexander, Lucas C. Carstensen, James R. Hinman, Florian Raudies, G. William Chapman, and Michael E. Hasselmo. "Egocentric boundary vector tuning of the retrosplenial cortex". In: *Science advances* 6.8 (2020), eaaz2322.

[2] M. Breslav, T. L. Hedrick, S. Sclaroff, and M. Betke. "Discovering Useful Parts for Pose Estimation in Sparesly Annotated Datasets". In: *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Placid, NY*. Mar. 2016.

[3] Tom B. Brown et al. *Language Models are Few-Shot Learners*. 2020. DOI: `10.48550/ARXIV.2005.14165`. URL: `https://arxiv.org/abs/2005.14165`.

[4] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. *Universal Sentence Encoder*. 2018. DOI: `10.48550/ARXIV.1803.11175`. URL: `https://arxiv.org/abs/1803.11175`.

[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*. 2020. DOI: `10.48550/ARXIV.2002.10322`. URL: `https://arxiv.org/abs/2002.10322`.

[6] Yu Cheng, Bo Yan, Bo Wang, and Robby T. Tan. "3D Human Pose Estimation using Spatio-Temporal Networks with Explicit Occlusion Training". In: *The*

Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20). 2020, pp. 10631–10638.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* 2018. DOI: 10.48550/ARXIV.1810.04805. URL: https://arxiv.org/abs/1810.04805.

[8] Timothy W. Dunn, Jesse D. Marshall, Kyle S Severson, Diego E. Aldarondo, David G.C. Hildebrand, Selmaan N. Chettih, William L. Wang, Amanda J. Gellis, David E. Carlson, Dmitriy Aronov, et al. "Geometric deep learning enables 3D kinematic profiling across species and environments". In: *Nature methods* 18.5 (2021), pp. 564–573.

[9] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. "Pictorial Structures for Object Recognition". In: *International Journal of Computer Vision* 61.1 (Jan. 2005), pp. 55–79. DOI: 10.1023/b:visi.0000042934.15159.49. URL: https://doi.org/10.1023/b:visi.0000042934.15159.49.

[10] Adam Gosztolai, Semih Günel, Victor Lobato Ríos, Marco Pietro Abrate, Daniel Morales, Helge Rhodin, Pascal Fua, and Pavan Ramdya. "LiftPose3D, a deep learning-based approach for transforming 2D to 3D pose in laboratory animals". In: *bioRxiv* (2021). eprint: https://www.biorxiv.org/content/early/2021/04/12/2020.09.18.292680.full.pdf. URL: https://www.biorxiv.org/content/early/2021/04/12/2020.09.18.292680.

[11] Jacob M. Graving, Daniel Chae, Hemal Naik, Liang Li, Benjamin Koger, Blair R. Costelloe, and Iain D. Couzin. "DeepPoseKit, a software toolkit for fast and robust animal pose estimation using deep learning". In: *eLife* 8 (2019), pp. 1–42. URL: https://doi.org/10.7554/eLife.47994.

[12] Semih Günel, Helge Rhodin, Daniel Morales, João Campagnolo, Pavan Ramdya, and Pascal Fua. "DeepFly3D: A deep learning-based approach for 3D limb and appendage tracking in tethered, adult Drosophila". In: *bioRxiv* (2019), p. 640375.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification". In: *CoRR* abs/1502.01852 (2015). arXiv: 1502.01852. URL: http://arxiv.org/abs/1502.01852.

[14] Karim Iskakov, Egor Burkov, Victor Lempitsky, and Yury Malkov. "Learnable triangulation of human pose". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 7718–7727.

[15] Daniel Joska, Liam Clark, Naoya Muramatsu, Ricardo Jericevich, Fred Nicolls, Alexander Mathis, Mackenzie W. Mathis, and Amir Patel. *AcinoSet: A 3D Pose Estimation Dataset and Baseline Models for Cheetahs in the Wild*. 2021. arXiv: 2103.13282 [cs.CV].

[16] Pierre Karashchuk, Katie L. Rupp, Evyn S. Dickinson, Eiman Azim, Bingni W. Brunton, and John C. Tuthill. "Anipose: A toolkit for robust markerless 3D pose estimation". In: *Cell Reports* 36.13 (Sept. 2021).

[17] Diederik Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *International Conference on Learning Representations (ICLR)*. 2015.

[18] Jessy Lauer, Mu Zhou, Shaokai Ye, William Menegas, Tanmay Nath, Mohammed Mostafizur Rahman, Valentina Di Santo, Daniel Soberanes, Guoping Feng, Venkatesh N. Murthy, George Lauder, Catherine Dulac, Mackenzie W. Mathis, and Alexander Mathis. "Multi-animal pose estimation and tracking with DeepLabCut". In: *bioRxiv* (2021). eprint: https://www.biorxiv.org/

content/early/2021/04/30/2021.04.30.442096.full.pdf. URL: https://www.biorxiv.org/content/early/2021/04/30/2021.04.30.442096.

[19] Chen Li and Gim Hee Lee. "From Synthetic to Real: Unsupervised Domain Adaptation for Animal Pose Estimation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2021, pp. 1482–1491.

[20] XiaoLe Liu, Si-yang Yu, Nico Flierman, Sebastian Loyola, Maarten Kamermans, Tycho M. Hoogland, and Chris I. De Zeeuw. "OptiFlex: Video-based animal pose estimation using deep learning enhanced by optical flow". In: *BioRxiv* (2020).

[21] Jesse D Marshall, Diego E Aldarondo, Timothy W Dunn, William L Wang, Gordon J Berman, and Bence P Ölveczky. "Continuous Whole-Body 3D Kinematic Recordings across the Rodent Behavioral Repertoire". In: *Neuron* 109.3 (Feb. 2021), 420–437.e8. ISSN: 0896-6273. DOI: 10.1016/j.neuron.2020.11.016. URL: https://europepmc.org/articles/PMC7864892.

[22] Jesse D. Marshall, Ugne Klibaite, Amanda Gellis, Diego E. Aldarondo, Bence P. Ölveczky, and Timothy W. Dunn. "The PAIR-R24M Dataset for Multi-animal 3D Pose Estimation". In: *bioRxiv* (2021). DOI: 10.1101/2021.11.23.469743. eprint: https://www.biorxiv.org/content/early/2021/11/25/2021.11.23.469743.full.pdf. URL: https://www.biorxiv.org/content/early/2021/11/25/2021.11.23.469743.

[23] Alexander Mathis, Pranav Mamidanna, Kevin M. Cury, Taiga Abe, Venkatesh N. Murthy, Mackenzie W. Mathis, and Matthias Bethge. "DeepLabCut: markerless pose estimation of user-defined body parts with deep learning". In: *Nature Neuroscience* 21 (2018). pdf, pp. 1281–1289.

[24] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. "Monocular 3D Human Pose Estimation In The Wild Using Improved CNN Supervision". In: *Fifth IEEE International Conference on 3D Vision (3DV)*. 2017, pp. 506–516. DOI: 10.1109/3dv.2017.00064. URL: http://gvv.mpi-inf.mpg.de/3dhp_dataset.

[25] Francesc Moreno-Noguer. "3D Human Pose Estimation From a Single Image via Distance Matrix Regression". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 10 pp. July 2017.

[26] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. "Harvesting multiple views for marker-less 3d human pose annotations". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 6988–6997.

[27] Talmo D Pereira, Diego E Aldarondo, Lindsay Willmore, Mikhail Kislin, Samuel S-H Wang, Mala Murthy, and Joshua W Shaevitz. "Fast animal pose estimation using deep neural networks". In: *Nature methods* 16.1 (2019), pp. 117–125.

[28] Talmo D. Pereira et al. "SLEAP: A deep learning system for multi-animal pose tracking". In: *Nature Methods* (Apr. 2022). DOI: 10.1038/s41592-022-01426-1. URL: https://doi.org/10.1038/s41592-022-01426-1.

[29] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. "Language Models are Unsupervised Multitask Learners". In: (2018). URL: https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf.

[30] C. Shorten and T. M. Khoshgoftaar. "A survey on Image Data Augmentation for Deep Learning". In: *J Big Data* 6.60 (2019). https://doi.org/10.1186/s40537-019-0197-0.

[31] D. H. Theriault, N. W. Fuller, B. E. Jackson, E. Bluhm, D. Evangelista, Z. Wu, M. Betke, and T. L. Hedrick. "A protocol and calibration method for accurate multi-camera field videography". In: *The Journal of Experimental Biology* 217 (Feb. 2014). Open access online, pdf., pp. 1843–1848.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need". In: *Advances in Neural Information Processing Systems*. 2017, pp. 5998–6008.

[33] Z. Wu, T. H. Kunz, and M. Betke. "Efficient Track Linking Methods for Track Graphs Using Network-flow and Set-cover Techniques". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pdf. Colorado Springs, June 2011, pp. 1185–1192.

[34] Zheng Wu and Margrit Betke. "Global Optimization for Coupled Detection and Data Association in Multiple Object Tracking". In: *Computer Vision and Image Understanding* 143 (2016), pp. 25–37.

[35] Libby Zhang, Tim Dunn, Jesse Marshall, Bence Olveczky, and Scott Linderman. "Animal pose estimation from video data with a hierarchical von Mises-Fisher-Gaussian model". In: *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*. Ed. by Arindam Banerjee and Kenji Fukumizu. Vol. 130. Proceedings of Machine Learning Research. PMLR, Apr. 2021, pp. 2800–2808. URL: https://proceedings.mlr.press/v130/zhang21h.html.

[36] Ce Zheng, Sijie Zhu, Matias Mendieta, Taojiannan Yang, Chen Chen, and Zhengming Ding. "3D Human Pose Estimation with Spatial and Temporal

Transformers". In: *IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 11656–11665.

CURRICULUM VITAE

[redacted]