ImageNet performance correlates with pose estimation robustness and generalization on out-of-domain data

Alexander Mathis [1] **Thomas Biasi** [1] **Mert Yüksekgönül** [1 2] **Byron Rogers** [3] **Matthias Bethge** [2 4]
Mackenzie Weygandt Mathis [1 4]

Abstract

Neural networks are highly effective tools for pose estimation. However, robustness to out-of-domain data remains a challenge, especially for small training sets that are common for real-world applications. Here, we probe the generalization ability with three architecture classes (MobileNetV2s, ResNets, and EfficientNets). We developed a novel dataset of 30 horses that allowed for both "within-domain" and "out-of-domain" (unseen horse) benchmarking - this is a crucial test for robustness that current human pose estimation benchmarks do not directly address. We show that better ImageNet-performing architectures perform better on both within- and out-of-domain data if they are first pretrained on ImageNet. Our results demonstrate that transfer learning is beneficial for out-of-domain robustness.

Pose estimation is an important tool for measuring behavior, and thus widely used in technology, medicine and biology (Ostrek et al., 2019; Maceira-Elvira et al., 2019; Mathis & Mathis, 2020). Due to innovations in both deep learning algorithms (Insafutdinov et al., 2017; Cao et al., 2017; He et al., 2017; Kreiss et al., 2019; Ning et al., 2020; Cheng et al., 2020) and large-scale datasets (Lin et al., 2014; Andriluka et al., 2014; 2018) pose estimation on humans has gotten very powerful. However, typical human pose estimation benchmarks, such as MPII pose and COCO (Lin et al., 2014; Andriluka et al., 2014; 2018), contain many different individuals ($> 10k$) in different contexts, but only very few example postures per individual. In real world application of pose estimation, users want to estimate the location of user-defined bodyparts by only labeling a few hundred frames on a small subset of individuals, yet want this to generalize to new individuals (Ostrek et al., 2019; Maceira-Elvira et al., 2019; Mathis & Mathis, 2020). Thus, one naturally asks the following question: Assume you have trained an algorithm that performs with high accuracy on a given (individual) animal for the whole repertoire of movement - how well will it generalize to different individuals that have slightly or dramatically different appearances? Unlike in common human pose estimation benchmarks, here the setting is that datasets have many (annotated) poses per individual (>200) but only a few individuals (≈ 10).

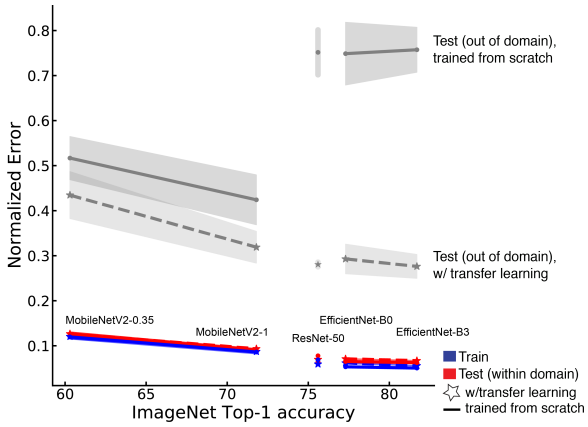

Figure 1. **Transfer Learning boosts performance, especially on out-of-domain data.** Normalized pose estimation error vs. ImageNet Top 1% accuracy with different backbones. While training from scratch reaches the same task performance as fine-tuning, the networks remain less robust as demonstrated by poor accuracy on out-of-domain horses. Mean ± SEM, 3 shuffles.

To allow the field to tackle this challenge, we developed a novel benchmark comprising 30 diverse Thoroughbred horses, for which 22 body parts were labeled by an expert in $8,114$ frames (Dataset available at `horse10.deeplabcut.org`). Horses have various coat colors and the "in-the-wild" aspect of the collected data at various Thoroughbred farms added additional complexity (Figure 2). With this dataset we could directly test the effect of pretraining on out-of-domain data. Here we report two key insights: (1) ImageNet performance predicts generalization for both within domain and on out-of-domain data for pose estima-

[1]Harvard University, Cambridge, MA USA [2]University of Tübingen, Tübingen, Germany [3]Performance Genetics [4]European Lab for Learning & Intelligent Systems (ELLIS). Correspondence to: Alexander Mathis <alexander.mathis@epfl.ch>.

Presented at the ICML 2020 Workshop on Uncertainty and Robustness in Deep Learning.

Figure 2. **Horse Dataset:** Example frames for each Thoroughbred horse in the dataset. The videos vary in horse color, background, lighting conditions, and relative horse size. The sunlight variation between each video added to the complexity of the learning challenge, as well as the handlers often wearing horse-leg-colored clothing. Some horses were in direct sunlight while others had the light behind them, and others were walking into and out of shadows, which was particularly problematic with a dataset dominated by dark colored coats. To illustrate the horse-10 task we arranged the horses according to one split: the ten leftmost horses were used for train/test within-domain, and the rest are the out-of-domain held out horses.

tion (Figure 1); (2) While we confirm that task-training can catch up with fine-tuning pre-trained models given sufficiently large training sets (He et al., 2018), we show this is not the case for out-of-domain data (Figure 4). Thus, transfer learning improves robustness and generalization.

1. Related Work

Transfer learning has become accepted wisdom: fine-tuning pretrained weights of large scale models yields best results (Donahue et al., 2014; Yosinski et al., 2014; Kümmerer et al., 2016; Mathis et al., 2018; Li et al., 2019; Zhuang et al., 2019). He et al. nudged the field to rethink this accepted wisdom. They demonstrated for various tasks that directly training on the task-data can match performance (He et al., 2018). We confirm this result, but show that on held-out individuals ("out-of-domain") this is not the case. Raghu et al. showed that for target medical tasks (with little similarity to ImageNet) transfer learning offers little benefit over lightweight architectures (Raghu et al., 2019). Kornblith et al. showed for many object recognition tasks, that better ImageNet performance leads to better performance on these other benchmarks (Kornblith et al., 2019). We show that this is also true for pose-estimation both for within-domain and out-of-domain data. Two more recent papers relate to our work. Hendrycks et al. study robustness to out-of distribution data on CIFAR 10, CIFAR 100 and TinyImageNet (but not pose estimation). The authors report that pretraining is important for adversarial robustness (Hendrycks et al., 2019). Shah et al. highlight that pose estimation algorithms are highly robust against adversarial attacks, but do not directly test out-of-domain robustness nor performance on smaller real-world sized datasets (Shah et al., 2019). This work substantially expands our earlier preprint (Mathis et al., 2019).

2. Methods

2.1. Horse Dataset and evaluation metrics

Here we developed a novel horse data set comprising $8,114$ frames across 30 different horses captured for $4-10$ seconds with a GoPro camera (Resolution: 1920×1080, Frame Rate: 60 FPS), which we call Horse-30. We downsampled the frames by a factor of 15% to speed-up the benchmarking process (288×162 pixels; one video was downsampled to 30%). We annotated 22 previously established anatomical landmarks (see Methods) for equines (Magnusson & Thafvellin, 1990; Anderson & McIlwraith, 2004). We created 3 splits that contain 10 randomly selected training horses each (referred to as Horse-10). For each training set we took a subset of 5% (≈ 160 frames), and 50% ($\approx 1,470$ frames) of the frames for training, and then evaluated the performance on the training, test, and unseen (defined as "out-of-domain") horses (i.e. the other horses that were not in the given split of Horse-10). As the horses could vary dramatically in size across frames, due to the "in-the-wild" variation in distance from the camera, we normalized the raw pixel errors by the eye-to-nose distance and report the fraction of this distance (normalized error) as well as percent correct keypoint metric (Andriluka et al., 2014); we used a matching threshold of 30% of the nose to eye distance.

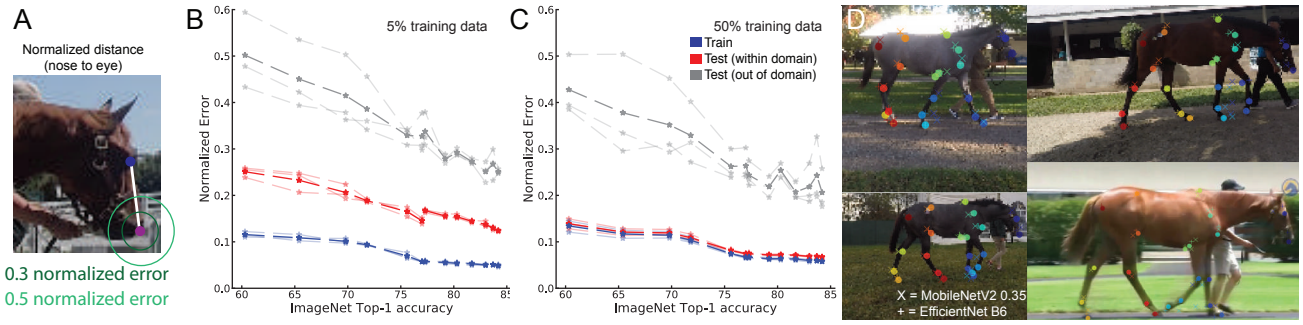

Figure 3. **Transfer Learning boosts performance, especially on out-of-domain data. A:** Normalized Error vs. Network performance as ranked by the Top 1% accuracy on ImageNet (order by increasing ImageNet performance: MobileNetV2-0.35, MobileNetV2-0.5, MobileNetV2-0.75, MobileNetV2-1, ResNet-50, ResNet-101, EfficientNets B0 to B6). The faint lines indicate data for the three splits. Test data is in red, train is blue, grey is out-of-domain data **B:** Same as A but with 50% training fraction. **C:** Example frames with human annotated body parts vs. predicted body parts for MobileNetV2-0.35 and EfficientNet-B6 architectures with ImageNet pretraining on out-of-domain horses.

2.2. Architectures and Training Parameters

For this study we adopted a pose estimation toolbox called DeepLabCut (Mathis et al., 2018; Nath et al., 2019) by adding MobileNetV2 (Sandler et al., 2018) and EfficientNet backbones (Tan & Le, 2019) to the ResNets (He et al., 2016) that were present (see Appendix). For training, a cosine learning rate schedule, as in Kornblith et al., 2019 with ADAM optimizer and batchsize 8 was used. Initial learning rates and decay target points were cross-validated for MobileNetV2-0.35 and -1.0, ResNet-50, EfficientNet B0, B3, and B5 for the pretrained and from scratch models (see Appendix). For each model that was not cross validated (MobileNetV2 0.5 and 0.75, ResNet-101, EfficientNet B1, B2, B4, B6), the optimal training parameters from the most similar cross validated model was used (i.e. the cross validated EfficientNet-B0 schedule was used for EfficientNet-B1; see Methods). For MobileNetV2s, we trained the batch normalization too (this had little effect on task performance for MobileNetV2-0.35). Pretrained models were trained for 30k iterations (as they converged), while models from scratch were trained for 180k iterations.

3. Results

To test within and out-of-domain performance we created a new dataset of 30 different Thoroughbreds that are led by different humans, resulting in a dataset of 8,114 images with 22 labeled body parts each. These videos differ strongly in horse appearance, context, and background (Figure 2). Thus, this dataset is ideal for testing robustness and out-of-sample generalization. We created 3 splits containing 10 random horses each, and then varied the amount of training data from these 10 horses (referred to as Horse-10, see Methods). As the horses could vary dramatically in size across frames, due to the "in-the-wild" variation in distance from the camera, we used a normalized pixel error; i.e. we normalized the raw pixel errors by the eye-to-nose distance and report the fraction within this distance (see Methods).

Table 1. average PCK@0.3 (%)

MODELS	WITHIN DOMAIN	OUT-OF-D.
MOBILENETV2-0.35	95.2	63.5
MOBILENETV2-0.5	97.1	70.4
MOBILENETV2-0.75	97.8	73.0
MOBILENETV2-1	98.8	77.6
RESNET-50	99.8	81.3
RESNET-101	99.9	84.3
EFFICIENTNET-B0	99.9	81.6
EFFICIENTNET-B1	99.9	84.5
EFFICIENTNET-B2	99.9	84.3
EFFICIENTNET-B3	99.9	86.6
EFFICIENTNET-B4	99.9	86.9
EFFICIENTNET-B5	99.9	87.7
EFFICIENTNET-B6	99.9	88.4

To probe the impact of ImageNet performance on pose estimation robustness, we selected modern architectures as backbones with a wide range of ImageNet performance (see Methods; 13 models spanning from 60% to 84% ImageNet performance). To fairly compare the MobileNetV2, ResNets and EfficientNet backbones, we cross validated the learning schedules for each model (see Methods). In total, we found that all ImageNet pretrained architectures exhibited strong performance on Horse-10 within domain, i.e. low average errors, and high average percent correct key points (aPCK; Figure 3, Table 1). Next, we directly compared the ImageNet performance to their respective performance on this pose estimation task. We found Top-1% ImageNet accuracy correlates with pose estimation test error (linear fit for test: slope -0.33%, $R^2 = 0.93$, $p = 1.4e-7$; Figures 3B).

Next, we evaluated the performance of the networks on out-

of-domain horses (Figures 3A-C). Most strikingly, on out-of-domain horses, the relationship between ImageNet performance and performance on Horse-10 was even stronger. This can be quantified by comparing the linear regression slope for out-of-domain test data: -0.93% pose-estimation improvement per percentage point of ImageNet performance, $R^2 = 0.93$, $p = 9e-8$ vs. within-domain test data -0.33%, $R^2 = 0.93$, $p = 1.4e-7$ (for 50% training data). In other words, *less* powerful models overfit more on the training data. We mused that this improved generalization could be a consequence of the ImageNet pretraining or the architectures themselves.

To assess the impact of ImageNet pretraining we also trained several architectures from scratch. Thereby we could directly test if the increased slope for out-of-domain performance across networks was merely a result of more powerful network architectures. He et al. recently showed that training Mask R-CNN with ResNet backbones directly on the COCO object detection, instance segmentation and key point detection tasks, *catches-up* with the performance of ImageNet-pretrained variants if training for substantially more iterations than typical training schedules (He et al., 2018). However, due to the nature of these tasks, they could not test this relationship on out-of-domain data.

For fine-tuning from ImageNet pretrained models, we trained for $30k$ iterations (as the loss had flattened; see Figures 4, 5). First, we searched for optimal schedules for training from scratch while substantially increasing the training time ($6X$ longer). We found that cosine decay with restart was best for out-of-domain performance (see Methods; Figure 5A). Consistent with He et al., 2018, we found that randomly initialized networks could closely match the performance of pretrained networks, given enough data and time (Figure 5A,B). As expected, for smaller training sets (5% training data; 160 images), this was not the case (Figure 5A). While task-training could therefore match the performance of pretrained networks given enough training data, this was not the case for novel horses (out-of-domain data). The trained from-scratch networks never caught up and indeed plateaued early (Figure 5A). Quantitatively, we also found that for stronger networks (ResNets and EfficientNets) generalization was worse if trained from scratch (Figure 5B). Interestingly that was not the case for the lightweight models, i.e. MobileNetV2s (cf. Raghu et al., 2019).

In summary, transfer learning offers multiple advantages. Not only does pretraining networks on ImageNet allow for using smaller datasets and shorter training time, it also strongly improves robustness and generalization, especially for more powerful, over-parameterized models. In fact, we found a strong correlation between generalization and ImageNet accuracy (Figure 3).

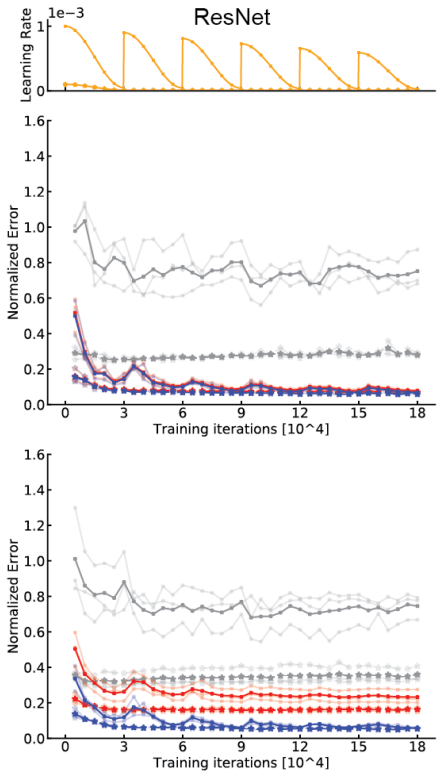

Figure 4. **Training randomly initialized networks longer cannot rescue out-of-domain performance. Top Row:** Normalized error vs. training iterations for ResNet-50 using 50% of the training data. Test errors when training from scratch (solid lines) closely match the transfer learning (dashed lines) performance after many iterations (See also Figure 5 to compare to MobileNetV2s and EfficientNets). **Bottom Row** Same as Top but using 5% of the training data; note, however, for just 5% training data, the test errors do not approach the test error of pre-trained models for larger models.

4. Discussion

We developed a novel pose estimation benchmark for out-of-domain robustness (`horse10.deeplabcut.org`). Furthermore, we report two key findings: (1) pretrained-ImageNet networks offer known advantages: shorter training times, and less data requirements, as well as a novel advantage: robustness on out-of-domain data, & (2) networks that have higher ImageNet performance lead to better generalization, if pretrained. Collectively, this sheds a new light on the inductive biases of "better ImageNet architectures" for visual tasks to be particularly beneficial for robustness.

While we found a significant advantage of using pretrained networks for out-of-domain robustness, there is still a gap to close. However, we believe that our work demonstrates that transfer learning approaches are powerful to build robust architectures. Furthermore, by sharing our animal pose robustness benchmark dataset, we also believe that the community can work towards closing the gap.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. Tensorflow: a system for large-scale machine learning. In *OSDI*, volume 16, pp. 265–283, 2016.

Anderson, T. and McIlwraith, C. Longitudinal development of equine conformation from weanling to age 3 years in the thoroughbred. *Equine veterinary journal*, 36(7): 563–570, 2004.

Andriluka, M., Pishchulin, L., Gehler, P., and Schiele, B. 2d human pose estimation: New benchmark and state of the art analysis. In *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition*, pp. 3686–3693, 2014. URL http://ieeexplore.ieee.org/document/6909866/.

Andriluka, M., Iqbal, U., Insafutdinov, E., Pishchulin, L., Milan, A., Gall, J., and Schiele, B. Posetrack: A benchmark for human pose estimation and tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5167–5176, 2018.

Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017.

Cheng, B., Xiao, B., Wang, J., Shi, H., Huang, T. S., and Zhang, L. Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5386–5395, 2020.

Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., and Darrell, T. DeCaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pp. 647–655, 2014. URL https://arxiv.org/abs/1310.1531.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016. URL https://arxiv.org/abs/1512.03385.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

He, K., Girshick, R., and Dollár, P. Rethinking imagenet pre-training. *arXiv preprint arXiv:1811.08883*, 2018.

Hendrycks, D., Lee, K., and Mazeika, M. Using pre-training can improve model robustness and uncertainty. In *ICML*, 2019.

Insafutdinov, E., Pishchulin, L., Andres, B., Andriluka, M., and Schiele, B. DeeperCut: A deeper, stronger, and faster multi-person pose estimation model. In *European Conference on Computer Vision*, pp. 34–50. Springer, 2016. URL https://arxiv.org/abs/1605.03170.

Insafutdinov, E., Andriluka, M., Pishchulin, L., Tang, S., Levinkov, E., Andres, B., and Schiele, B. Arttrack: Articulated multi-person tracking in the wild. In *CVPR'17*, 2017. URL http://arxiv.org/abs/1612.01465.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization, 2014.

Kornblith, S., Shlens, J., and Le, Q. V. Do better imagenet models transfer better? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2661–2671, 2019.

Kreiss, S., Bertoni, L., and Alahi, A. Pifpaf: Composite fields for human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11977–11986, 2019.

Kümmerer, M., Wallis, T. S., and Bethge, M. Deepgaze ii: Reading fixations from deep features trained on object recognition. *arXiv preprint arXiv:1610.01563*, 2016. URL https://arxiv.org/abs/1610.01563.

Li, H., Singh, B., Najibi, M., Wu, Z., and Davis, L. S. An analysis of pre-training on object detection. *arXiv preprint arXiv:1904.05871*, 2019.

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

Maceira-Elvira, P., Popa, T., Schmid, A.-C., and Hummel, F. C. Wearable technology in stroke rehabilitation: towards improved diagnosis and treatment of upper-limb motor impairment. *Journal of neuroengineering and rehabilitation*, 16(1):142, 2019.

Magnusson, L.-E. and Thafvellin, B. Studies on the conformation and related traits of standardbred trotters in sweden. *Journal of Animal Physiology and Animal Nutrition (Germany, FR)*, 1990.

Mathis, A., Mamidanna, P., Cury, K. M., Abe, T., Murthy, V. N., Mathis, M. W., and Bethge, M. Deeplabcut: markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience*, 21(9):1281–1289, 2018.

Mathis, A., Yüksekgönül, M., Rogers, B., Bethge, M., and Mathis, M. W. Pretraining boosts out-of-domain robustness for pose estimation. *arXiv preprint arXiv:1909.11229*, 2019.

Mathis, M. W. and Mathis, A. Deep learning tools for the measurement of animal behavior in neuroscience. *Current Opinion in Neurobiology*, 60:1–11, 2020.

Nath, T., Mathis, A., Chen, A. C., Patel, A., Bethge, M., and Mathis, M. W. Using deeplabcut for 3d markerless pose estimation across species and behaviors. *Nature Protocols*, 14:2152–2176, 2019.

Ning, G., Pei, J., and Huang, H. Lighttrack: A generic framework for online top-down human pose tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1034–1035, 2020.

Ostrek, M., Rhodin, H., Fua, P., Müller, E., and Spörri, J. Are existing monocular computer vision-based 3d motion capture approaches ready for deployment? a methodological study on the example of alpine skiing. *Sensors*, 19 (19):4323, 2019.

Raghu, M., Zhang, C., Kleinberg, J., and Bengio, S. Transfusion: Understanding transfer learning for medical imaging. In *Advances in Neural Information Processing Systems*, pp. 3342–3352, 2019.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4510–4520, 2018.

Shah, S., Sharma, A., Jain, A., et al. On the robustness of human pose estimation. *arXiv preprint arXiv:1908.06401*, 2019.

Tan, M. and Le, Q. V. Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv preprint arXiv:1905.11946*, 2019.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pp. 3320–3328, 2014.

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., and He, Q. A comprehensive survey on transfer learning. *arXiv preprint arXiv:1911.02685*, 2019.

5. Appendix

5.1. List of annotated horse bodyparts

The following 22 body parts were labeled by an expert in Thoroughbred horses [BR] across $8,114$ frames: Nose, Eye, Nearknee, Nearfrontfetlock, Nearfrontfoot, Offknee, Offfrontfetlock, Offfrontfoot, Shoulder, Midshoulder, Elbow, Girth, Wither, Nearhindhock, Nearhindfetlock, Nearhindfoot, Hip, Stifle, Offhindhock, Offhindfetlock, Offhindfoot, Ischium. We used the DeepLabCut2.0 toolbox (Nath et al., 2019) for labeling.

5.2. Pose estimation Architecture and Training Parameters

For this study we adopted a pose estimation toolbox called DeepLabCut (Mathis et al., 2018; Nath et al., 2019). The TensorFlow (Abadi et al., 2016)-based network architectures was adapted while keeping data loading, training, and evaluation consistent. The feature detectors in DeepLabCut consist of a backbone followed by deconvolutional layers to predict pose scoremaps and location refinement maps, which can then be used for predicting the pose while also proving a confidence score (Insafutdinov et al., 2016; Mathis et al., 2018). For the backbone we utilized MobileNetV2 (Sandler et al., 2018), residual networks (ResNets) (He et al., 2016) and EfficientNet backbones (Tan & Le, 2019).

We utilize an output stride of 16 for the ResNets (achieved by atrous convolution) and then upsample the filter banks with deconvolutions by a factor of two to predict the heatmaps and location-refinement at 1/8th of the original image size scale. For MobileNetV2 (Sandler et al., 2018), we configured the output-stride as 16 (by changing the (otherwise) last stride 2 convolution to stride 1). We utilized four variants of MobileNetV2 with different expansion ratios (0.35, 0.5, 0.75 and 1) as this ratio modulates the ImageNet accuracy from 60.3% to 71.8%, and pretrained models on ImageNet are available from TensorFlow (Abadi et al., 2016). The base EfficientNet model was designed by Tan & Le, 2019 through a neural architecture search to optimize for accuracy and FLOPS. From B0 to B6, compound scaling is used to increase the width, depth, and resolution of the network, which directly corresponds to an increase in ImageNet performance Tan & Le, 2019. We used the AutoAugment pre-trained checkpoints as well as adapted the EfficientNet implementation from Tensorflow[1] and configured the output-stride as 16 (by changing the (otherwise) last stride 2 convolution to stride 1).

The training loss is defined as the cross entropy loss for the scoremaps and the location refinement error via a Huber

[1]URL: https://github.com/tensorflow/tpu/tree/master/models/official/efficientnet#2-using-pretrained-efficientnet-checkpoints

loss with weight 0.05 (Mathis et al., 2018). For training, a cosine learning rate schedule, as in (Kornblith et al., 2019) with ADAM optimizer (Kingma & Ba, 2014) and batchsize 8 was used.

5.3. Cross Validation of Learning Schedules

To fairly compare the pose estimation networks with different backbones, we cross-validated the learning schedules. For models with pretraining and from scratch, we cross validated the cosine learning rate schedules by performing a grid search of potential initial learning rates and decay targets to optimize their performance on out of domain data. Given that our main result is that task-training can catch up with fine-tuning pre-trained models given sufficiently large training sets (He et al., 2018), we show that this is not the case for out-of-domain data. Thus, in order to give from scratch trained models the best shot, we optimized the performance on out of domain data.

Because of the extensive resources required to cross validate all models, we only underwent the search on MobileNetsV2 0.35 and 1.0, ResNet 50, and EfficientNets B0, B3, and B5 for the pretraining and from scratch variants. For all other models, the parameters from the most similar networks were used for training (i.e. EfficientNet-B1 used the parameters for EfficientNet-B0). The grid search started with the highest possible initial learning rate that was numerically stable for each model; lower initial learning rates were then tested to fine tune the schedule. A zero and nonzero decay target point were tested for each initial learning rate. In addition to the initial learning rates and decay targets, we experimented with shortening the cosine decay and incorporating restarts. All cross validation experiments were performed on the three splits with 50% of the data for training. The table below lists the various initial learning rates explored during cross validation for each model with pretraining. For the learning schedules we use the following abbreviations: Initial Learning Rates (ILR) and decay target (DT).

The table below list the various initial learning rates explored during cross validation for each model with pretraining:

Model	ILR			
MobileNetV2-0.35	1e-2	5e-3	1e-3	5e-4
MobileNetV2-1	1e-2	5e-3	1e-3	5e-4
ResNet-50	1e-3	5e-4	1e-4	5e-5
EfficientNet-B0	2.5e-3	1e-3	7.5e-4	5e-4
EfficientNet-B3	1e-3	5e-4	1e-4	5e-5
EfficientNet-B5	5e-4	1e-4		

For the ImageNet pretrained case, the learning rate schedule without restarts was optimal on out of domain data, and the resulting optimal parameters are as follows:

Models	ILR & DT	
MobileNetV2s 0.35, 0.5	1e-2	0
MobileNetV2s 0.75, 1.0	1e-2	1e-4
ResNets 50, 101	1e-4	1e-5
EfficientNets B0, B1	5e-4	1e-5
EfficientNets B2,B3,B4	5e-4	0
EfficientNets B5,B6	5e-4	1e-5

The initial learning rates explored for the from scratch models during cross validation are as follows:

Model	ILR			
MobileNetV2-0.35	1e-2	5e-3	1e-3	5e-4
MobileNetV2-1	1e-1	1e-2	1e-3	1e-4
ResNet-50	1e-3	5e-4	1e-4	5e-5
EfficientNet-B0	1e-3	5e-4	1e-4	5e-5
EfficientNet-B3	1e-3	5e-4	1e-4	5e-5

For models trained from scratch, we found that using restarts lead to the best performance on out of domain data. The optimal learning rates found during the search are as follows:

Models	ILR & DT	
MobileNetV2s 0.35, 0.5	5e-2	5e-3
MobileNetV2s 0.75, 1.0	1e-2	0
ResNet 50	5e-4	5e-5
EfficientNets B0, B3	1e-3	0

5.4. Task-learning vs. Transfer learning

Figure 5 below shows results for transfer learning vs. training from scratch forMobileNetV2s and EfficientNets. The results for ResNet 50 are in the main text (Figure 4).

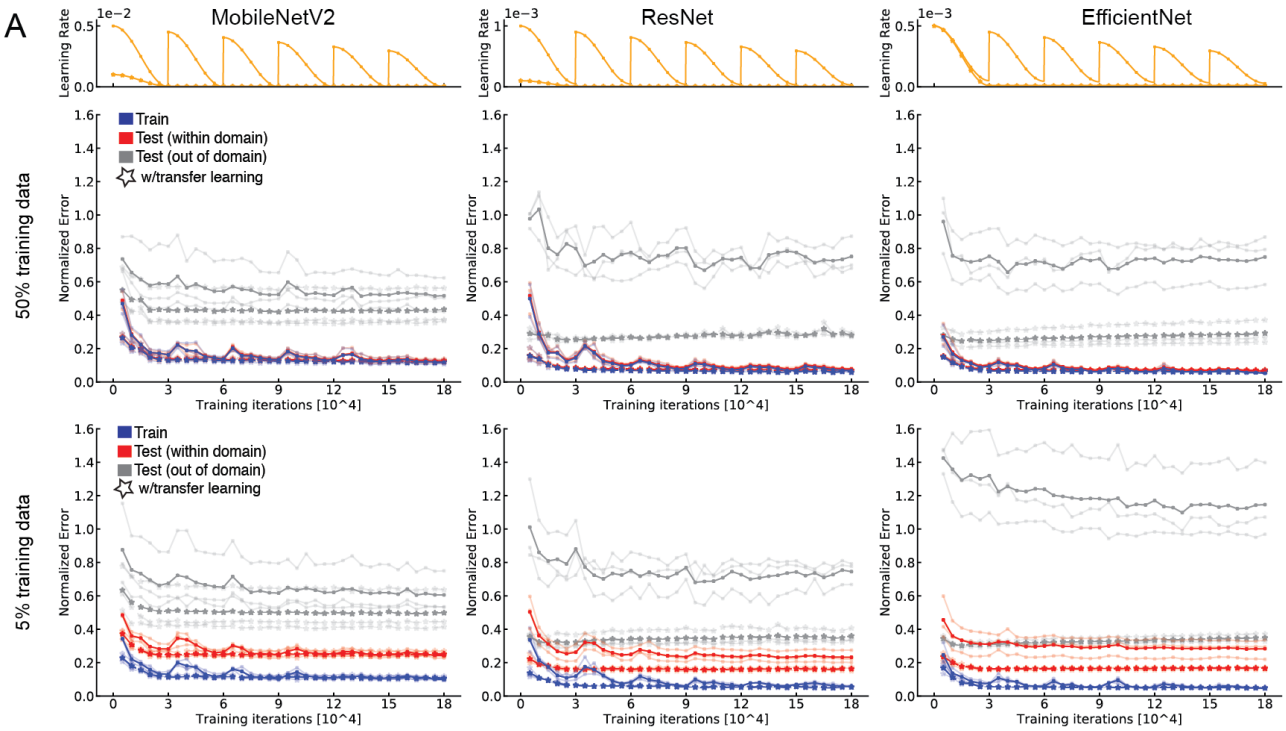

Figure 5. **Training randomly initialized networks longer cannot rescue out-of-domain performance. A: Top Row:** Normalized error vs. training iterations for MobileNetV2-0.35, ResNet-50 and EfficientNet-B0 using 50% of the training data. Test errors when training from scratch (solid lines) closely match the transfer learning (dashed lines) performance after many iterations. Crucially, out-of-domain testing does not approach performance for pretrained network (stars). **A: Bottom Row** Same as Top but using 5% of the training data; note, however, for just 5% training data, the test errors do not approach the test error of pre-trained models for larger models. **B** Normalized error vs. ImageNet Top 1% accuracy for all 14 models. From scratch training showed poor accuracy on out-of-domain horses.