Semantically-Based Animal Pose Estimation in the Wild

Margarita N. Favorskaya [1], Dmitriy N. Natalenko [1]

[1] Reshetnev Siberian State University of Science and Technology, Institute of Informatics and Telecommunications, 31, Krasnoyarsky Rabochy ave., Krasnoyarsk, 660037 Russian Federation – favorskaya@sibsau.ru, dmitriy.natalenko@mail.ru

Commission II, WG II/8

Keywords: Semantic Segmentation, Animal Pose Estimation, Deep Learning, Wild Animals.

Abstract

Accurate animal pose estimation in the wild is potentially useful for many downstream applications such as wildlife conservation. Currently, the main approach to assessing animal poses is based on identifying keypoints of the body and constructing the skeleton. However, a direct application of frameworks to human pose estimation is not successful due to the features of the skeletal structure of humans and mammals. In this study, we propose a two-stage method: coarse-tuning with animal detection using a bounding box, as is done in most similar methods, and fine-tuning with semantic segmentation of animal. The YOLOv8 Pose Estimation and Pose Keypoint Classification model was chosen as the base model for keypoint extraction. Extensive training experiments were conducted using the AwA2 dataset (with a small number of samples from own dataset), the AP-10K dataset, and the Tiger-Pose dataset. The trained model was tested on own dataset collected from camera traps in the Ergaki National Park, Russia. Experimental results show that the proposed algorithm using additional semantic segmentation increases the accuracy of animal pose estimation by 3.6-4.8% on samples of the Ergaki dataset.

1. Introduction

Animal pose estimation is a key step for understanding animal behaviour. It can be considered as a branch of the well-studied human pose estimation. The vast variation between each species, non-rigid deformations, small datasets, and limited models make it difficult to reliably and accurately estimate animal pose. At the same time, this task plays an important role in learning of animal behaviour, understanding of wildlife migration, and even protecting endangered species. Animal pose estimation has many challenges, one of which is small datasets, especially for the wild animals. In addition, huge differences in physical characteristics between animal species cause large discrepancies, and networks trained on one species cannot generalize to another species with good accuracy, although some studies with generalization ability on the unseen animals have been conducted.

Over the past few years, the situation has improved and various models have been developed to estimate 2D and 3D animal poses, and new publicly available animal pose datasets have become available (Yu et al., 2021). There are two approaches to training models: transfer between human and animal poses, and using pose models that are specific to animals due to differences in the “bones” of quadruped mammals (we are not talking about more specialized species here). Taxonomy for animal pose estimation models includes (Jiang et al., 2022):

1. 2D animal pose estimation as single animal pose estimation for animal recognition, multiple pose estimation to analyzing and understanding the social interaction between animals, and video-based pose estimation for animal behaviour prediction
2. 3D animal pose estimation as monocular 3D pose estimation (an unsolved problem), multi-view pose estimation using multiple cameras to simultaneously capture multiple photographs of animals, and 3D mesh reconstruction using data collected from RGBD cameras and 3D scanners
3. 3D animal mesh recovery, which is usually based on parameterized deformable templates

The choice of model depends significantly on the initial data. In the case when a camera trap captures several photographs (the so-called “session”) every 3-5 s, a reasonable solution is to use 2D models for single animal pose estimation. The camera trap is triggered by any movement, be it an animal, a person or branches swaying under the influence of a strong wind The selection of informative images is a complex and crucial task, especially when the set of images has been collected for more than half a year in the camera trap's storage device (Favorskaya and Buryachenko, 2019). The main sources of uninformative images are shooting artifacts caused by complex lighting and weather conditions in the wild, as well as artifacts related to the shape of animals, depending on the location relative to the camera trap. Informative scores based on production rules help to classify raw images into eleven classes, with some images accepted without processing, some images required enhancement using traditional digital image processing methods or even deep learning methods, and some images excluded from consideration (Favorskaya and Natalenko, 2024).

However, the selected images usually have a cluttered background, making it difficult to estimate the animals' poses. Our contribution is two-folded:

1. We propose a two-stage method: coarse-tuning with animal detection using a bounding box, as is done in most similar methods, and fine-tuning with semantic segmentation of animal. Semantic segmentation is preferable when the vast majority of images used for training represent a single animal. In case of multiple animals, instance segmentation should be used
2. The proposed method was tested on the following datasets: the AwA2 dataset (Xian et al., 2019), the AP-10K dataset (Yu et al., 2021), the Tiger-Pose dataset (Tiger-Pose Dataset, 2024), and own dataset collected from camera traps in the Ergaki National Park, Russia, as well as on open source images. It has been shown that semantic

segmentation improves the quality of pose estimation, especially in complex cases

The paper is structures as follows. Section 2 describes the related work. Section 3 provides the proposed method in detail. Rich experimental results are presented in Section 4, and Section 5 concludes the paper.

2. Related Work

In the past, traditional animal pose estimation methods were highly specialized and required manual intervention to correct errors. In contrast, deep learning methods achieve near human-level accuracy in almost any computer vision scenario. And this task is no exception. The use of CNNs for animal pose estimation began in 2018–2019 (Mathis et al., 2018), when markerless pose estimation based on transfer learning was first used to track different body parts of several species across a wide range of behaviors. A weakly- and semi-supervised cross-domain adaptation scheme was proposed to estimate poses of domestic four-legged mammals, including dogs, cats, horses, sheep, and cows (Cao et al., 2019). The pose-labeled animal and human samples together improved supervised keypoint estimation while minimizing the total loss as the sum of the losses in animal pose estimation and human pose estimation.

The main idea of omni-supervised joint detection and pose estimation for kangaroos in various poses was to extract training samples from unlabeled data using a "teacher" model, with the "student" model acting as a pose detector or classifier model (Zhang et al., 2020). As a result, the authors received an extended annotation for each image, indicating the position of the kangaroo (front, back, right, and left) without constructing a skeleton.

In (Kim et al., 2022), a pre-trained style transfer model was proposed to bridge the gap between human and animal domains. A "Mean Teacher" architecture was used to generate robust pseudo-labels and train on an unlabeled target region. However, unlike humans, there is a huge variety of animals with different bone lengths, number of joints and additional body parts. Therefore, transfer learning objectively cannot provide adequate results for estimating the pose of most animals. Since the 2020s, several datasets have been created with real (AP-10K) and synthetic (PASyn) annotated data for animal pose estimation, as well as a family of animal pose estimation and tracking (APT-36K, APTv2) datasets.

A lightweight and efficient stacked hourglass network model for animal pose estimation was proposed in (Zhang et al., 2023b). This network optimized the balance of model computation and accuracy based on lightweight efficient channel attention modules and lightweight dual-branch fusion module that integrated high-level semantic information and low-level detailed features.

Recently proposed MAPoseNet (animal pose estimation network via multi-scale convolutional attention) (Liu et al., 2023) utilized both semantic and spatial information by obtaining the animal's bounding box and then estimating the pose. This model was trained on the AP-10K dataset and tested on animal poses from the Macaque Pose and Animal Pose datasets, demonstrating the generalizability. ViTs were used to model the relationships between different key points or to directly extract high-resolution features. An original approach to estimating animal poses in the wild is based on an annotated dataset of 3D animal poses, which was used to refine 2D animal pose estimation (Dai et al., 2023). The cross-modal animal pose estimation (CLAMP) paradigm effectively uses prior language knowledge, such as text prompts (right eye, right knee, nose, etc.) to predict the available pose of an animal (Zhang et al., 2023a). The ScarceNet model, based on pseudo labeling using scarce annotations, adopted the following training strategy (Li and Lee, 2023). First, the animal pose estimation network was trained on a small set of labeled data, after which pseudo labels were created for the unlabeled images. Second, high-loss unlabeled samples were refined based on the agreement check. Third, the student-teacher network was trained with the reusable sample relabeling and a consistency constraint. This approach was evaluated on the AP-10K dataset and tested on the TigDog dataset.

One of the challenges of 2D pose estimation is the limited training data. There are three possible solutions: synthetic data augmentation, use of unlabeled real data with pseudo-labels, and transfer learning from existing animal or even human datasets. In this study, the last solution was applied.

Another branch of investigation is to overcome the limitations of image-based animal pose estimation by leveraging prior knowledge of animal poses in the language modality. The contrastive language-image pre-training (CLIP) model (Radford et al., 2021) based on a text encoder and an image encoder enables collaborative training in both modalities. This idea was developed in (Hu and Liu, 2024) for animal pose estimation, where text prompt templates and image feature conditional tokens are used to construct dynamic conditional prompts. Text prompts contain key points and corresponding descriptions of animal poses. Image feature conditional tokens transformed by a fully connected non-linear network are embedded into these prompts, generating so called dynamic conditional prompts. However, the limitations of this approach are the need to prepare text prompt templates and train a more complex neural model. In (Hu and Liu, 2024), a multimodal collaborative training and contrastive learning model were utilized to estimate animal poses. Multimodal training was based on text prompt templates and image feature conditional tokens using prior knowledge of animal poses in language modalities. The experiments were conducted on the AP-10K and Animal Pose datasets.

Additionally, it can be noted that modifications of the YOLO model are used in related tasks. For example, the WilDect-YOLO model was proposed in (Roy et al., 2023) as an accurate model for automated endangered wildlife detection. The WildARe-YOLO model (Bakana et al., 2024) is positioned as a lightweight wild animal recognition.

In summary, a brief overview shows that the wide variety of animal species encourages the development of more specialized deep learning models and datasets for pose estimation only, as well as for pose estimation and tracking for subsequent animal behaviour analysis.

3. The Proposed Method

Animal pose estimation faces challenges such as limited training data, extensive annotation requirements, and non-rigid projections, among others. However, estimating animal poses in the wild has additional challenges, such as cluttered background and various meteorological effects. Also, the colour of animals often correlates with the colours of the environment, which

makes it difficult to correctly determine the animal's posture. This means that semantic segmentation is an important stage in pose estimation.

A visual description of an animal can be presented at different levels of detail depending on the problem statement: at the bounding box level, at the structured level (point level, skeleton level) and at the pixel level (segmentation, semantic segmentation, instance segmentation, and panoptic segmentation as a combination of semantic and instance). Examples of segmentation types are shown in Figure 1.

a b c

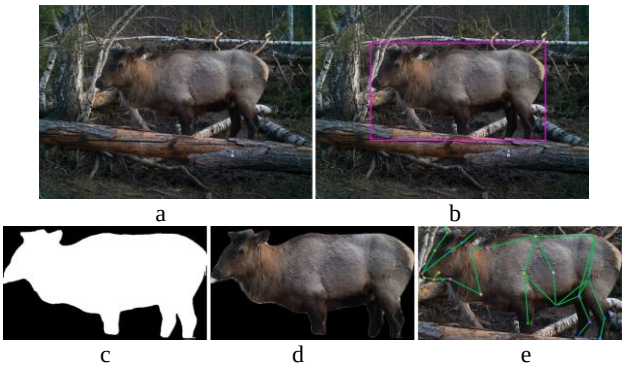
Figure 1. Different levels of animal description: **a** bounding box, **b** point level, **c** semantic segmentation.

Many previous studies have described only one type of animal. However, the difficulty of estimating the pose of animals in the wild, especially those living in the forests of Russia, requires a different approach. We decided to test not only direct training of the YOLOv8 model for animal pose estimation using an annotated dataset (coarse-tuning), but also a two-stage method when we first use a semantic segmentation model and then apply a pose builder (fine-tuning). For this purpose, the re-trained YOLOv8 Pose Estimation was applied.

We analyzed several deep semantic segmentation models such as FCN, DeepLab, and U-Net. All of them have an encoder-decoder architecture that provides different segmentation results depending on the complexity of the model, the computational cost of training and the state of the input images. At the same time, there are several frameworks and deep learning models for human pose estimation such as:

1. OpenPose is one of the most popular open-sourced frameworks for real-time and multi-person pose estimation
2. TensorFlow Pose Estimation is optimized for low-power edge devices
3. High-Resolution Net is a neural network for human pose estimation with high-resolution representations, used for human pose detection in televised sports
4. DeepCut is used for detecting the poses of multiple people in videos or images with multi-persons/objects
5. Regional Multi-Person Pose Estimation is used for detecting poses in the presence of inaccurate human bounding boxes
6. DeepPose and PoseNet are human pose estimators that use of deep neural networks
7. DensePose is a pose estimation technique for mapping 2D human pixels to the 3D surface of the human body
8. DeepLabCut is a toolbox for markerless 2D and 3D animal pose estimation
9. YOLOv8 Pose Estimation and Pose Keypoint Classification is an advanced technology that identifies and displays human body keypoints, suitable for various pose estimation tasks.

However, not all known pose estimation software tools are suitable for animal pose estimation. Thus, for experiments, we decided to use YOLOv8 Pose Estimation and Pose Keypoint Classification as animal pose estimators. An illustration to the proposed approach is depicted in Figure 2.



a b c d e

Figure 2. Proposed approach: **a** initial image, **b** bounding box and image cropping, **c** semantic segmentation mask, **d** result of semantic segmentation, **e** 2D pose.

Finally, the obtained coordinates of the corresponding animal keypoints are compared with representative examples from the pose database to determine active actions.

4. Experimental Results

The AwA2 dataset (Xian et al., 2019) was used for the experiments. It includes 10,052 images of 35 animals. However, some of these images are not suitable for our task, since giraffes and elephants are not wild inhabitants of Siberian forests. As a result, some of the images were deleted or replaced with images from a dataset taken by camera traps in the Ergaki National Park, Russia. In total, this training dataset included 4,051 images, of which 3,251 was responsible for training and 800 for validation. It should be noted that the variety of classes in the resulting data set allow to test the proposed method in working with different species of wild animals. A complete list of all animal classes is presented in Table 1.

№	Name of animal class	Number of images	№	Name of animal class	Number of images
1	Antelope	283	8	Moose	302
2	Boar	100	9	Otter	294
3	Bobcat	304	10	Rabbit	315
4	Deer	295	11	Squirrel	313
5	Fox	315	12	Tiger	296
6	Bear	601	13	Weasel	202
7	Maral	135	14	Wolf	296

Table 1. List of animal class

All images were annotated accordingly, according to a scheme using 39 key points. Each of the points was responsible for certain parts of the animal’s body. The way points are labeled and presented follows the YOLO format. A complete list of key points is presented in Table 2.

During the training process, five pre-trained models of the YOLOv8 family were used: nano, small, medium, large and x-large. All listed networks were trained under the same conditions, using the previously designated data set. All networks were trained for 75 epochs, with the amount of data fed to the model equal to 16. The plots of accuracy data when training each network are shown in Figure 3, and the plots of the loss data are presented in Figure 4.

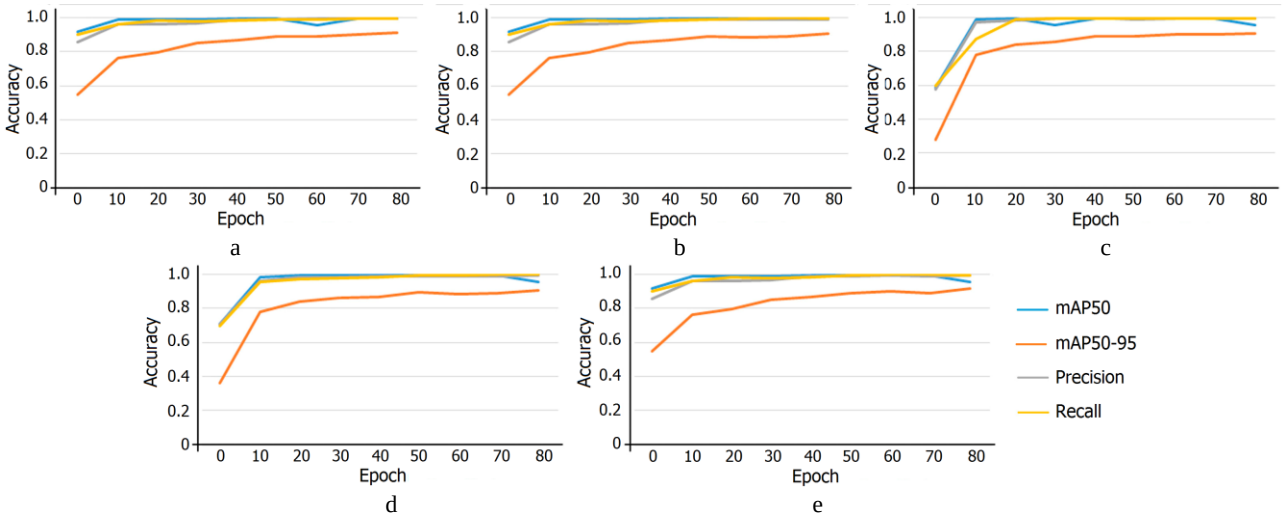


Figure 3. Accuracy plots: **a** nano model, **b** small model, **c** medium model, **d** large model, **e** x-large model.

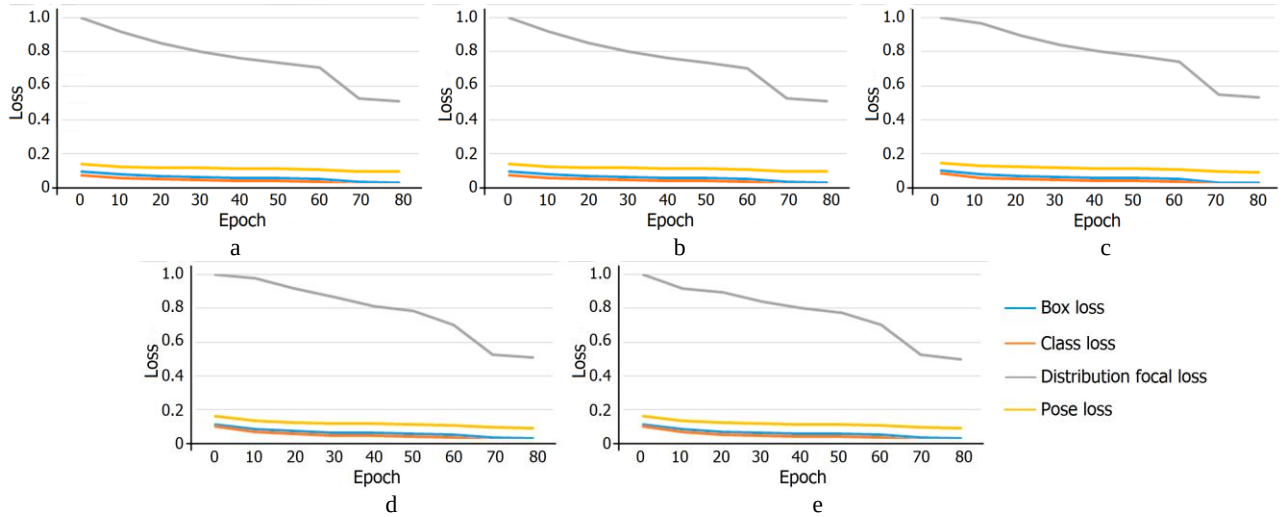


Figure 4. Training loss plots: **a** nano model, **b** small model, **c** medium model, **d** large model, **e** x-large model.

№	Name of keypoints	№	Name of keypoints
1	Nose	21	Back end
2	Upper jaw	22	Back middle
3	Lower jaw	23	Tail base
4	Mouth end right	24	Tail end
5	Mouth end left	25	Front left shoulder blade
6	Right eye	26	Front left knee
7	Right ear base	27	Front left paw
8	Right ear end	28	Front right shoulder blade
9	Right antler base	29	Front right paw
10	Right antler end	30	Front right knee
11	Left eye	31	Back left knee
12	Left ear base	32	Back left paw
13	Left ear end	33	Back left shoulder blade
14	Left antler base	34	Back right shoulder blade
15	Left antler end	35	Back right paw
16	Neck base	36	Back right knee
17	Neck end	37	Belly bottom
18	Throat base	38	Body middle right
19	Throat end	39	Body middle left
20	Back base		

Table 2. List of keypoints

Three datasets that can be found in open sources were used for the experiments. As the first dataset, we used pre-filtered data from the AwA2 dataset. Although the presented images are less representative of the reality of Siberian forests, they still have the most suitable annotation structure, thanks to which all 39 keypoints can be checked for accuracy.

The second dataset for testing is AP-10K. The full dataset contains 10,015 images collected from 60 animal species, annotated with 17 keypoints presented in Table 3. Although this dataset does not cover all the necessary points, it nevertheless has similarities in many places.

№	Name of keypoints	№	Name of keypoints
1	Left eye	10	Right Elbow
2	Right eye	11	Right Front Paw
3	Nose	12	Left Hip
4	Neck	13	Left Knee
5	Root of Tail	14	Left Back Paw
6	Left Shoulder	15	Right Hip
7	Left Elbow	16	Right Knee
8	Left Front Paw	17	Right Back Paw
9	Right Shoulder		

Table 3. List of AP-10K keypoints

The third testing set is one of the official Ultralytics network training sets. This is a relatively small set of tiger images annotated with 12 keypoints presented in Table 4.

№	Name of keypoints	№	Name of keypoints
1	Nose	7	Front right shoulder blade
2	Neck base	8	Front right paw
3	Back base	9	Back left shoulder blade
4	Back end	10	Back left paw
5	Front left shoulder blade	11	Back right shoulder blade
6	Front left paw	12	Back right paw

Table 4. List of Tiger-Pose keypoints

To validate the method, images from previously described datasets were used. Verification was performed by comparing three types of images: a manually annotated original image, an image without using segmentation, and an image using segmentation.

№	Average standard deviations, %				
	nano	small	medium	large	x-large
1	0.67/0.61	0.73/0.57	1.24/0.57	1.31/0.54	1.25/0.51
2	0.39/0.32	0.46/0.58	0.74/0.35	0.58/0.39	0.39/0.47
3	0.57/0.81	0.59/0.63	0.61/0.25	0.57/0.36	0.55/0.41
4	0.57/0.48	0.17/0.22	0.35/1.60	0.33/1.71	0.42/1.84
5	0.35/0.38	0.20/0.19	0.60/0.35	0.61/0.40	0.57/0.43
6	0.42/1.55	0.21/0.23	1.02/0.78	1.08/0.96	1.11/1.04
7	0.33/1.02	0.47/0.17	0.94/0.49	0.77/0.47	0.56/0.42
8	1.93/0.46	1.90/1.88	1.26/3.82	1.54/3.18	1.54/3.18
9	0.60/2.12	0.61/0.36	1.15/1.05	1.44/1.32	1.49/1.35
10	0.64/1.61	0.98/0.64	1.09/0.57	0.98/0.88	0.98/0.88
11	0.92/1.15	0.25/0.51	1.02/0.78	0.87/0.59	0.87/0.53
12	0.89/1.02	0.41/0.54	0.94/0.49	1.49/1.44	1.49/1.44
13	2.53/2.61	1.70/1.14	1.93/2.75	1.72/2.75	1.73/2.76
14	0.85/1.06	0.34/0.39	0.29/2.03	1.16/3.28	1.23/3.16
15	1.61/3.01	1.49/1.46	1.24/1.72	1.33/1.82	1.36/1.94
16	0.72/0.40	0.74/0.63	1.26/2.01	1.01/1.58	1.15/1.67
17	0.33/1.56	0.76/0.74	0.87/0.72	0.87/0.72	0.84/0.72
18	1.57/1.12	0.79/1.50	0.09/1.29	0.16/1.22	0.22/1.18
19	1.05/0.80	1.21/1.11	1.44/0.82	1.11/1.05	1.28/1.16
20	0.12/1.46	0.83/0.71	0.49/1.87	0.93/2.74	1.44/2.93
21	2.46/2.21	1.68/2.18	3.25/2.43	3.51/2.52	3.25/2.43
22	1.16/0.44	0.75/0.88	0.42/1.83	0.29/1.40	0.41/1.29
23	2.51/4.61	1.40/2.53	1.79/2.90	1.79/2.90	2.36/4.33
24	3.26/3.31	1.40/2.53	1.39/1.68	1.01/1.38	1.11/1.17
25	1.15/2.66	2.45/1.91	0.67/1.55	0.46/1.95	0.76/2.16
26	3.11/3.30	2.40/2.49	1.37/3.11	1.07/3.41	1.07/3.22
27	9.80/9.73	7.60/7.61	5.46/7.39	6.14/7.05	5.58/6.97
28	1.14/0.89	1.48/1.04	2.03/1.14	1.83/1.53	1.89/1.63
29	7.34/9.71	8.13/7.43	5.53/7.98	6.77/8.21	6.77/8.21
30	2.38/2.13	1.67/1.88	0.94/0.80	0.94/0.80	0.91/0.72
31	1.89/3.74	4.77/3.62	1.36/3.78	1.26/3.84	1.32/3.89
32	3.86/3.55	4.39/4.20	1.97/4.30	1.67/4.69	1.44/4.03
33	3.71/3.82	2.48/3.80	4.50/4.43	4.32/4.57	4.06/4.75
34	0.86/1.19	2.06/1.83	0.89/0.67	0.89/0.67	0.93/0.87
35	2.51/2.69	4.23/6.06	2.97/4.30	2.67/4.44	2.83/4.55
36	4.72/6.66	6.57/4.96	4.93/6.00	4.63/6.09	4.36/6.14
37	2.13/0.74	0.51/1.15	1.33/0.95	1.41/1.15	1.17/1.31
38	0.73/1.89	1.37/0.74	0.29/1.38	0.17/1.39	0.18/1.41
39	1.05/2.32	0.45/0.46	0.39/1.74	0.51/2.24	0.37/2.04

Table 5. Average standard deviations results for the AwA2 dataset

Each image was resized into a special template, and the differences between the keypoints of the processed image and the annotated template were measured. Such differences were calculated pixel by pixel and normalized to standard deviations. The average standard deviations for 39 keypoints and all five YOLOv8 models (nano, small, medium, large, and x-large) trained on the images from the AwA2 dataset are presented in Table 5. In each cell, two values present the average standard deviations of the keypoints with/without segmentation. The same goes for Tables 6-9.

As can be seen from Table 5, the nano model performed negatively, worsening post-processing results, while the x-large model performed better in terms of mean average standard deviations. In general, there is a tendency for results to improve when using a more accurate version of the model. Each subsequent model demonstrates higher accuracy, but with increased processing time. All models except nano showed positive results, that is, after applying segmentation, the average standard deviations relative to the templates decreased. The best result was shown by the x-large model. However, in all models, the keypoints of the muzzles and paws were shifted more strongly due to the implementation features of the network training.

The average standard deviations for all 5 trained models on the images from the AP-10K dataset are presented in Table 6.

№	Average standard deviations, %				
	nano	small	medium	large	x-large
1	1.12/0.87	0.99/1.49	0.63/0.83	0.57/0.80	0.58/0.77
2	0.33/0.61	0.24/0.79	0.57/0.70	0.69/0.71	0.79/0.68
3	1.80/1.33	1.29/2.29	0.87/1.11	0.98/1.19	0.95/1.39
4	2.93/3.40	2.66/2.44	1.49/3.21	1.51/3.16	1.24/3.21
5	1.11/0.46	0.56/0.74	1.10/0.78	1.46/0.89	1.66/1.02
6	5.26/6.38	5.32/5.14	6.19/5.84	6.15/5.63	6.04/5.49
7	5.45/5.93	6.09/5.74	7.03/6.79	7.29/6.77	7.17/6.81
8	11.1/11.93	11.51/11.4	11.85/11.9	8.33/11.72	7.33/11.37
9	6.15/6.20	6.44/6.16	5.86/7.10	5.86/6.95	5.47/6.89
10	9.39/8.96	8.90/8.72	9.93/11.03	8.64/10.03	9.24/10.03
11	13.2/13.05	11.47/13.2	12.70/14.8	10.59/13.7	9.99/13.74
12	3.98/3.38	3.15/4.10	4.28/3.79	3.98/3.34	3.94/3.34
13	9.40/10.36	8.87/9.49	8.86/10.66	8.90/10.08	7.86/10.66
14	1.79/0.80	0.99/1.49	0.75/1.30	0.41/1.41	0.36/1.26
15	1.15/1.17	0.87/1.89	1.89/1.41	1.17/2.22	1.35/2.13
16	6.71/6.78	6.69/6.49	5.99/6.16	5.13/6.51	5.67/6.21
17	5.35/4.81	2.61/3.92	1.42/3.85	1.74/3.19	2.22/3.91

Table 6. Average standard deviations results for the AP-10K dataset

As can be seen from testing 17 key points, the nano model again performed the worst results, while the x-large model showed the best results in terms of average standard deviations. The trend identified during the previous experiment continues. During the current experiment, all models except the nano and small models showed positive results. The most problematic areas, such as the muzzle and paws, remain the same.

The average standard deviations results for all 5 trained models on the images from the Tiger-Pose dataset are presented in Table 7. As can be seen from testing 12 key points, the trend identified during previous experiments continues. During the current experiment, none of the presented models provided a negative result, which is most likely due to an explicit decrease in the number of keypoints. The best result was shown by the x-large model.

№	Average standard deviations, %				
	nano	small	medium	large	x-large
1	0.92/1.29	1.36/1.63	2.21/1.39	1.31/2.07	1.27/2.05
2	1.00/1.89	5.88/4.37	4.41/5.10	4.43/3.55	4.27/3.69
3	1.81/3.47	1.90/3.16	3.44/2.45	2.37/3.87	2.24/3.89
4	3.18/2.61	3.57/2.41	2.86/3.44	3.42/2.84	3.27/2.54
5	2.70/2.23	3.30/2.77	2.68/2.52	1.26/2.61	1.41/2.58
6	14.83/16.6	10.89/16.7	14.6/11.50	8.57/12.25	8.24/11.06
7	5.39/6.87	4.98/6.02	7.18/5.64	5.88/7.43	5.72/7.61
8	9.27/9.17	7.83/9.02	9.63/8.08	8.01/9.79	7.03/9.85
9	3.44/4.63	5.26/4.14	4.09/4.70	5.29/4.19	5.13/4.22
10	6.77/5.63	5.79/5.28	5.66/5.04	5.55/5.87	4.44/6.01
11	8.42/8.89	6.07/8.45	7.37/6.33	3.73/4.84	4.49/5.97
12	10.68/10.7	8.86/10.89	10.71/8.66	8.58/10.90	7.68/10.03

Table 7. Average standard deviations results for the Tiger-Pose dataset

The final testing results for all datasets and models are presented in Table 8.

Dataset	Average standard deviations, %				
	nano	small	medium	large	x-large
AwA2	1.87/2.29	1.81/1.83	2.12/1.54	1.57/2.25	1.57/2.29
AP-10K	5.08/5.08	4.62/5.02	4.79/5.37	4.32/5.19	4.23/5.23
Tiger-Pose	5.70/6.17	5.47/6.24	5.34/5.40	4.87/5.85	4.60/5.79

Table 8. Final average standard deviations results

A few images were randomly selected from all used datasets to visualize the results presented. The images depicted in Figures 5-9 show positive examples, and the images depicted in Figure 10 show negative example of keypoint detection.

The dataset provided by the Ergaki National Park was used for pose estimation. Several of the most popular classes, such as deer, bear, and boar, were selected for pose estimation. The results are presented in Table 9.

Animal pose	Average accuracy of pose estimation, %		
	Deer	Bear	Boar
Standing	83.34/80.71	81.17/81.06	80.04/80.29
Sitting	–	66.71/65.91	60.36/59.88
Lying down	76.24/74.05	77.42/76.22	75.30/73.08
Moving	77.33/75.01	75.94/75,38	71.28/70.62
Eating/Searching	63.45/63.20	62.57/62.81	62.57/62.11

Table 9. Average accuracy of pose estimation.

Thus, based on the results of testing on three datasets and visual observations, we can say that the method has a positive impact on the position accuracy of determining keypoints. This method is not able to improve the results of poorly trained or less accurate models. However, in the case of accurate models, such as the medium model and above, experiments show a stable positive result.

5. Conclusions

In this study, we solve the problem of estimating the pose of animals, which helps determine the behavior of an animal in the wild. We have proposed a method of sequential image processing based on semantic segmentation and estimation of the animal's pose based on accurate detection of keypoints. The analysis of frameworks and deep learning models for human pose estimation led to the selection of the YOLOv8 Pose Estimation and Pose Keypoint Classification model, starting from the medium model and above. Experimental results show that the proposed algorithm using additional semantic segmentation increases the accuracy of animal pose estimation by 3.6-4.8% on samples of the Ergaki dataset.

References

Bakana, S.R., Zhang, Y., Twala, B., 2024. WildARe-YOLO: A lightweight and efficient wild animal recognition model. *Ecological Informatics* 80:102541.1-102541.22.

Cao, J., Tang, H., Fang, H.-S., Shen, X., Lu, C., Tai, Y.-W., 2019. Cross-domain adaptation for animal pose estimation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South), 9498-9507.

Dai, X., Li, S., Zhao, Q., Yang, H., 2023 Animal pose estimation based on 3D priors. *Appl. Sci.* 13, 1466.1-1466.17.

Favorskaya, M., Buryachenko, V., 2019. Selecting informative samples for animal recognition in the wildlife. In: Czarnowski, I., Howlett, R., Jain, L. (eds) *Intelligent Decision Technologies 2019*, SIST, vol. 143, Springer, Singapore, 65-75.

Favorskaya, M.N., Natalenko D.N., 2024. Informative evaluation of images captured by camera traps based on production rules. In: Nakamatsu, K., Patnaik, S, Kountcheva, R. (eds) *Advanced Intelligent Technologies and Sustainable Society: Proceedings of the 4th International Conference on Advanced Intelligent Techniques (ICAIT 2023)*, SIST, vol. 391, Springer, Singapore, 3-18.

Hu, X., Liu, C. 2024. Animal pose estimation based on contrastive learning with dynamic conditional prompts. *Animals* 14, 1712.1-1712.14.

Jiang, L., Lee, C., Teotia, D., Ostadabbas. S., 2022. Animal pose estimation: A closer look at the state-of-the-art, existing gaps and opportunities. *Computer Vision and Image Understanding* 222, 103483.1-103483.15.

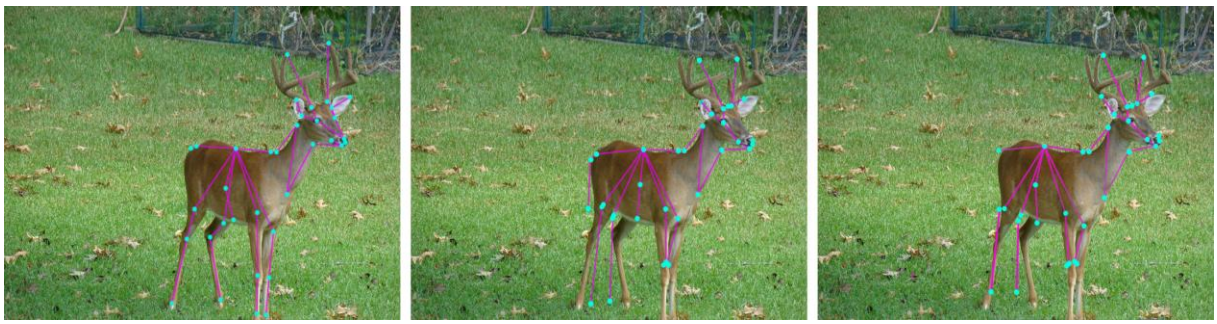
Kim, D., Wang, K., Saenko, K., Betke, M., Stan Sclaroff, S., 2022. A unified framework for domain adaptive pose estimation. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds) *Computer Vision – ECCV 2022. ECCV 2022*. LNCS, vol. 13693. Springer, Cham, 603-620.

Li, C., Lee, G.H., 2923. ScarceNet: Animal pose estimation with scarce annotations. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* Vancouver, BC, Canada, pp. 17174-17183.

Liu, S., Fan, Q., Li, S., Zhao, C., 2023. MAPoseNet: Animal pose estimation network via multi-scale convolutional attention. *J. Vis. Commun. Image R.* 97, 103989.1-103989.11.

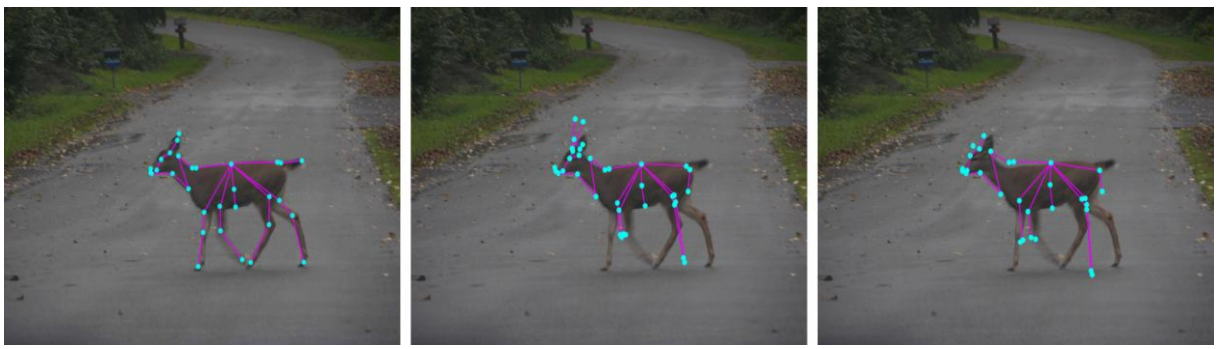
Mathis, A., Mamidanna, P., Cury, K.M., Abe, T., Murthy, V.N., Mathis, M.W., Bethge, M., 2018. DeepLabCut: Markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience* 21:1281-1289.

Roy, A.M., Bhaduri, J., Kumar, T., Raj, K. 2023. WilDect-YOLO: An efficient and robust computer vision-based accurate object localization model for automated endangered wildlife detection. *Ecological Informatics* 75:101919.1-101919.22.



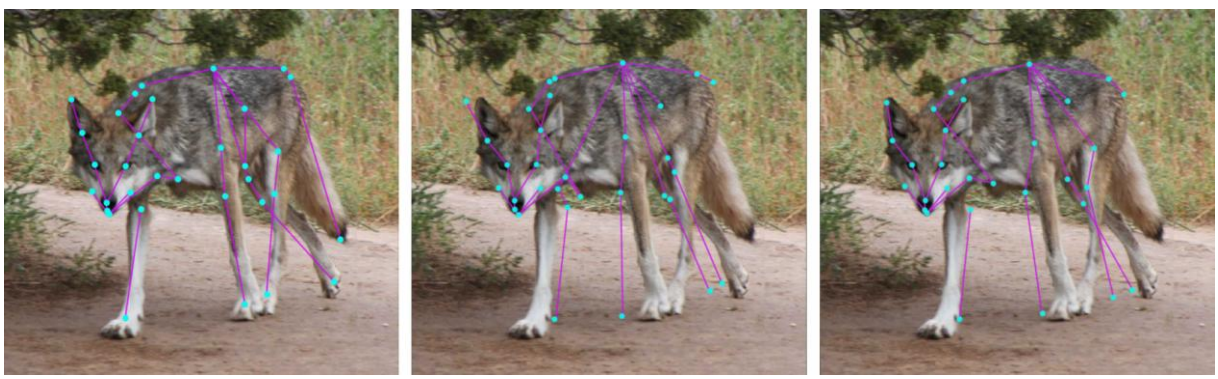
a b c

Figure 5. Visualized results of the nano model: **a** original template, **b** image without segmentation, **c** image with segmentation.



a b c

Figure 6. Visualized results of the small model: **a** original template, **b** image without segmentation, **c** image with segmentation.



a b c

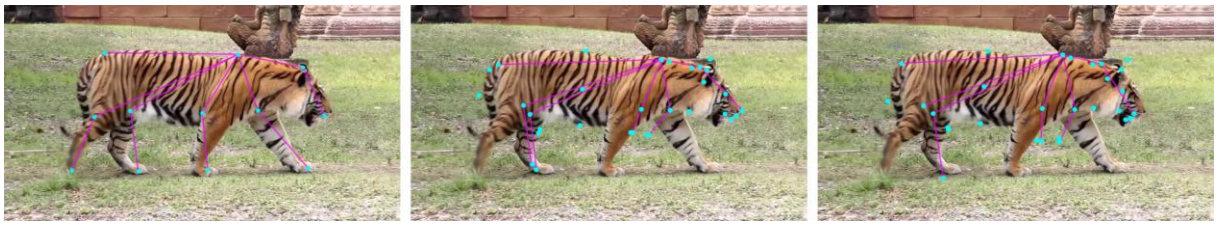
Figure 7. Visualized results of the medium model: **a** original template, **b** image without segmentation, **c** image with segmentation.

a b c

Figure 8. Visualized results of the large model: **a** original template, **b** image without segmentation, **c** image with segmentation.

a b c

Figure 9. Visualized results of the x-large model: **a** original template, **b** image without segmentation, **c** image with segmentation.



a b c

Figure 10. Negative visualized results: **a** original template, **b** image without segmentation, **c** image with segmentation.

Tiger-Pose Dataset. Ultralytics YOLO Docs https://docs.ultralytics.com/datasets/pose/tiger-pose/#introduction (14 June 2024).

Xian, Y., Lampert, C.H., Schiele, B., Akata, Z. 2019. Zero-shot learning - a comprehensive evaluation of the good, the bad and the ugly. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41(9):2251-2265.

Yu, H., Xu, Y., Zeng, J., Zhao, W., Guan, Z., Tao, D., 2021. AP-10K: A benchmark for animal pose estimation in the wild. *35th Conference on Neural Information Processing Systems (NeurIPS 2021)*, Online Conference, Canada, 1-12.

Zhang, T., Liu, L., Zhao, K., Wiliem, A., Graham Hemson, G., Lovell, B., 2020. Omni-supervised joint detection and pose estimation for wild animals. *Pattern Recognition Letters* 132:84-90.

Zhang, X., Wang, W., Chen, Z., Xu, Y., Zhang, J., Tao, D., 2023a. CLAMP: Prompt-based contrastive learning for connecting language and animal pose. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* Vancouver, BC, Canada, 23272-23281.

Zhang, W., Xu, Y., Bai, R., Li, L., 2023b. Animal pose estimation algorithm based on the lightweight stacked hourglass network. *IEEE Access* 11: 5314-5327.