Lost in Translation: Modern Image Classifiers still degrade even under simple Translations

Leander Kurscheidt [1] **Matthias Hein** [1]

Abstract

Modern image classifiers are used potentially in safety-critical applications and thus should not be vulnerable to natural transformations of the image as it can happen due to variations in the image acquisition. While it is known that image classifiers can degrade significantly in performance with respect to translations and rotations, the corresponding works did not ensure that the object of interest is fully contained in the image and also introduce boundary artefacts so that the input is not a natural image. In this paper we leverage pixelwise segmentations of the ImageNet-S dataset (Gao et al., 2021) in order to search for the translation and rotation which ensures that the object is i) fully contained in the image (potentially together with a zoom) and ii) the image is natural (no padding with black pixels) such that the resulting natural image is misclassified. We observe a consistent drop in accuracy over a large set of image classifiers showing that natural adversarial changes are an important threat model which deserves more attention.

1. Introduction

Due to the usage of neural networks in various safety-critical applications, it is of paramount importance to study their behaviour and failure modes.

2. The Task

For tasks in computer vision, we would expect models deployed on real world-tasks to have a minimum level of robustness towards natural input changes. Previous works have considered various changes, for example corruptions

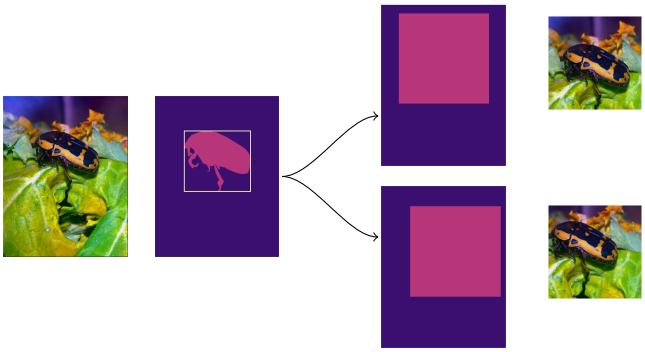


Figure 1. Illustration of the work flow for translations. We generate from the segmented images translated versions of the image always containing the bounding box of the object. The cropped regions and the resulting images are shown on the right.

(Hendrycks & Dietterich, 2018; Taori et al., 2020), changes in the background (Xiao et al., 2020) or chains of simple, restricted transformation functions like noise-injection, flow-field or color-distortion (Suzuki & Sato, 2019). Additionally, adversarial samples via differentiable rendering (Liu et al., 2019) or learning the perturbation set (Wong & Kolter, 2021) have been considered. This paper focuses on the geometric transformations of translations and rotations, which can naturally follow from a slightly different angle or position of the camera, due to e.g. movement. While existing work has already shown the brittleness of existing models to these simple geometric transformations (Athalye et al., 2018; Engstrom et al., 2019; Gokhale et al., 2021), in these studies it was not guaranteed that i) the translated/rotated image still contains the object and ii) the images were not natural as padding with black pixels or other variants has been used which leads to an image which would be considered unnatural. In contrast, we leverage the recent work of Gao et. al (2021), who published Imagenet-S dataset containing a subset of ImageNet images, with full semantic segmentations according to the ImageNet classes. Using these groundtruth segmentations, we ensure that the object corresponding to the class of the image is fully contained in the translated/rotated version of the image and all images are "natural", in the sense that there are no black pixels

[1]Department of Computer Science, University of Tübingen, Tübingen, Germany. Correspondence to: Leander Kurscheidt <Leander.Kurscheidt@gmx.de>, Matthias Hein <matthias.hein@uni-tuebingen.de>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

or other forms of padding like reflection. This pipeline is illustrated by Figure 1. Since we manage to transform the input without the use of padding, the generated images could naturally occur by a slight translation/rotation of the camera configuration and thus no potential bias is introduced and thus our benchmark yields a realistic worst-case evaluation. The worst-case translation/rotation is found by a grid-search which is adaptively refined for improved efficiency. While it has been argued that CNN architectures are not necessarily translation invariant (Kayhan & Gemert, 2020), it is nevertheless surprising that state-of-the-art models trained typically with heavy data augmentation still show significant performance degradation when such simple transformations such as translation and rotation are used. We also observe quite different degradation of SOTA ImageNet models, showing that certain architectures/training schemes lead to significantly more robust models.

To guarantee full visibility of the objects to classify, we need to track their position during the transformations. To achieve this, we use the segmentation Imagenet-S Dataset (Gao et al., 2021). We have chosen the Imagenet-S 300 version for the validation partition, which is the only partition for which the data can be accessed. This dataset in turn is based on the relabeled ImageNet from Beyer et. al (2020). We perform a rudimentary cleaning of the dataset by sorting out the data points where the classes from the segmentation-annotation do not agree with the new labels from Beyer et. al (2020). The segmentation-data and the labels can be downloaded with instructions on the respective repositories.

To prepare the data for the transformations, we first fit a bounding box around the object and enclose the bounding box with a best-fit centered square of the desired input dimension t for the model, usually $224px$, which encodes the current crop. The crop is the section of the image that the model will get as input. To allow for batch-processing, we have to bring the image into a uniform representation. We do this by padding, or cropping, the image to t pixels around crop-area of length t, so an starting with an image of arbitrary (h, w) dimensions, we arrive at $(3t, 3t)$. Summing up, we arrive at the padded image-data with dimension $(3t, 3t)$ and the metadata consisting of the bounds of the embedded image, the position of the crop and the position of the object inside the crop-area. This process is visualized in Figure 1. Not every object is embedded in an image with enough background so that it can be fully shown without leaving the image area. We discard those images and arrive at a total number of 2013 instances to classify. In the rare case that multiple instances are on the same image, we treat them as separate classification targets.

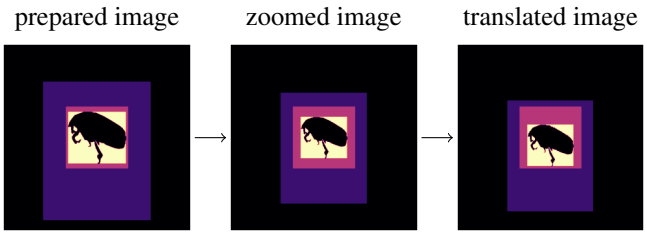


Figure 2. This figure visualizes the translation pipeline from the perspective of the meta-information.

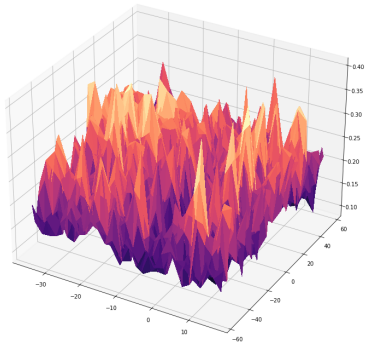


Figure 3. Predicted probability for ice cream over all translations of an image of the class ice cream for the BEIT model from Bao et. al (2021).

2.1. Translation

We now have everything we need to compute the bounds for our transformations. Starting with the translation, we first zoom out up to 20% to gain the freedom to translate without the object leaving our crop-area. We then compute the bounds both from the crop-area to the surrounding image-space, but also to the enclosed object and combine them into our maximum translation bounds. This pipeline is visualized in Figure 2.

The confidence plot in the correct class over the set of possible translations shows often drastic changes, as shown in Figure 3. Curiously, we also notice a $1px$-periodic pattern in the confidence surface for most images. In order to find the worst-case translation without doing an exhaustive search, we perform adaptive sampling. First, we find the minimum for the periodic pattern by a dense sampling of a small patch near the origin. We then perform a coarse sampling using $2px$ steps over the whole possible range of translations keeping the object fully visible in the image and then perform a denser sampling around the neighborhood of the translations leading to the smallest confidence. It takes about 90 minutes to analyze the 2013 samples on a Nvidia Tesla V100-SXM2.

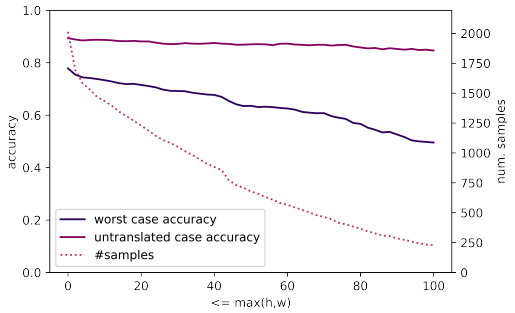


Figure 4. Accuracy on untranslated images versus worst-case accuracy over all possible translations for the SEResNeXt-model (more models in the appendix). The x-axis varies over the set of images which allow at least a minimum degree of freedom $\max(h, w)$ of the translation (the dotted line shows the number of such images). The gap increases when the degree of freedom becomes larger.

2.2. Rotation

Analyzing rotations is easier, since the parameter space is only one-dimensional and quite restricted. We first try to zoom out to nearly 29.28% to get more freedom for the rotation. Next, we compute the maximum possible rotation both with respect to the image bounds and to the object bounds. The minimum over those values defines the parameter-space so that the object is visible for all generated rotations and there is no padding with black pixels. We then sample the parameter evenly spaced 500 times and record our results. It takes about two hours to analyze the 2013 samples on a Nvidia Tesla V100-SXM2.

3. Analysis

We evaluate a range of recent state-of-the-art models and classical models to compare the performance and assess whether improvements were made. We employ the pre-trained BEIT (Bao et al., 2021) (beit_large_patch16_224), Swin Transformer (Liu et al., 2021) (swin_large_patch4_window7_224) and SEResNeXt (Hu et al., 2017) (seresnext50_32x4d) from the PyTorch Image Models (timm)-library (Wightman, 2019). We also study the pre-trained ConvNext model (Liu et al., 2022) in the large-configuration. Additionally, the pre-trained Resnet50 (He et al., 2015), Resnet18 (He et al., 2015) and VGG16 (Simonyan & Zisserman, 2015) from the torchvision model-zoo are investigated. All models expect the input to be of dimension $(224, 224)$.

3.1. Translation

When analyzing the translations, the first thing to notice is that the worst-case performance of the models depends on the possible degrees of freedom for translation. This is visualized in Figure 4. The translations are feasible, measured by $max(h, w)$ where h and w are the translation bounds for with and height in pixels, the wider the gap. Note that this changes the set of images we can work with and thus we report the original performance on these images too. Results for all models are shown in Table 1. The differences in performance are astounding and when investigating the locations of the label flips, we often find localized areas which are usually disjoint between models. A visualization of this phenomena can be found in the Appendix. When visualizing the various translated images, the similarity between correctly classified images and wrongly classified is striking. The reason for the miss classification of the models is usually not obvious, they appear very similar to humans, an example is shown in Figure 5. The drop in accuracy is surprising when considering that we only translate the visible crop inside the image bounds, without changing anything about the image itself.

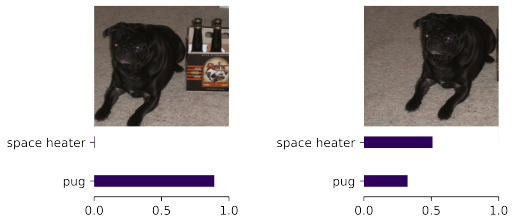


Figure 5. A correctly and an incorrectly classified sample for the BEiT-Model. The dog of class "pug" is fully visible in both images.

3.2. Rotation

The degradation in prediction performance is less severe for rotation than for translation, but one can still observe a drop in worst-case accuracy as shown in Table 2. Similar to translation, we see a dependence on the freedom of rotation available, with drastically strong degradation if we restrict ourselves to instances which allow for a minimum of 30 degrees rotation while keeping the object in the image and no padding with black pixels is necessary.

4. Conclusion

In this paper we have analyzed various existing models on ImageNet with respect to their behavior with respect to translation and rotation. In contrast to previous work we ensure that the object is always visible and there are no artefacts from potential padding of the image. Even though the full class information is available in every transformed image we see large drops in worst case accuracy. Thus even SOTA models can be fooled by this simple shifts mimicking variations in the camera position. In particular for images where one can test larger translations and rotations the performance drops are very high. On the other hand we see

Table 1. **Translation:** For all models we report top-1 ImageNet-accuracy $acc_{ImageNet}$. For the subset of 2013 images from ImageNet-S (Gao et al., 2021) we report the accuracy for the untranslated subset $acc_{untrans}$ and the worst case accuracy acc_{worst} over all translations leaving the object fully visible. Additionally, we report separately for the subset of 229 images which allow translations of at least 100 pixels ($max(h, w) \geq 100$) the accuracy on these (untranslated) images and with the worst case over all possible translations. Note the significant drop in accuracy of up to 17% for all images and up to 40% for the images which allow larger translations. The BEiT and ConvNext model are significantly more robust than the Swin Transformer even though having similar ImageNet accuracy.

		$max(h,w) \geq 0$			$max(h,w) \geq 100$		
Models	$acc_{ImageNet}$	$acc_{untrans}$	acc_{worst}	Δ	$acc_{untrans}$	acc_{worst}	Δ
BEiT	**87.4**	**89.5**	**84.8**	**4.7**	82.0	**69.3**	**12.7**
Swin Transformer	86.3	**89.4**	77.8	11.5	**84.6**	49.6	35.1
ConvNext	84.3	87.9	81.4	6.5	80.8	61.1	19.7
SEResNeXt	81.2	86.1	75.3	10.9	77.2	50.0	27.2
Resnet50	76.1	82.9	68.7	14.2	76.0	39.7	36.2
VGG16	71.5	78.5	63.7	14.8	69.7	37.3	32.5
Resnet18	69.7	78.0	61.1	17.0	71.1	30.7	40.4

also quite significant differences in the worst-case accuracy of the models even though the original ImageNet performance is very similar. It is therefore an interesting open question if this is due to the employed architecture, the data augmentation during training or the amount of training data.

References

Athalye, A., Engstrom, L., Ilyas, A., and Kwok, K. Synthesizing robust adversarial examples. In *ICML*, 2018. doi: 10.48550/ARXIV.1707.07397.

Bao, H., Dong, L., and Wei, F. Beit: BERT pre-training of image transformers. *CoRR*, abs/2106.08254, 2021. URL https://arxiv.org/abs/2106.08254.

Beyer, L., Hénaff, O. J., Kolesnikov, A., Zhai, X., and van den Oord, A. Are we done with imagenet? *CoRR*, abs/2006.07159, 2020. URL https://arxiv.org/abs/2006.07159.

Engstrom, L., Tran, B., Tsipras, D., Schmidt, L., and Madry, A. Exploring the landscape of spatial robustness, 2019.

Gao, S., Li, Z.-Y., Yang, M.-H., Cheng, M.-M., Han, J., and Torr, P. Large-scale unsupervised semantic segmentation, 2021. URL https://arxiv.org/abs/2106.03149. arXiv:2106.0314.

Gokhale, T., Anirudh, R., Kailkhura, B., Thiagarajan, J. J., Baral, C., and Yang, Y. Attribute-guided adversarial training for robustness to natural perturbations. In *AAAI*, 2021. doi: 10.48550/ARXIV.2012.01806. URL https://arxiv.org/abs/2012.01806.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. URL http://arxiv.org/abs/1512.03385.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2018.

Hu, J., Shen, L., and Sun, G. Squeeze-and-excitation networks. *CoRR*, abs/1709.01507, 2017. URL http://arxiv.org/abs/1709.01507.

Kayhan, O. S. and Gemert, J. C. v. On translation invariance in cnns: Convolutional layers can exploit absolute spatial location. In *CVPR*, pp. 14274–14285, 2020.

Liu, H.-T. D., Tao, M., Li, C.-L., Nowrouzezahrai, D., and Jacobson, A. Beyond pixel norm-balls: Parametric adversaries using an analytically differentiable renderer. In *ICLR*, 2019. doi: 10.48550/ARXIV.1808.02651.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. *CoRR*, abs/2103.14030, 2021. URL https://arxiv.org/abs/2103.14030.

Liu, Z., Mao, H., Wu, C., Feichtenhofer, C., Darrell, T., and Xie, S. A convnet for the 2020s. *CoRR*, abs/2201.03545, 2022. URL https://arxiv.org/abs/2201.03545.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1409.1556.

Suzuki, T. and Sato, I. Adversarial transformations for semi-supervised learning. *CoRR*, abs/1911.06181, 2019. URL http://arxiv.org/abs/1911.06181.

Table 2. **Rotation:** For all models we report top-1 ImageNet-accuracy $acc_{ImageNet}$. For the subset of 2013 images from ImageNet-S (Gao et al., 2021) we report the accuracy for the unrotated subset $acc_{unrotated}$ and the worst case accuracy acc_{worst} over all rotations leaving the object fully visible. Additionally, we report separately for the subset of 109 images which allow rotations of at least 30 degrees the accuracy on these (unrotated) images and the worst case over all possible rotations. Note that for the worst case over images allowing rotations of at least 30 degrees we see performance drops of up to 35%. Again we observe that the BEiT and ConvNext model are significantly more robust than the Swin Transformer even though having similar ImageNet accuracy.

		$min(deg) \geq 0$			$min(deg) \geq 30$		
Models	$acc_{ImageNet}$	$acc_{unrotated}$	acc_{worst}	Δ	$acc_{unrotated}$	acc_{worst}	Δ
BEiT	**87.4**	**90.5**	**89.5**	**1.0**	87.6	**80.0**	**7.6**
Swin Transformer	86.3	90.6	87.8	2.8	88.6	61.0	27.6
ConvNext	84.3	89.4	82.8	6.6	84.6	69.2	15.4
SEResNeXt	81.2	87.0	84.8	2.2	83.8	61.9	21.9
Resnet50	76.1	85.4	82.0	3.4	81.0	45.7	35.2
VGG16	71.5	80.7	77.1	3.6	75.2	43.8	31.4
Resnet18	69.7	79.9	76.3	3.6	74.3	40.0	34.3

Taori, R., Dave, A., Shankar, V., Carlini, N., Recht, B., and Schmidt, L. Measuring robustness to natural distribution shifts in image classification. *NeurIPS*, 33:18583–18599, 2020.

Wightman, R. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019.

Wong, E. and Kolter, J. Z. Learning perturbation sets for robust machine learning. In *ICLR*, 2021. doi: 10.48550/ARXIV.2007.08450.

Xiao, K. Y., Engstrom, L., Ilyas, A., and Madry, A. Noise or signal: The role of image backgrounds in object recognition. In *ICLR*, 2020.

A. The Accuracy Gap - Translation

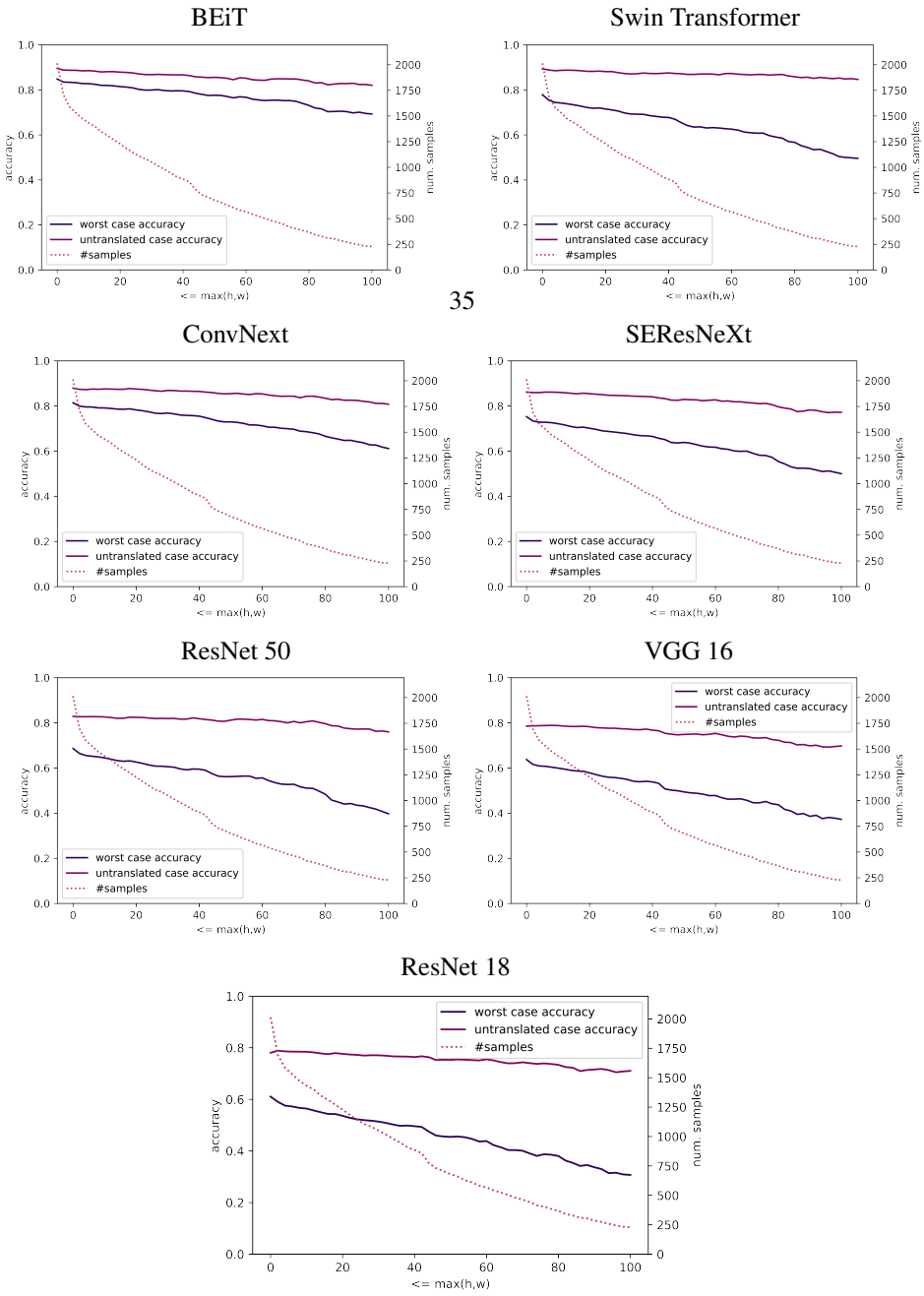
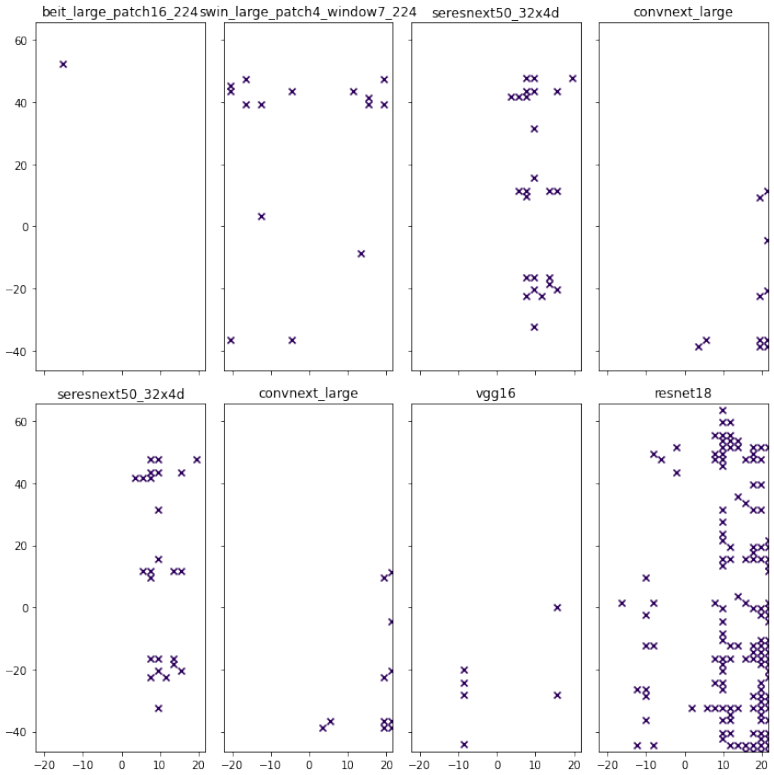


Figure 6. Accuracy gap vs freedom of translation for various models

B. Position of the Label Flips



The corresponding sample of the class *doberman*

Figure 7. The position of the label flips in the parameter-space of the translation by model.