Finding Differences Between Transformers and ConvNets Using Counterfactual Simulation Testing

Anonymous Author(s)
Affiliation
Address
email

Figure 1: A sample of objects in our proposed **Naturalistic Variation Object Dataset (NVD)** in canonical pose with different lighting. NVD includes 272k images of object pose, scale, viewpoint and occlusion naturalistic variations of 92 object models with 27 HDRI skybox lighting environments.

Abstract

Contemporary deep neural networks tend to be evaluated on static test sets. One shortcoming of this is the fact that these deep neural networks cannot be easily evaluated for robustness issues with respect to specific scene variations. For example, it is hard to study the robustness of these networks to variations of object scale, object pose, scene lighting and 3D occlusions. The main reason is that collecting real datasets with fine-grained naturalistic variations can be extremely time-consuming and expensive. In this work, we present *Counterfactual Simulation Testing*, a counterfactual framework that allows us to study the robustness of neural networks with respect to some of these naturalistic variations by building realistic synthetic scenes that allow us to ask *counterfactual questions* to the models, ultimately providing answers to questions such as *"Would your classification still be correct if the object were viewed from the top?"* or *"Would your classification still be correct if the object would be partially occluded by another object?"*. Our method allows for a fair comparison of the robustness of recently released, state-of-the-art Convolutional Neural Networks and Vision Transformers, with respect to these naturalistic variations. Among other interesting differences between these classes of architectures, we find evidence for differences in performance with respect to object viewpoint, object scale and occlusions.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

1 Introduction

Testing computer vision models is a challenging endeavour. In order to claim the superiority of a model, the computer vision community usually studies validation accuracy on the ImageNet [12] dataset. Comparing models uniquely using Top-1 and Top-5 accuracy on this dataset can result in an incomplete evaluation of the advantages and disadvantages of each model. Recently, Vision Transformers (ViTs) [15] have been proposed as an alternative deep neural network model to rival Convolutional Neural Networks (ConvNets) [36] for computer vision tasks.

ViTs have achieved impressive results, rivaling ConvNets and sometimes outperforming them in ImageNet accuracy. As it stands, if we restrict training data to ImageNet-22k, BEiT-L [5], a ViT variant, stands at the top of the ImageNet-1k leaderboard, with ConvNext-XL [38], a ConvNet, as a close second. Although the competition between these two classes of architectures has not yet been decided, there have been high-profile studies of potential advantages of ViTs compared to ConvNets. ViTs are believed to be more robust to certain adversarial attacks compared to ConvNets [51, 2, 8]. Although, recent work has also shown that this is not necessarily the case when ConvNets adopt the training recipes of ViTs [4]. Also, ViTs are believed to be more robust to domain generalization than CNNs [42, 4] and against occlusion, as described in Naseer et al. [41]. This last work shows that ViTs are more resistant than ConvNets to different types of patch removal from images, which is a way to simulate occlusions. One limitation of this study is that patch removal is a convenient but not fully realistic proxy for occlusion. It is natural to use such a proxy since datasets to study the effects of occlusion on object recognition and detection are scarce.

A more general limitation to works that compare properties of ViTs and ConvNets is that, even though they try to compare models of similar sizes and ImageNet accuracies, they do not account for the fact that *the compared ConvNets use slightly out-of-date design and training recipes*. In their impressive work, Liu et al. [38] propose a new class of ConvNets with modernized architecture design that seeks to closely resemble the design of ViTs, yet only uses convolutional layers. This work allows for a closer inspection of whether Transformers are superior to ConvNets due to the difference in inductive biases between transformer and convolutional layers.

We believe that studying the differences that arise in learned representations between Transformers and ConvNets to natural variations such as lighting, occlusions, object scale, object pose and others is important. A priori, convolutional and transformer layers have different inductive biases that should manifest themselves in different performance characteristics. For example, there have been conjectures that ViTs should outperform ConvNets with respect to partial occlusion since the transformer layers allow for early capture of long-range dependencies.

Until now, there have been two main obstacles that prevent the careful study of these differences: (1) Transformer and ConvNet architectures were not comparable in terms of overall design techniques and training recipe details, entangling these differences with transformer vs. convolutional layer differences (2) there is a scarcity of datasets that include fine-grained naturalistic variations of object scale, object pose, scene lighting and 3D occlusions, among others.

In our work we propose an initial attempt to bridge these two obstacles, and strive to tackle the fundamental question:

Between Transformers and ConvNets; which of these models is more robust to naturalistic scene variations such as object pose, object scale, camera viewpoint, lighting and occlusions?

In order to overcome (1) we compare the ConvNext [38] convolutional architecture to the Swin [37] Transformer architecture. Naturally, our contributions lie in our answers to obstacle (2). For this, we propose a method to test a computer vision architecture in a counterfactual manner using simulated images. We call this method Counterfactual Simulation Testing. Specifically, our method allows us to ask *counterfactual questions* to the models, ultimately providing answers to questions such as *"Would your classification still be correct if the object were viewed from the top?"* or *"Would your classification still be correct if the object were partially occluded by another object?"*. This allows us to abstract from the base rate of domain gap between synthetic and real images for each architecture and to compare them fairly.

Using Counterfactual Simulation Testing we find evidence of performance differences between comparable ConvNets and Transformers with respect to object viewpoint, camera viewpoint, object scale and occlusions. We observe that consistently across different network sizes ConvNext is on

average more robust than Swin with respect to *object pose* and *camera rotations*. We also observe that ConvNext architecture usually outperform Swin architectures in terms of recognizing small scale objects. Finally, we find that Swin and ConvNext architectures are roughly equivalent in terms of robustness with respect to occlusion, with Swin pulling ahead of ConvNext for severe occlusion. In order to find these differences we generate five different realistic synthetic test sets of objects using the MIT ThreeDWorld (TDW) [18] platform with these specific naturalistic scene variations. The full dataset, named Naturalistic Variation Object Dataset (NVD), contains 272k images of 92 object models with 27 HDRI skybox lighting environments in an indoor scene.

In summary, our contributions include:

Counterfactual Simulation Testing, a method to test computer vision models for robustness with respect to naturalistic scene variations in a counterfactual manner using simulated images by varying scene parameters one-at-a-time and evaluating the stability of predictions.

Naturalistic Variation Object Dataset (NVD), a dataset containing 272k images of 92 object models with 27 HDRI skybox lighting environments in a kitchen scene with 5 subsets of naturalistic scene variations: object pose, object scale, 360° panoramic camera rotation, top-to-frontal object view and occlusion with different objects. We hope that this dataset will allow for evaluation of modern computer vision models with respect to generalization to the synthetic domain and robustness to the naturalistic variations contained therein. A sample of NVD is shown in Fig. 1.

2 Counterfactual Simulation Testing

Testing robustness to natural variations in data. One major obstacle in evaluating models is that often, an aggregate metric, such as top-1 accuracy, will be used for comparison purposes. These metrics can give a certain sense of the power of the model, but can hide intricacies in performance with respect to natural data variations. For example, it is not possible to know to what degree a model is robust to occlusion given top-1 an top-5 ImageNet accuracy metrics.

We tackle this problem by generating large amounts of realistic synthetic data, due to the dearth of real datasets exploring these variations. An object recognition network can be written as $y = f(x)$, where y is the predicted label from the image x by the network f. The network f can be either a convolutional neural network of a vision transformer. Our scene generator can be expressed as follows: $x = g(\theta^i, \psi^i, \kappa_0^i)$, where g is the simulator, θ^i is the variable scene parameter of interest (e.g. object pose, scale, etc.), ψ^i are the constant parameters controlling the main object model type, occluder object model type and lighting environment and κ_0^i are the constant scene parameters (kitchen objects in scene, camera focal length, field of view, etc.). The variable i denotes the different trials where the selection of variable scene parameter θ^i changes (thus κ_0^i also changes).

In terms of scene content, in our work, only the main object (and occluder objects) vary in the scene. The rest of the scene is a pre-designed indoor scene. For same θ_i we generate different scenes with different object models (determined by ψ^i) and different lighting environments (also in ψ^i) for diversity purposes. For different scene variations encoded in θ_i we select object occlusions, object scale, object rotation around its vertical axis, camera elevation and camera panoramic rotation around the main object (i.e. $i = 5$). We describe the details of the dataset in Section 3.

Once we have generated such a dataset in this manner, we would like to test our network for robustness with respect to all selected variations θ_i. One way is to test the network on all generated images and produce an expectation metric (averaged over ψ^i) with respect to the variable scene parameter θ_i:

$$M_f(\theta^i) = \mathbb{E}_{\psi^i}[m[f(x_{\theta^i}), \hat{y}]], \tag{1}$$

where $x_{\theta^i} = g(\theta^i, \psi^i, \kappa_0^i)$, $\hat{y}$ is the true label, and m can be any metric. Naturally, we could use top-1 or top-5 accuracy as this metric and we could plot M with respect to θ^i. This could allow us to compare different networks to find which network is more robust. Unfortunately, this solution *gives rise* to another important problem: domain generalization. By comparing two networks f and h, we seek to understand their differences when applied to real data. Yet both models might have different generalization performance to the synthetic domain, having been trained on real data. This render a comparison of M_f and M_h unfeasible.

How to address the Real to Synthetic domain gap. Comparing pre-trained models out-of-the-box on a synthetic domain, even if highly realistic like ours, will elicit differing performances given the

capabilities of models to generalize to this out-of-distribution domain. In order to avoid the brunt of this issue, we propose a way of asking counterfactual questions such as *"Is your answer still correct when I rotate the object by 60 degrees?"* or *"Is your answer still correct when I shrink the object to half of its size?"* to a model. As opposed to naively computing expected metrics, this abstracts from all prediction failures due to the model's inability to classify an object in the simulated domain under ideal circumstances due to the domain gap. This allows us to ask the deeper question *"On average, is ConvNext-Small better at recognizing small objects compared to Swin-Small?"*.

Let $\tilde{\theta}^i$ be the reference condition with respect to which we will ask all counterfactual questions. In most cases, this condition should be selected to be the average ideal condition for object recognition. For example, with respect to pose, this should be the canonical pose that elicits highest model accuracy. We compute the *proportion of correct conserved predictions* (PCCP) metric with respect to this reference condition as follows:

$$C_f(\theta^i) = \mathbb{E}_{\psi^i}[\frac{\sum_{\theta^i \in S^i} \mathbb{1}(f(x_{\tilde{\theta}^i}) = f(x_{\theta^i}))}{n(S^i)}] \tag{2}$$

where S^i is the set of all θ^i in which $f(x_{\tilde{\theta}^i}) = y$, and $n(S^i)$ is the cardinality of S^i. We can then study this metric under variable θ^i and compare different models using point estimates, integrals over intervals (or spaces) of θ^i, as well as plots of the metric. Under our counterfactual framework, computing $C_f(\theta^i)$ is effectively asking the question *"What proportion of your answers would still be correct if $\tilde{\theta}^i \rightarrow \theta^i$ happened?"*. In this way, PCCP metrics of different networks are comparable quantities of the relative robustness of the network's predictions with respect to the reference condition. In our problem we select reference conditions that present the easiest conditions for object recognition in order to have a large amount of initial correct predictions.

In summary, this approach, along with our selection of object models that are easy to classify, and high performance of our tested networks on these objects in the synthetic domain under canonical pose and ideal lighting, allows us to abstract from the synthetic domain gap. We can also consider different metrics such as *stability of predictions*, or even *proportion of incorrect predictions that become correct*. We discuss these in the supplementary material, we limit ourselves to PCCP in our main work since it suffices to study differences between ConvNets and Transformers.

3 Naturalistic Variation Object Dataset (NVD)

NVD is composed of five different subsets which seek to study naturalistic variations of scenes for object classification. Specifically, object pose, object scale, 360° panoramic camera rotation, top-to-frontal object view and occlusion with different objects. NVD is built using our customizable scene generator built on top of the MIT ThreeDWorld (TDW) [18] platform.

Main objects. We first study the intersection of the ImageNet-1k label space with the available object models in the TDW model library. After selecting all context appropriate object types we end up with 18 model classes. From these classes, we filter the 'iPod' and 'paintbrush' classes due to very low generalization accuracy across all architectures. The final classes are: *'banana'*, *'baseball'*, *'cowboy hat, ten-gallon hat'*, *'cup'*, *'dumbbell'*, *'frying pan, frypan, skillet'*, *'hammer'*, *'ice cream, icecream'*, *'laptop, laptop computer'*, *'microwave, microwave oven'*, *'mouse, computer mouse'*, *'orange'*, *'pillow'*, *'plate'*, *'screwdriver'*, *'spatula'* and *'vase'*. We found 99 object models in the TDW model library that are contained in this set of classes. We filter out 7 object models due to low quality or inconsistent size with other objects. In total we use 92 models to generate NVD. The TDW model library provides high-quality objects that are useful for studying object recognition due to the level of realism that is hard to find in open-source projects. A subset of the models can be seen in Fig. 1. Due to the relatively limited amount of realistic indoor object models in this library, some classes have more diversity in terms of different object models than others. We provide details on the amount of objects per class in the supplementary material.

HDRI skybox lighting environments. In order to increase the amount of variability we light the scene using 27 different HDRI skybox lighting environments that span the range of dark to oversaturated and contain unique features such as overhead artificial lighting, shadows, colored sunlight/moonlight, among others. We use the InteriorSceneLighting addon for TDW for this, and

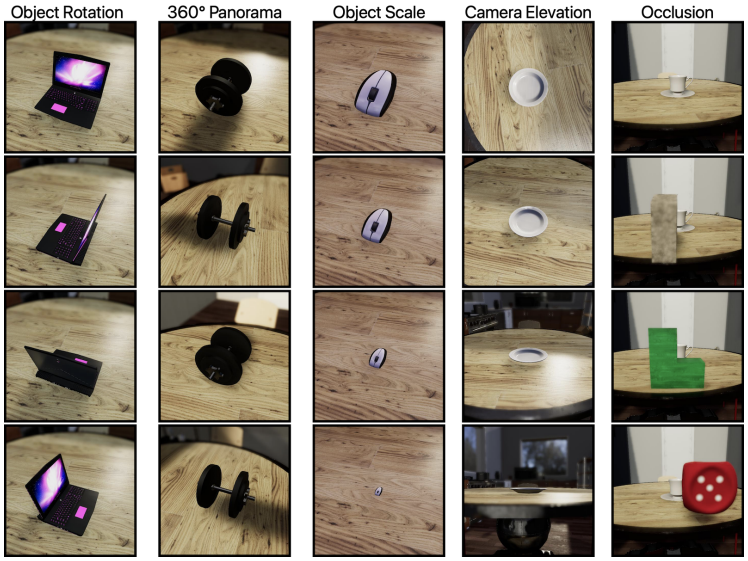

Figure 2: NVD includes the following naturalistic variations: *object rotation*, *360° panoramic camera rotation*, *object scale*, *top-to-frontal object view* and *occlusion* with different objects. All variations are performed on 92 object models under 27 different lighting environments.

the 27 pre-defined available HDRI skyboxes from TDW. Some examples of different lighting can be observed in Fig. 1 and examples of all of them are included in the supplementary material.

Naturalistic variations. We generate 5 subsets of NVD with variations in object pose, panoramic camera rotation, object scale, top-to-frontal object view and occlusion with different objects. For object pose, we perform a full rotation of the main object around its vertical axis with step size of 15°. For 360° panoramic camera rotation we perform a full camera rotation around the main object with step size of 30°, for object scale we scale the main object by $s = u(o) * 0.25$, where $u(o)$ is the unit scale of object o computed using the TDW *get_unit_scale* function. We multiply by 0.25 in order for the objects to be scaled in line with the context objects in the kitchen scene. Then we vary this scale multiplier by multipying it by a scale factor in the $[0.2, 1]$ interval, with step size 0.05. For top-to-frontal object view, we vary the camera elevation in the $[0°, 90°]$ range with step size of 5. For occlusion, we position an occluder object between the camera and the main object, and vary its x-axis position, such that it starts at the left of the frame and ends at the right, passing directly between the camera center and the main object. Thus, the main object is occluded by the occluder in a variable manner. We use three different occluder objects that have very distinct visual characteristics and occlude the object in different manners: a stone bookend, a red die and a green L-shaped block. These objects do not have a corresponding label in the ImageNet-1k dataset, and thus do not act as first-order distractors with respect to the main object, although they may have second-order associations with classes in ImageNet and thus may still act as distractors.

4 Experiments

For all experiments, we compare ConvNext architectures with their "twin" Swin architectures. ConvNext and Swin network both have five overlapping architectures that have extremely similar number of parameters and ImageNet validation accuracy. For a more detailed inspection of these statistics, see Table 1. We use the official open-sourced code for both models, as well as the public model weights. All PCCP metrics in this section are computed using top-5 predictions for greater stability of results. We use two GeForce RTX 2080 GPU to perform all experiments.

ConvNext networks generalize better to the NVD synthetic domain than Swin Transformers. We compare the mean performance of twin ConvNext and Swin architectures in Table 1. Firstly, we observe that all twin architectures have almost identical model sizes and FLOPs. They also have very comparable ImageNet accuracies, with some architecture pairs only differing by tenths of a percentage point. We observe that all ConvNext models obtain higher mean accuracy on NVD than

Table 1: Architectures used in our paper, with model details and corresponding mean accuracies on the ImageNet-1 validation set (IN) and the entirety of NVD.

Architecture	param.	FLOPs	IN top-1	NVD top-5
ConvNext-T	28M	4.5G	82.1	46.7
Swin-T	28M	4.5G	81.2	41.1
ConvNext-S	50M	8.7G	83.1	50.0
Swin-S	50M	8.7G	83.2	42.9
ConvNext-B	89M	15.4G	83.8	50.2
Swin-B	88M	15.4G	83.5	45.2
ConvNext-B-22k	89M	15.4G	85.8	58.9
Swin-B-22k	88M	15.4G	85.2	56.6
ConvNext-L-22k	198M	34.4G	86.6	67.6
Swin-L-22k	197M	34.5G	86.3	61.4

their twin Swin models. In particular, even when the corresponding ConvNext network performs worse than the twin Swin network on ImageNet (e.g. ConvNext-S/Swin-S) the ConvNext networks performs better on NVD.

This is a surprising finding that goes against the belief that Transformers are more robust to OOD generalization with findings of greater OOD generalization detailed in Bai et al. [4] and Paul et al. [42]. The improvement in OOD generalization in ConvNext networks is most likely a consequence of the modernized design of the architecture, since Bai et al. [4] align training recipes in their work.

4.1 Counterfactual Simulation Testing

Swin Transformers are more robust to occlusions than ConvNext networks. Several works have claimed superiority of Transformers over ConvNets in the case of occlusions. Naseer et al. [41] compare ResNets to DeiTs by dropping random test image patches. They show that DeiTs are much more robust than ResNets to this transformation. Although DeiTs are superior to ResNets in this scenario, we can question whether (1) this is a general phenomenon brought about by the differences between conv layers and transformer layers and (2) this type of patch dropping actually approximates naturalistic occlusions due to other objects in a scene.

For (1), we provide evidence that this difference between ResNets and DeiTs is not due to the nature of transformer vs. conv layers. In order to do so, we re-run the random patch drop experiments in [41] on all pairs of ConvNext and Swin networks. We present results for different levels of information loss in Fig. 3. For all networks we observe a drop in performance as information loss increases. We show that ConvNext networks *do not* suffer the same critical failure mode that DenseNet121, ResNet50, SqueezeNet and VGG19 exhibit in Naseer et al. [41]. The top accuracy of these four classic ConvNets collapses to half after about 10% of patches are occluded and to 0 after about 40% information loss. ConvNext shows that it is much more resistant, conserving a large part of its accuracy even after 50% information loss. Next, we observe that ConvNext networks are slightly less robust than Swin Transformers for low amounts of information loss (under 50%). Thereafter, Swin becomes comparatively much more robust as occlusion becomes severe.

For (2), we conduct counterfactual simulation testing of ConvNext and Swin networks with variable occlusion. We generate images of the main object on the center table of the scene, and add an occluder object between the camera and the main object. The details of this subset are explained in Section 3. In this counterfactual experiment we compare all occluded scenes with the non-occluded initial scene where the occluder is not in the scene. We observe in Fig. 4 that both Swin and ConvNext tend to have similar failure responses to the variable occlusions and that performance drops precipitously, achieving a minimum at zero in the x-axis. It is important to note that for maximal occlusions (i.e. occluder object at zero in x-axis) all Swin networks exhibit stronger robustness than ConvNext networks. For many of the lesser occlusions, Swin networks are still slightly ahead of ConvNext networks. We conclude that Swin Transformers are slightly more robust to occlusions than ConvNext networks and this echoes the findings of our experiment on real ImageNet images with patch drop.

ConvNext networks are more robust to object scale than Swin Transformers. In order to compare the robustness of ConvNext and Swin networks to scale, we first generate images of all main object models with unit object scale. Although objects have different volume in this state (e.g. microwaves are larger than spatulas), they take a fair amount of space in the frame and don't present much of a challenge to large architectures. Specifically, *top-5 accuracies* for Swin-L and ConvNext-L

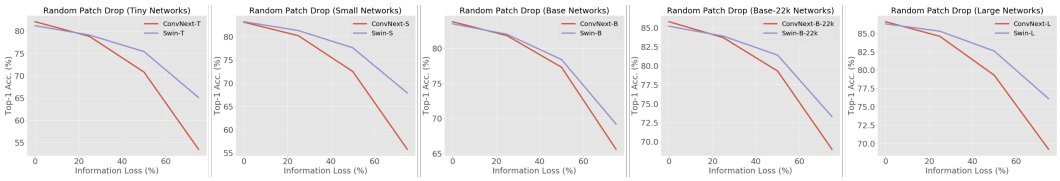

Figure 3: Random patch drop occlusion study on ConvNext and Swin networks on the ImageNet-1k validation set. Swin Transformers are slightly more robust to this type of artificial occlusion than ConvNext networks when the information loss is small, although they become comparatively much stronger as the information loss is increased.

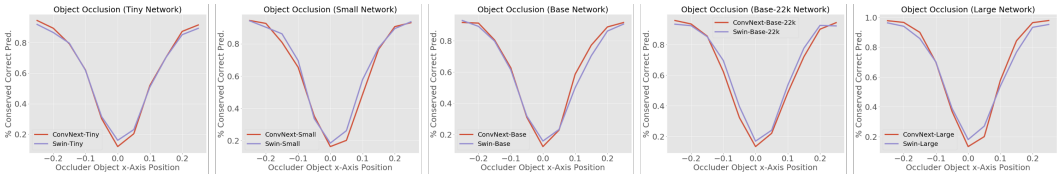

Figure 4: Counterfactual study of all sizes of ConvNext and Swin networks for occlusion of main object occluder objects.

are **90.5%** and **93.8%** respectively. We then perform counterfactual simulation testing by plotting the stability of correct predictions when the object scale multiplier is reduced from 1. We show these results in Fig. 5. We observe that for Small, Base-22k and Large-22k networks, ConvNext vastly outperforms Swin for smaller objects. For Base-22k and Large, the difference becomes large after the scale multiplier is under 0.6. For Tiny and Base networks, ConvNext and Swin perform in very similar manner, with Swin slightly outperforming ConvNext on very small objects by a small margin.

ConvNext networks are more robust to rare top and frontal object views than Swin networks. Here we compare the robustness of ConvNext and Swin networks to different viewpoints of the main object. Specifically, we start with a camera viewpoint that captures the main object from a fully-frontal view to a fully-top view. Object recognition is known to fail when the object is viewed from rare poses such as from the top [1]. In this experiment we compare all views to the canonical view with a camera pointing at the object with a 45° elevation.

We show in Figure 6, that on ImageNet-1k and ImageNet-22k-trained ConvNext and Swin networks, this is still the case. Both network classes exhibit drastic drops in proportion of conserved correct predictions when the pose is modified from a canonical pose with a camera at 45° of elevation to either a fully-frontal pose at 0° or a fully-top view at 90°. We observe that both increasing network size and pre-training on ImageNet-22k increase robustness on both architectures, with the best model being ConvNext-L-22k which achieves a conservation of correct answers in top view of around 73%.

We observe as well that all ConvNext networks obtain a higher proportion of conserved correct predictions for views from the top (e.g. with high elevation) than comparable Swin networks. We conclude that ConvNext networks are more robust to rare top views of objects than Swin Transformers given comparable architecture sizes and identical training data and recipes. Finally, we find that most ConvNext networks have higher proportion of conserved correct predictions across different elevation angles, with some notable exceptions such as a plummeting robustness of ConvNext-S in low elevation, and slightly lower robustness of ConvNext-B and ConvNext-B-22k compared to their twin Swin networks in certain elevation intervals.

ConvNext networks are, on average, more robust than Swin networks to different lateral object viewpoints. In order to test the robustness of both classes of networks to different lateral views of objects, we conduct two counterfactual studies. The first scene variation consists in a panoramic 360° camera rotation around the object. In this scenario the lighting on the object can change drastically. Many lighting skyboxes in NVD contain a light source coming through a window behind the object, and when the camera is rotated we can either have a high contrast view of the object with the main light source behind the camera, or a shaded view of the object against the light. The second variation consists of a fixed camera setting with the main object performing a full 360° rotation around its vertical axis. This allows us to keep lighting fixed while we evaluate the networks on different poses

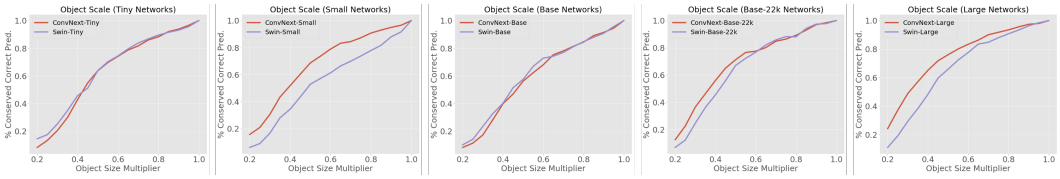

Figure 5: Counterfactual study of all sizes of ConvNext and Swin networks for object scale.

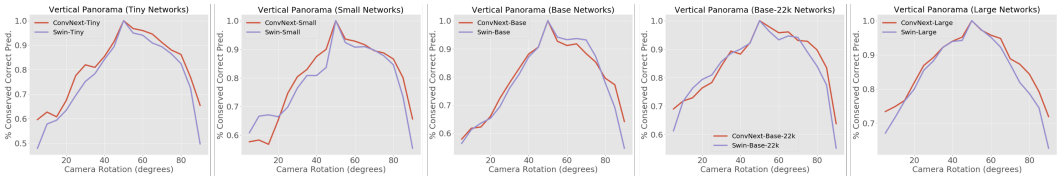

Figure 6: Counterfactual study of all sizes of ConvNext and Swin networks for frontal-to-top camera rotation around the main object.

for the same object. Using both experiments we can disentangle decreases in performance due to object pose or the interplay between object pose and lighting.

We plot the proportion of conserved correct predictions for the panoramic camera rotation in Fig. 7. First, we observe that both networks struggle to a higher extent with images taken on opposite to the position, with the light source behind the camera. This is due to two factors: the rear-view object are, in general, rare cases (e.g. the back of a microwave) and some lighting environments provide very strong light resulting in overexposed scenes with less object detail. Interestingly, we observe that for all networks ConvNext is more robust than Swin, with the slight exception of the 150° to 180° range for Base-22k networks. We also see in this experiment that curves are much less smooth and irregular. This is partly due to the nature of object rotations: some objects are easy to recognize in multiple different views, while others present very challenging views within small rotation intervals.

This experiment entangles lighting and object pose. In order to partially disentangle them, we perform a study where the camera is fixed and the main object is rotated along its vertical axis. We can observe in Fig. 8, that, on average, ConvNext is more robust to these rotations than Swin. Even so, the difference between both networks is lower than in the previous experiment, suggesting that ConvNext is more robust to different lighting and that this contributes to performance differences in Fig. 7.

5 Related Work and Discussion

Simulation In recent years, the community has explored using synthetic data for training deep networks [17, 48, 47, 16, 44, 6]. Some work goes a step further and learns the distribution of data needed to improve model performance [49, 39, 19, 7, 30, 3]. There is an easy way of measuring success of these models by evaluating them on real data. There are also attempts to benchmark and test models using synthetic data [45, 40, 28, 6, 31, 50]. The synthetic-real domain gap complicates this endeavour, making it hard to generalize insights to the real-world. In contrast, we propose a counterfactual study of model failure using a photorealistic simulator. This allows us to abstract from the model domain gap base rate. A different direction of work that could allow for this is the domain adaptation literature which adapts pixels or features to bridge the gap [20, 9, 56, 55, 26, 43].

ConvNet and Transformer comparative studies ConvNets have dominated the field of computer vision for the last decade [36, 34, 52, 23, 27, 53, 38]. Vision Transformers have recently proven to be extremely competitive, sometimes outperforming ConvNets in computer vision tasks [15, 54, 57, 58, 37, 5, 14]. There have recently been studies into the differences between ConvNets and Transformers. Among other studies [41, 59], Raghu et al. [46] study the internal representation of ViTs, finding early aggregation of global information and strong propagation of features from lower to higher layers. Bai et al. [4] show that CNNs can be as adversarially robust as ViTs, but lag behind in OOD generalization. In contrast to this work, we compare recent CNNs and ViTs with respect to naturalistic scene variations such as occlusion and object pose in a fine-grained manner, with some surprising findings. To the best of our knowledge, ours is the first work that seeks to compare these

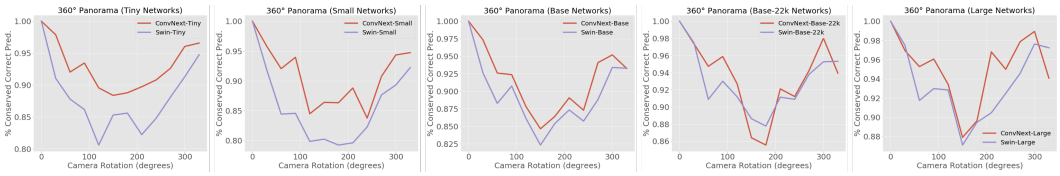

Figure 7: Counterfactual study of all sizes of ConvNext and Swin networks for panoramic 360° camera rotation.

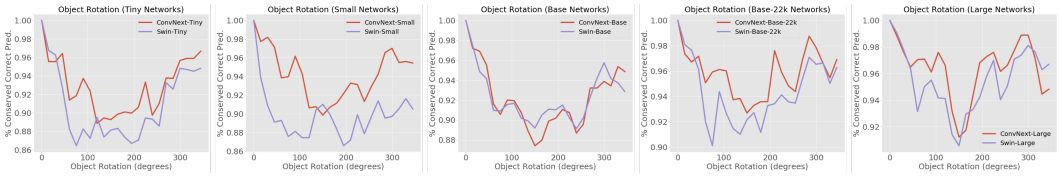

Figure 8: Counterfactual study of all sizes of ConvNext and Swin networks for 360° object rotation.

classes of architectures in this way and that proposes a way to equalize the (synthetic domain) playing field using counterfactual analysis.

Robustness and counterfactual fairness There is research on OOD robustness of networks including synthetic images [44], stylized images [21], corrupted images [24] and natural adversarial images [25]. Our effort complements this direction with a study of robustness with respect to natural scene variations. There exist investigations into network fragility regarding rare object poses [1] and occlusion [60, 33] - and work that tries to address the weaknesses [32]. Our work represents a serious attempt at providing a flexible and general method with which to study these types of variations, along with many others that are less common in the literature such as realistic lighting environment changes. Our counterfactual framework allows us to compare the robustness of different architectures that might not have the same base-level performance in a fair setting. There exists work on counterfactual fairness [35, 10, 29], a field that studies ML decision fairness from a counterfactual viewpoint. The closest literature to our work is the sparse literature of counterfactual interpretability [22] and sensitivity [11, 13, 50]. To the best of our knowledge, we are the first to propose a method to study object recognition networks for robustness with respect to natural scene variations by combining two key ingredients: a counterfactual approach and realistic synthetic data with fine-grained variations.

Limitations Due to limited real data, we test networks on simulated data. It can be hard to generalize insight into the real world. We tackle this limitation by (1) generating a photorealistic dataset using state-of-the-art research software, with networks achieving very high accuracy on object recognition under ideal scene parameters (2) proposing counterfactual studies in order to further abstract from domain gap. General statements about ConvNets and Transformers are hard to make, given that architectures evolve. In contrast to previous explorations, we propose to study the most comparable architectures to date; ConvNext and Swin, due to their similar design choices and aligned training recipes. To further increase robustness, we study 4 different sizes of these networks with 2 different training datasets (ImageNet-1k and ImageNet-22k).

6 Conclusion

We conduct a study of Transformers and ConvNets by proposing NVD; a novel realistic synthetic dataset of naturalistic scene variations. By using our proposed Counterfactual Simulation Testing approach, we are able to fairly compare state-of-the-art ConvNets and Transformers and find evidence of differences in their ability to handle variations in object pose, scale, camera pose and occlusion. We release both NVD and our flexible and customizable scene generator.

Broader Impact Our work can be used to address model bias, e.g. face analysis networks with a respective simulator. Interactions exist with applications that can negative for society, for example, knowing weaknesses of networks can lead to simple ways of attacking them using natural inputs. As always, improving computer vision models goes hand in hand with enabling applications such as malicious surveillance. We include a more detailed discussion in the supplementary material.

References

[1] M. A. Alcorn, Q. Li, Z. Gong, C. Wang, L. Mai, W.-S. Ku, and A. Nguyen. Strike (with) a pose: Neural networks are easily fooled by strange poses of familiar objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4845–4854, 2019.

[2] A. Aldahdooh, W. Hamidouche, and O. Deforges. Reveal of vision transformers robustness against adversarial attacks. *arXiv preprint arXiv:2106.03734*, 2021.

[3] O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, et al. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020.

[4] Y. Bai, J. Mei, A. Yuille, and C. Xie. Are transformers more robust than cnns? In *NeurIPS*, 2021.

[5] H. Bao, L. Dong, and F. Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.

[6] A. Barbu, D. Mayo, J. Alverio, W. Luo, C. Wang, D. Gutfreund, J. Tenenbaum, and B. Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32, 2019.

[7] S. Beery, Y. Liu, D. Morris, J. Piavis, A. Kapoor, N. Joshi, M. Meister, and P. Perona. Synthetic examples improve generalization for rare classes. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[8] P. Benz, S. Ham, C. Zhang, A. Karjauv, and I. S. Kweon. Adversarial robustness comparison of vision transformer and mlp-mixer to cnns. *arXiv preprint arXiv:2110.02797*, 2021.

[9] Y.-H. Chen, W.-Y. Chen, Y.-T. Chen, B.-C. Tsai, Y.-C. Frank Wang, and M. Sun. No more discrimination: Cross city adaptation of road scene segmenters. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[10] S. Chiappa. Path-specific counterfactual fairness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7801–7808, 2019.

[11] T. Christensen and B. Connault. Counterfactual sensitivity and robustness. *arXiv preprint arXiv:1904.00989*, 2019.

[12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[13] E. Denton, B. Hutchinson, M. Mitchell, T. Gebru, and A. Zaldivar. Image counterfactual sensitivity analysis for detecting unintended bias. *arXiv preprint arXiv:1906.06439*, 2019.

[14] M. Ding, B. Xiao, N. Codella, P. Luo, J. Wang, and L. Yuan. Davit: Dual attention vision transformers. *arXiv preprint arXiv:2204.03645*, 2022.

[15] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

[16] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun. Carla: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR, 2017.

[17] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig. Virtual worlds as proxy for multi-object tracking analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4340–4349, 2016.

[18] C. Gan, J. Schwartz, S. Alter, D. Mrowca, M. Schrimpf, J. Traer, J. De Freitas, J. Kubilius, A. Bhandwaldar, N. Haber, et al. Threedworld: A platform for interactive multi-modal physical simulation. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021.

[19] Y. Ganin, T. Kulkarni, I. Babuschkin, S. M. A. Eslami, and O. Vinyals. Synthesizing programs for images using reinforced adversarial learning. In *ICML*, 2018.

[20] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17(1):2096–2030, Jan. 2016.

[21] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019.

[22] Y. Goyal, Z. Wu, J. Ernst, D. Batra, D. Parikh, and S. Lee. Counterfactual visual explanations. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2376–2384. PMLR, 09–15 Jun 2019.

[23] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[24] D. Hendrycks and T. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.

[25] D. Hendrycks, K. Zhao, S. Basart, J. Steinhardt, and D. Song. Natural adversarial examples. *CVPR*, 2021.

[26] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. Efros, and T. Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pages 1989–1998. PMLR, 2018.

[27] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[28] J. Johnson, B. Hariharan, L. Van Der Maaten, L. Fei-Fei, C. Lawrence Zitnick, and R. Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2910, 2017.

[29] J. Joo and K. Kärkkäinen. Gender slopes: Counterfactual fairness for computer vision models by attribute manipulation. In *Proceedings of the 2nd International Workshop on Fairness, Accountability, Transparency and Ethics in Multimedia*, pages 1–5, 2020.

[30] A. Kar, A. Prakash, M.-Y. Liu, E. Cameracci, J. Yuan, M. Rusiniak, D. Acuna, A. Torralba, and S. Fidler. Meta-sim: Learning to generate synthetic datasets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[31] A. Kortylewski, B. Egger, A. Schneider, T. Gerig, A. Morel-Forster, and T. Vetter. Empirically analyzing the effect of dataset biases on deep face recognition systems. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 2093–2102, 2018.

[32] A. Kortylewski, J. He, Q. Liu, and A. L. Yuille. Compositional convolutional neural networks: A deep architecture with innate robustness to partial occlusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[33] A. Kortylewski, Q. Liu, H. Wang, Z. Zhang, and A. Yuille. Combining compositional models and deep networks for robust object classification under occlusion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1333–1341, 2020.

[34] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[35] M. J. Kusner, J. Loftus, C. Russell, and R. Silva. Counterfactual fairness. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[36] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[37] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[38] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie. A convnet for the 2020s. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[39] G. Louppe and K. Cranmer. Adversarial variational optimization of non-differentiable simulators. *arXiv preprint arXiv:1707.07113*, 2017.

[40] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4040–4048, 2016.

[41] M. M. Naseer, K. Ranasinghe, S. H. Khan, M. Hayat, F. Shahbaz Khan, and M.-H. Yang. Intriguing properties of vision transformers. *Advances in Neural Information Processing Systems*, 34, 2021.

[42] S. Paul and P.-Y. Chen. Vision transformers are robust learners. *AAAI*, 2022.

[43] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1406–1415, 2019.

[44] X. Peng, B. Usman, N. Kaushik, D. Wang, J. Hoffman, and K. Saenko. Visda: A synthetic-to-real benchmark for visual domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 2021–2026, 2018.

[45] N. Pinto, J. J. DiCarlo, and D. D. Cox. Establishing good benchmarks and baselines for face recognition. In *Workshop on Faces In'Real-Life'Images: Detection, Alignment, and Recognition*, 2008.

[46] M. Raghu, T. Unterthiner, S. Kornblith, C. Zhang, and A. Dosovitskiy. Do vision transformers see like convolutional neural networks? In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 12116–12128. Curran Associates, Inc., 2021.

[47] S. R. Richter, V. Vineet, S. Roth, and V. Koltun. Playing for data: Ground truth from computer games. In *European Conference on Computer Vision*, pages 102–118. Springer, 2016.

[48] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3234–3243, 2016.

[49] N. Ruiz, S. Schulter, and M. Chandraker. Learning to simulate. In *International Conference on Learning Representations*, 2018.

[50] N. Ruiz, B.-J. Theobald, A. Ranjan, A. H. Abdelaziz, and N. Apostoloff. Morphgan: One-shot face synthesis gan for detecting recognition bias. In *32nd British Machine Vision Conference 2021, BMVC 2021, Virtual Event, UK*, 2021.

[51] R. Shao, Z. Shi, J. Yi, P.-Y. Chen, and C.-J. Hsieh. On the adversarial robustness of visual transformers. *arXiv e-prints*, pages arXiv–2103, 2021.

[52] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[53] M. Tan and Q. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[54] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jegou. Training data-efficient image transformers amp; distillation through attention. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10347–10357. PMLR, 18–24 Jul 2021.

[55] Y.-H. Tsai, W.-C. Hung, S. Schulter, K. Sohn, M.-H. Yang, and M. Chandraker. Learning to adapt structured output space for semantic segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[56] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017.

[57] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 568–578, October 2021.

[58] L. Yuan, Y. Chen, T. Wang, W. Yu, Y. Shi, Z.-H. Jiang, F. E. Tay, J. Feng, and S. Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 558–567, October 2021.

[59] H.-Y. Zhou, C. Lu, S. Yang, and Y. Yu. Convnets vs. transformers: Whose visual representations are more transferable? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2230–2238, 2021.

[60] H. Zhu, P. Tang, A. L. Yuille, S. Park, and J. Park. Robustness of object recognition under extreme occlusion in humans and computational models. In A. K. G. 0001, C. M. Seifert, and C. Freksa, editors, *Proceedings of the 41th Annual Meeting of the Cognitive Science Society, CogSci 2019: Creativity + Cognition + Computation, Montreal, Canada, July 24-27, 2019*, pages 3213–3219. cognitivesciencesociety.org, 2019.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] Extended discussion in supp. material.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes] Supp. Material
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] No people involved in study.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] No personally identifiable information.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]