Cross-Domain Vision Transformer for Animal Pose Estimation

Danni Yang[1] , Weihao Li[1] , Xunfa Lai[1]
[1]Media Analytics and Computing Lab, Department of Artificial Intelligence, School of Informatics, Xiamen University, 361005, China.
ydn_xmu@outlook.com

Abstract

Animal pose estimation(APE) can boost the understanding of animal behaviors. Vision-based animal pose estimation has attracted extensive attention in the past few years due to the advantages of contactless and senseless. The main challenge for this task is the lack of labeled data. Existing works circumvent this problem with pseudo labels generated from data of other easily accessible domains such as synthetic data. However, these pseudo labels are noisy. To address this problem, We use human pose data from the COCO keypoint detection dataset since humans share skeleton similarities with some animals. Since there are some domain gap between the human pose data and animal pose data, we further introduce the domain adaptation method to this problem. Meanwhile, with the widespread use of transformers in computer vision and the emergence of some large-scale and well-labeled animal datasets recently, we have been inspired to apply the vision transformer in the APE task. Consequently, we combine the vision transformer and cross-domain method to improve the accuracy and generalization ability of our model. We evaluate our approach on the AP-10K and Animal-Pose datasets. Experiments show that our proposed method can achieve convincing results on animal pose estimation.

Introduction

Animal pose estimation aims to predict the locations of animal body parts and joints from images or videos. APE has received increasing attention from scholars over the last few years. Parsing animal posture can help to promote the understanding of animal behaviors, which is the foundation of some disciplines such as biomechanics, neuroscience, and behavior. For the APE task, we need to detect the keypoint of the main parts of the animals and then output the location parameters of the keypoint. The image is preprocessed, then used as the input of the pose estimation module to perform feature extraction and fusion in the pose and keypoint prediction. Next, through the post-processing module the final output is the keypoint information of the animal.

In our work, we design our method from two perspectives. The first is to improve the feature extraction ability of the backbone network, and the second is to increase the animal species generalization ability of the model.

Convolutional neural network has become the dominant method for most visual tasks since 2012. However, with the advent of more efficient structures and convergence of computer vision and natural language processing, using the transformer for visual tasks(Dosovitskiy et al. 2020) has become a new research direction which can achieve better performance than CNNs. And the purpose of the transformer is to reduce the complexity of the structure, and explore scalability and training efficiency. ViTPose(Xu et al. 2022) recently has achieved the best results for human pose estimation. Specifically, ViTPose employs plain and non-hierarchical vision transformers as backbones to extract features for a given person instance and a lightweight decoder for pose estimation. However, to the best of our knowledge, there is no attempt to apply transformer backbone to APE task until now. In this paper, in a bid to achieve better accuracy and generalization ability, we apply the vision transformers to APE task.

Although human pose estimation has made great progress in recent years, due to the lack of large annotated animal datasets and the existence of animal species diversity, directly transfer human pose estimation algorithms to animal datasets usually fails to achieve good accuracy and generalization capabilities. To solve this problem, most of the existing methods adopt the cross-domain approach(Cao et al. 2019; Mu et al. 2020; Li et al. 2021), that is, we can transfer knowledge from other more readily available domains such as human data to the real target domain.

We evaluate our approach on the AP-10K and Animal-Pose datasets. Experiments show that our proposed method can achieve convincing results. Conclusively, the main contributions of our work are as below:

- We design a simple but efficient encoder and decoder structure based on vision transformer, which consists of a feature extractor, a domain discriminator and a keypoint estimator.
- We propose a joint learning strategy and cross-domain method which can transfer knowledge from human domain to real animal domain to achieve better accuracy and generalization ability.
- Our approach achieves state-of-the-art results on the AP-10K dataset and the Animal-Pose dataset, verifying the effectiveness of our proposed approach.

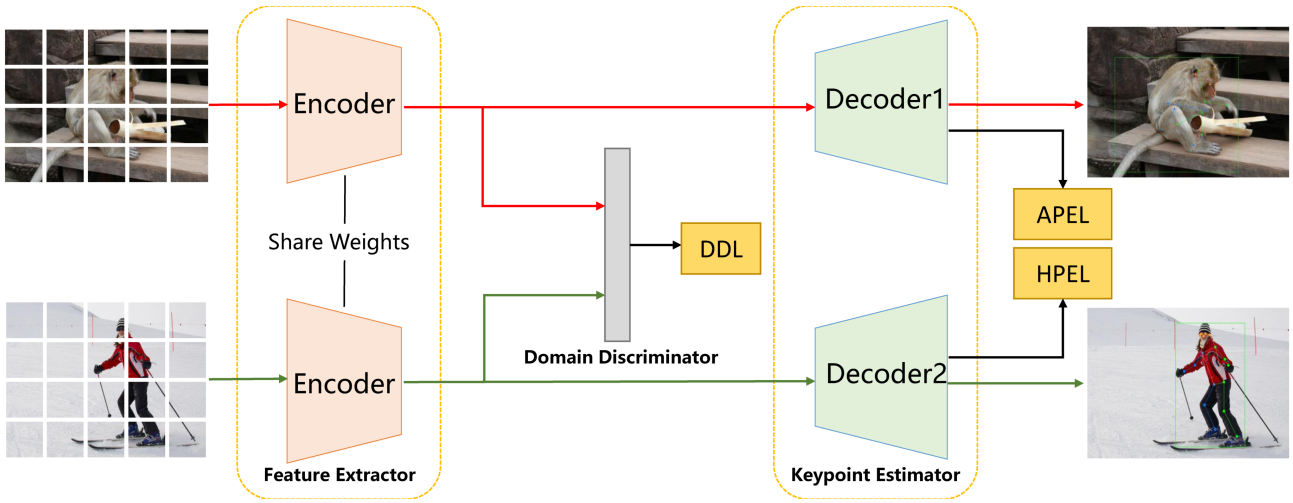

Figure 1: Pipeline of our model. Lines with color describe the flow of features along different paths. Specifically, the red line represents the flow of animal data, and the green line represents the flow of human data. "DDL" indicates the domain discrimination loss. "APEL" and "HPEL" indicate animal/human pose estimation loss respectively. The cooperation of keypoint estimator and domain discriminator does not just improves the pose estimation capacity on pose-labeled samples but also forces the model to gain this through better extracting and leveraging common features shared by pose-labeled and pose-unlabeled samples.

Related Work

Human pose estimation

Human pose estimation aims at predicting the poses of human body parts from images or videos. Since pose motions are often driven by some specific human actions, knowing the body pose of a human is critical for action recognition. One of the early approaches(Sapp, Jordan, and Taskar 2010) is the pictorial structure which uses a tree structure to model the spatial relationships among body parts. However, these methods do not perform well in complex scenarios because of the limited representation capabilities. In the past few years, the rise of deep neural models(Newell, Yang, and Deng 2016; Xiao, Wu, and Wei 2018; Sun et al. 2019) based on CNN has improved the results but brings data hunger to develop a high-powered model. Recently, Xu et al.(Xu et al. 2022) applied the vision transformer in human pose estimation and achieved good results.

Animal pose estimation

Animal pose estimation is relatively under-explored compared to human pose estimation mainly due to the lack of labeled data. To solve this problem, Cao et al.(Cao et al. 2019) propose a cross-domain adaptation scheme to learn a shared feature space between human and animal images such that their network can learn from existing human pose datasets. They also select pseudo labels into the training based on the confidence score. Mu et al.(Mu et al. 2020) use synthetic animal data generated from CAD models to train their model, which is then used to generate pseudo labels for the unlabeled real animal images. Domain adaptation becomes very difficult when domains face severe domain shift and no extra information is available to align feature representation on different domains. We propose a cross-domain method for animal pose estimation based on vision transformer.

Our Method

Recently, most visual tasks has experienced rapid development from CNNs to the vision transformer networks. Transformer has been used in the human pose estimation (Xu et al. 2022) with competitive performance. In this paper, we first use the vision transformer structure to propose a simple yet effective baseline model for animal pose estimation. Due to the lack of pose-labeled animal datasets, we randomly sample 10K samples from the MS COCO Keypoint Detection dataset for training with the AP-10K animal dataset together to train a strong representational encoder backbone. Then we design two decoders to estimate the heatmaps of the keypoints. But there exists domain shift between the source domain and target domain, so we further design a simple and efficient cross-domain method to solve this problem. Specifically, we add a domain discriminator which is a fully-convolutional architecture to reduce the domain gap. Our model architecture is shown in Figure1. In the following section we will describe our approach in detail.

Vision Transformer Architecture

Inspired by the great success of ViTPose in human pose estimation, we expect to use this structure in our work. Concretely, we use the encoder based on the vision transformer to extract features of the input images. And for the decoder part, we use simple bilinear layer and a prediction layer, as in(Xiao, Wu, and Wei 2018).

Encoder As Figure 2 shown, our encoder consists of several transformer blocks. Specifically, given an animal in-

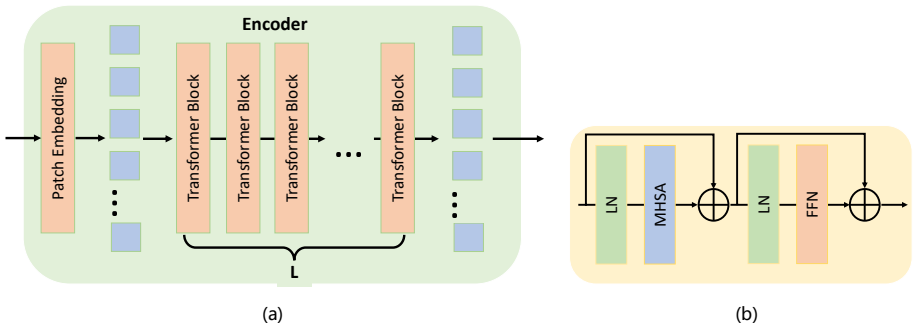

Figure 2: (a) The framework of Encoder. (b) The transformer block

stance image $X \in \mathcal{R}^{\mathcal{H}\times\mathcal{W}\times 3}$ as input, Our model first embeds the images into tokens via a patch embedding layer $F \in \mathcal{R}^{\frac{H}{d}\times\frac{W}{d}\times C}$, where d equal to 16 by default, and C is the channel dimension. After that, the embedded tokens are processed by several transformer layers, each of which is consisted of a multi-head self-attention(MHSA) layer and a feed-forward network(FFN), i.e.,

$$F'_{i+1} = F_i + \mathrm{MHSA}\left(\mathrm{LN}\left(F_i\right)\right) \tag{1}$$

$$F_{i+1} = F'_{i+1} + \mathrm{FFN}\left(\mathrm{LN}\left(F'_{i+1}\right)\right) \tag{2}$$

where i represents the output of the ith transformer layer and the initial feature $F_0 = \mathrm{PatchEmbed}(X)$ denotes the features after the patch embedding layer. According to the original design of the vision transformer, the spatial and channel information of each block input and output remains unchanged, so the output of the final encoder can be expressed as $F_{out} \in \mathcal{R}^{\frac{H}{d}\times\frac{W}{d}\times C}$.

Decoder As to the decoder, to minimize time consumption, we use two simple decoders to process the human or animal features extracted from the backbone network and localize the keypoints.

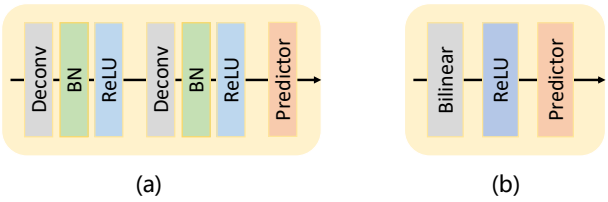

Figure 3: (a) The classic decoder. (b) The simple decoder.

As shown in Figure 3, (a) is a classic decoder, which is composed of two deconvolution blocks. Each of the block contains one deconvolution layer followed by batch normalization(Ioffe and Szegedy 2015) and ReLU (Agarap 2018), Following the common setting of previous methods(Zhang, Chen, and Tao 2021), each block upsamples the feature maps by 2 times. Then, a convolution layer with the kernel size 1 × 1 is utilized to get the localization heatmaps for the keypoints.

$$K = \mathrm{Conv}_{1\times 1}\left(\mathrm{Deconv}\left(\mathrm{Deconv}\left(F_{\text{out}}\right)\right)\right) \tag{3}$$

Apart from this, as Figure 3 (b) shown, we try to use another simpler decoder in our methods, thanks to the strong representation ability of the vision transformer backbone, this decoder also is proved effective. Specifically, we directly upsample the feature maps by 4 times with bilinear interpolation, followed by a ReLU and a convolution layer with the kernel size 3 × 3 to get the heatmaps, i.e.,

$$K = \mathrm{Conv}_{3\times 3}\left(\text{ Bilinear }\left(\mathrm{ReLU}\left(F_{\text{out}}\right)\right)\right) \tag{4}$$

Joint learning It is worth mentioning that the data is mixed during the training process. We randomly sample instances from multiple training datasets for each iteration and feed them into the backbone and the different decoders to estimate the heatmaps corresponding to each dataset. Animal Pose Estimation Loss(APEL) and Human Pose Estimation Loss (HPEL), the loss function of HPEL and APEL respectively and are usually both mean-square error. The overall loss for pose estimation is as follows,

$$\mathcal{L}_{\text{pose}} = \sum_{i=1}^{N}\left(w_2 y_i \mathcal{L}_A\left(I_i\right) + \left(1-y_i\right)\mathcal{L}_H\left(I_i\right)\right) \tag{5}$$

where L_H and L_A indicate loss function of HPEL and APEL respectively, w_2 is weighting factor to alleviate the effect of dataset volume gap. To overcome the imbalance between human samples and animal samples, w_2 should be set to a larger value, otherwise model tends to perform almost equivalent to only trained on human samples.

Cross Domain Method

As Figure 1, to bridge the domain gap between the source and target domains, we train a domain discriminator to classify the obtained features, which is a fully-convolutional network. The domain discriminator attempts to classify the real target data from the synthetic source data using a cross-entropy loss to construct the domain discrimination loss(DDL):

$$\begin{aligned}\mathcal{L}_{DDL} = &- w_1 \sum_{i=1}^{N}\left(y_i \log\left(\hat{y}_i\right) + \left(1-y_i\right)\log\left(1-\hat{y}_i\right)\right) \\ &- \sum_{i=1}^{N} y_i\left(z_i \log\left(\hat{z}_i\right) + \left(1-z_i\right)\log\left(1-\hat{z}_i\right)\right)\end{aligned} \tag{6}$$

where y_i indicates whether x_i is a human or animal sample(y_i = 1 for animals and y_i = 0 for human); z_i indicates whether x_i comes from the target domain (z_i = 1 if it is pose-unlabeled sample and otherwise z_i = 0). $\hat{y}_i$ and $\hat{z}_i$ are predictions by the domain discriminator. w_1 is a weighting factor.

Total Loss Functions

The losses of domain discriminator and keypoint estimator are set to be adversarial. As pose estimation is the main task, the domain discriminator serves for domain confusion during feature extraction. Through this design, the model is expected to perform better on pose-unlabeled samples by leveraging better features that are shared on domains. The total loss of the model is formulated as:

$$\mathcal{L}_{total} = \alpha\mathcal{L}_{DDL} + \beta\mathcal{L}_{\text{pose}} \tag{7}$$

Since domain discriminator and keypoint estimator are adversarial, so we need to set $\alpha * \beta < 0$, which encouraging domain confusion and boosting pose estimation performance at the same time. In the experiments, we set $\alpha = 1, \beta = -0.002$.

Experiment

Datasets

AP-10K The dataset(Yu et al. 2021) is the first large-scale benchmark for mammal animal pose estimation, which consists of 10,015 images collected and filtered from 23 animal families and 54 species following the taxonomic rank and high-quality keypoint annotations labeled and checked manually. We train our model on the AP-10K dataset. The definition of the skeleton shown in table 1.

MS COCO Keypoint Detection The COCO dataset(Lin et al. 2014)contains over 200, 000 images and 250, 000 person instances labeled with 17 keypoints. We randomly sample 10K samples from the MS COCO Keypoint Detection dataset for training with the animal dataset together.

Animal-Pose Dataset This dataset(Cao et al. 2019)contains annotations for five animal categories:dog, cat, horse, sheep and cow. 5,517 instances of these 5 categories are distributed in more than 3,000 images. We use this dataset to test the generalization capacity of our model.

Table 1: The definition of the skeleton joint in quadrupeds.

Keypoint	Definition	Keypoint	Definition
0	Left Eye	9	Right Elbow
1	Right Eye	10	Right Front Paw
2	Nose	11	Left Hip
3	Neck	12	Left Knee
4	Root of tail	13	Left Back Paw
5	Left Shoulder	14	Right Hip
6	Left Elbow	15	Right Knee
7	Left Front Paw	16	Right Back Paw
8	Right Shoulder		

Implementation Details

Experimental Settings We benchmark several representative pose estimation frameworks with different CNN backbone networks based on the MMPose codebase (Contributors 2020). Two A100 GPUs with 40GB memory is used during both the training and testing for all the experiments. Our backbones are initialized with MAE(He et al. 2022) pre-trained weights. We use the 256 × 256 input resolution and AdamW(Reddi, Kale, and Kumar 2019) optimizer with a learning rate of 5e-4. Udp(Huang et al. 2020)is used for post-processing. The models are trained for 210 epochs with a learning rate decay by 10 at the 170th and 200th epoch.

Metrics

Average Precision and Recall Scores In the basic experiment part we use standard evaluation metric which is based on Object Keypoint Similarity (OKS):

$$OKS = \frac{\sum_i exp(-d_i^2/2s^2K_i^2)\delta(v_i > 0)}{\sum_i \delta(v_i > 0)} \tag{8}$$

where d_i is the Euclidean distance between the detected keypoint and the corresponding ground truth, v_i is the visibility flag of the ground truth, s is the object scale, and k_i is a per-keypoint constant that controls falloff. We report standard average precision and recall scores: AP(the mean of AP scores at 10 positions, OKS = 0.50, 0.55,..., 0.90, 0.95), AP^{50}(AP at OKS = 0.50); AR at OKS = 0.50, 0.55,..., 0.90, 0.955, AR^{50}(AR at OKS = 0.50).

Percentage of Correct Keypoints (PCK) We adopt the percentage of correct keypoints (PCK) as the evaluation metric in generalization test. PCK measures the accuracy between the predicted joint location and the true joint location.

$$PCK_i^k = \frac{\sum_p \delta\left(\frac{d_{pi}}{d_p^{\text{def}}} \leq T_k\right)}{\sum_p 1} \tag{9}$$

where i denote ith keypoint in image, T_k denotes kth threshold, range from 0 to 1, p denote pth target, d_pi denote the distance between predicted and ground truth of ith keypoint for target p, d_p^{def} denote the scale factor of pth target. In this work T_k is 0.05 and d_p^{def} is the size of heatmap.

Quantitative Analysis

Comparison with the methods based on CNNs We compared the HRNet and SimpleBaseline on the AP-10K dataset, as shown in Table 2.We can see that our method outperforms the previous convolutional methods by a large margin. Specifically, at both 256x256 image resolutions, our approach achieved 77.12% AP and 80.32% AR.

Ablation Study To verify the contribution of our proposed joint-learning and cross-domain module respectively, we conduct the ablation study. In Table 3, "JL" means that we use joint-learning module for training on COCO dataset and AP-10K dataset. We can find that AP and AR score have decreased. We analyze that this is because there are some domain gap between human pose data and the animal pose data. "CD" means adding the domain discrimintor.

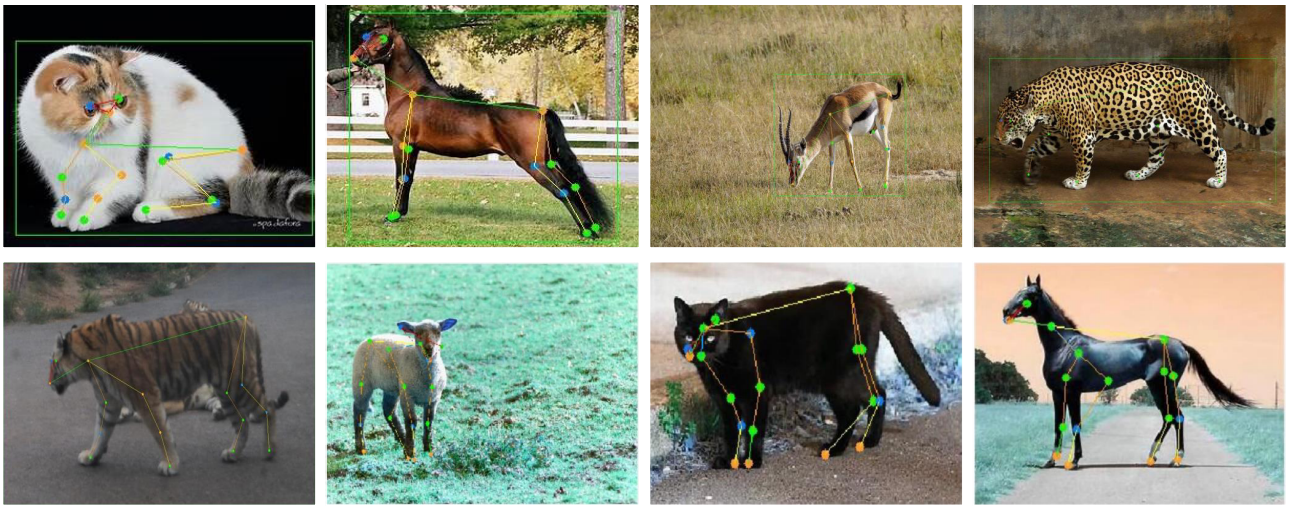

Figure 4: Competitive results on AP-10K dataset and Animal-Pose dataset.

Table 2: Comparison with HRNet and SimpleBaseline on the AP-10K test set.

Model	Backbone	Params(M)	Resolution	AP	AP^{50}	AR	AR^{50}
HRNet	HRNet-w32	28.54	256x256	72.46	94.24	75.81	94.95
HRNet	HRNet-w48	63.59	256x256	72.95	94.28	76.28	95.04
SimpleBaseline	ResNet50	33.99	256x256	67.96	91.92	71.68	92.88
SimpleBaseline	ResNet101	52.99	256x256	68.25	92.01	71.78	92.95
Ours	ViT-B	104.08	256x256	**77.12**	**96.24**	**80.32**	**97.12**

Compared with only using ViT, our method with JL moudle and CD moudle has improved by 1.14% AP and 1.06% AR, which verified the effectiveness of our cross-domain method when we try to use joint-learning.

Table 3: Ablation study of joint-learning(JL) and cross-domain(CD).

ViT	JL	CD	AP	AP^{50}	AR	AR^{50}
✔			75.98	95.42	79.26	96.11
✔	✔		75.22	95.72	78.71	96.31
✔	✔	✔	**77.12**	**96.24**	**80.32**	**97.12**

Generalization tests on Animal-Pose Dataset Table 4 shows that other cross-domain methods perform poorly on the unseen species like dog, cat and sheep. Our method perform the best on the five categories which indicates that our method has top-performing generalization abilities.

Qualitative Analysis

Visualization Figure 4 shows some competitive results on AP-10K dataset and Animal-Pose dataset. It can be seen that the results of our method are accurate and have good generalization, which can accurately predict the keypoints for a variety of animals with different poses.

Table 4: PCK@0.05 accuracy for the generalization compared with CC-SSL and UDA on Animal-Pose test dataset.

	Horse	Dog	Cat	Sheep	Cow	Mean
CC-SSL	65.35	30.27	15.05	52.39	63.71	47.6
UDA	72.84	42.48	27.65	59.51	71.31	56.77
Ours	**86.86**	**77.01**	**69.36**	**79.11**	**88.26**	**81.53**

Conclusion

In this paper, we propose an approach for cross-domain animal pose estimation based on vision transformer. We changed the backbone network for extracting features from the traditional convolutional network to a visual transformer, which can achieve very good performance. In addition, in order to approach the problem of insufficient animal datasets and the diversity of animal species, we propose a framework for joint training of multiple datasets, that is, to perform mixed training on human pose data and animal pose data. At the same time, to bridge the gap between different domains, we introduce a domain classifier to narrow the gap between the human domain and the animal domain, resulting the model insensitive to species, with stronger generalization ability and robustness. In the future, we will further explore different domain generalization methods on larger datasets to better improve the performance of the model.

References

Agarap, A. F. 2018. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*.

Cao, J.; Tang, H.; Fang, H.-S.; Shen, X.; Lu, C.; and Tai, Y.-W. 2019. Cross-Domain Adaptation for Animal Pose Estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Contributors, M. 2020. Openmmlab pose estimation toolbox and benchmark.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16000–16009.

Huang, J.; Zhu, Z.; Guo, F.; and Huang, G. 2020. The devil is in the details: Delving into unbiased data processing for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5700–5709.

Ioffe, S.; and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, 448–456. PMLR.

Li, B.; François-Lavet, V.; Doan, T.; and Pineau, J. 2021. Domain adversarial reinforcement learning. *arXiv preprint arXiv:2102.07097*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, 740–755. Springer.

Mu, J.; Qiu, W.; Hager, G. D.; and Yuille, A. L. 2020. Learning from synthetic animals. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12386–12395.

Newell, A.; Yang, K.; and Deng, J. 2016. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, 483–499. Springer.

Reddi, S. J.; Kale, S.; and Kumar, S. 2019. On the convergence of adam and beyond. *arXiv preprint arXiv:1904.09237*.

Sapp, B.; Jordan, C.; and Taskar, B. 2010. Adaptive pose priors for pictorial structures. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 422–429. IEEE.

Sun, K.; Xiao, B.; Liu, D.; and Wang, J. 2019. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5693–5703.

Xiao, B.; Wu, H.; and Wei, Y. 2018. Simple baselines for human pose estimation and tracking. In *Proceedings of the European conference on computer vision (ECCV)*, 466–481.

Xu, Y.; Zhang, J.; Zhang, Q.; and Tao, D. 2022. ViTPose: Simple Vision Transformer Baselines for Human Pose Estimation. *arXiv preprint arXiv:2204.12484*.

Yu, H.; Xu, Y.; Zhang, J.; Zhao, W.; Guan, Z.; and Tao, D. 2021. Ap-10k: A benchmark for animal pose estimation in the wild. *arXiv preprint arXiv:2108.12617*.

Zhang, J.; Chen, Z.; and Tao, D. 2021. Towards high performance human keypoint detection. *International Journal of Computer Vision*, 129(9): 2639–2662.