3D Common Corruptions for Object Recognition

Anonymous Authors[1]

Abstract

We introduce a set of image transformations that can be used as corruptions to evaluate the robustness of models. The primary distinction of the proposed transformations is that, unlike existing approaches such as Common Corruptions (Hendrycks & Dietterich, 2019), the geometry of the scene is incorporated in the transformations – thus leading to corruptions that are more likely to occur in the real world. We apply these corruptions to the ImageNet validation set to create 3D Common Corruptions (ImageNet-3DCC) benchmark. The evaluations on recent ImageNet models with robustness mechanisms show that ImageNet-3DCC is a challenging benchmark for object recognition task. Furthermore, it exposes vulnerabilities that are not captured by Common Corruptions, which can be informative during model development.

1. Introduction

Computer vision models deployed in the real world will encounter naturally occurring distribution shifts from their training data. These shifts range from lower-level distortions, such as motion blur and illumination changes, to semantic ones, like object occlusion. Each of them represents a possible failure mode of a model and has been frequently shown to result in profoundly unreliable predictions (Dodge & Karam, 2017; Hendrycks & Dietterich, 2019; Szegedy et al., 2013; Jo & Bengio, 2017; Geirhos et al., 2020). Thus, a systematic testing of vulnerabilities to these shifts is critical before deploying these models in the real world.

This work presents a set of distribution shifts in order to test models' robustness for object recognition task. To achieve this, we leverage recently proposed framework in (Kar et al., 2022), denoted as 3D Common Corruptions (3DCC). In contrast to previously proposed shifts which perform uniform 2D modifications over the image, such as Common Corruptions (2DCC, or equivalently, ImageNet-2DCC) (Hendrycks & Dietterich, 2019), 3DCC incorporates 3D information to generate corruptions that are consistent with the scene geometry. This leads to shifts that are more likely to occur in the real world.

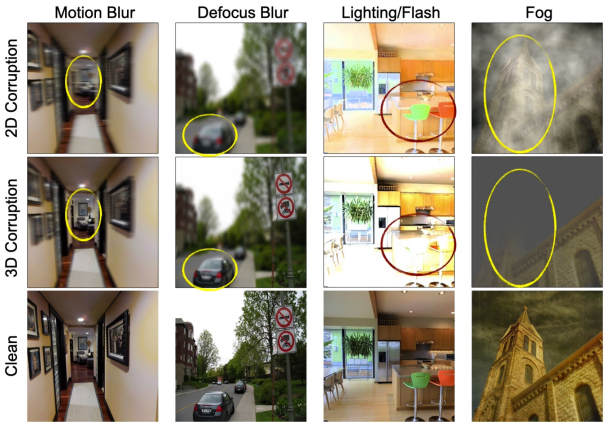

Figure 1. **Using 3D information to generate real-world corruptions.** This is shown for image samples that are taken from different datasets (ImageNet (Deng et al., 2009), COCO (Lin et al., 2014), Taskonomy (Zamir et al., 2018)). The top row shows sample 2D corruptions applied uniformly over the image, e.g. as in Common Corruptions (Hendrycks & Dietterich, 2019), disregarding 3D information. This leads to corruptions that are unlikely to happen in the real world, e.g. having the same motion blur over the entire image irrespective of the distance to camera (top left). Middle row shows their 3D counterparts from 3D Common Corruptions (3DCC) (Kar et al., 2022). The circled regions highlight the effect of incorporating 3D information. More specifically, in 3DCC, **1. motion blur** has a *motion parallax* effect where objects further away from the camera seem to move less, **2. defocus blur** has a *depth of field* effect, akin to a large aperture effect in real cameras, where certain regions of the image can be selected to be in focus, **3. lighting** takes the scene geometry into account when illuminating the scene, **4. fog** gets denser further away from the camera. We apply corruptions from 3DCC over ImageNet validation images to create **ImageNet-3DCC** benchmark.

Using the pipeline provided in (Kar et al., 2022), we apply 12 corruptions from 3DCC on ImageNet (Deng et al., 2009) validation set. We denote the resulting set as *ImageNet-3DCC* (See Fig. 1 for examples). ImageNet-3DCC addresses several aspects of the real world, such as camera motion, weather, depth of field, and lighting. Figure 2 provides an overview of all corruptions in ImageNet-3DCC.

We show that the performance of the methods aiming to improve robustness of ImageNet models, including those with diverse data augmentation, reduce drastically under ImageNet-3DCC. Thus, it can serve as a challenging testbed

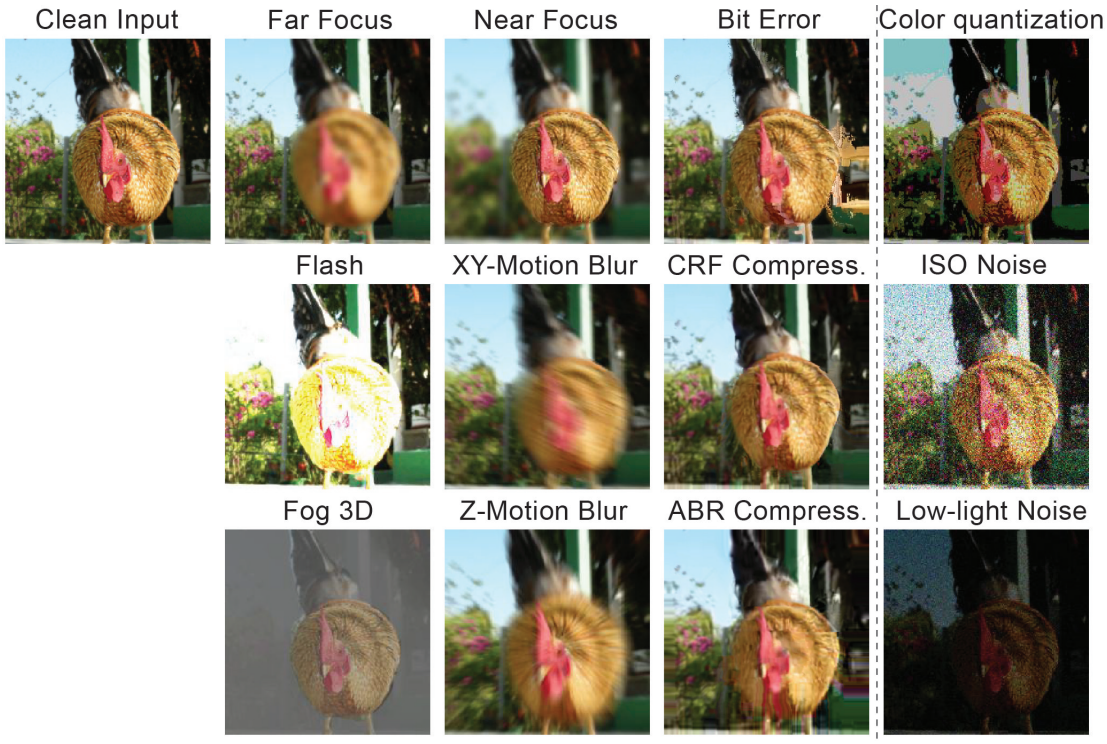

Figure 2. **ImageNet-3DCC benchmark.** We propose a *diverse* set of new corruption operations ranging from defocusing (near/far focus) to lighting changes. As ImageNet (Deng et al., 2009) dataset does not come with 3D labels, we leverage a state-of-the-art depth estimator to obtain depth predictions, and then apply the methods from (Kar et al., 2022) using predicted depth to generate corruptions (See Sec. 2.1 for details). A subset of the corruptions marked in the last column are novel and commonly faced in the real world, but are not 3D based. We include them in our benchmark.

for real-world corruptions, especially those that depend on scene geometry. It also exposes vulnerabilities that are not captured by ImageNet-C, hence it can be used to better assess generalization of the existing models which may have overfitted to ImageNet-C, which can be informative for development of better robustness mechanisms.

2. ImageNet-3DCC Benchmark

2.1. Corruption Types

Following (Kar et al., 2022), we define different corruption types, namely *depth of field*, *camera motion*, *lighting*, *video*, *weather*, and *noise*, resulting in 12 corruptions in ImageNet-3DCC. Most of the corruptions require an RGB image and scene depth, except the noise ones that can be generated from RGB image directly. As ImageNet images do not have depth labels, we generate depth predictions using a state-of-the-art depth estimator trained on Omnidata (Eftekhar et al., 2021) dataset with 3D data augmentations (Kar et al., 2022) and consistency constraints (Zamir et al., 2020). Note that the corruptions generated using predicted depth from the state-of-the-art models (Ranftl et al., 2021; Eftekhar et al., 2021) are similar to those generated from ground truth depth, as shown in (Kar et al., 2022) (Sec. 5.2.4). Furthermore, as shown there, the 3D corruptions are also applicable to other object recognition datasets without 3D labels as well, e.g. COCO (Lin et al., 2014).

To generate the corruptions, we use the methods from (Kar et al., 2022), as explained in more detail below. Note that the semantic corruptions in (Kar et al., 2022) would require a mesh with semantic annotations to generate. We dropped them as ImageNet does not have those labels.

Depth of field corruptions create refocused images. They keep a part of the image in focus while blurring the rest. We consider a layered approach (Eftekhar et al., 2021; Barsky & Kosloff, 2008) that splits the scene into multiple layers. For each layer, the corresponding blur level is computed using the pinhole camera model. The blurred layers are then composited with alpha blending. We generate *near focus* and *far focus* corruptions by changing the focus region to the near or far part of the scene.

Camera motion creates blurry images due to camera movement during exposure. To generate this effect, we first transform the input image into a point cloud using the depth information. Then, we define a trajectory (camera motion)

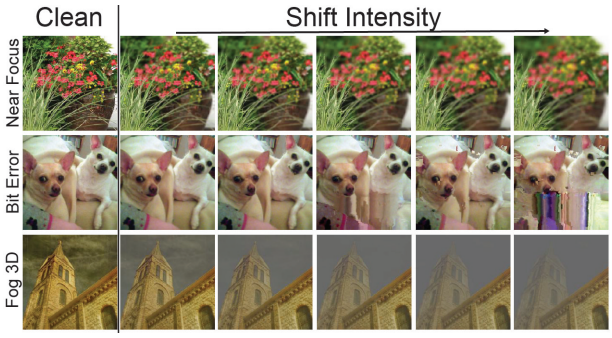

Figure 3. **Visualizations of ImageNet-3DCC with increasing shift intensities**. Increasing the shift intensity results in larger blur, more artifacts, and denser fog.

and render novel views along this trajectory. As the point cloud was generated from a single RGB image, it has incomplete information about the scene when the camera moves. Thus, the rendered views will have disocclusion artifacts. To alleviate this, we apply an inpainting method from (Niklaus et al., 2019). The generated views are then combined to obtain parallax-consistent motion blur. We define *XY-motion blur* and *Z-motion blur* when the main camera motion is along the image XY-plane or Z-axis, respectively.

Video corruptions arise during the processing and streaming of videos. Using the scene 3D, we create a video using multiple frames *from a single image* by defining a trajectory, similar to motion blur. Inspired by (Yi et al., 2021), we generate *average bit rate (ABR)* and *constant rate factor (CRF)* as H.265 codec compression artifacts, and *bit error* to capture corruptions induced by imperfect video transmission channel. After applying the corruptions over the video, we pick a single frame as the final corrupted image.

Weather corruptions degrade visibility by obscuring parts of the scene due to disturbances in the medium. We define a single corruption and denote it as *fog 3D* to differentiate it from the fog corruption in 2DCC. We use the standard optical model for fog (Fattal, 2008; Sakaridis et al., 2018; Von Bernuth et al., 2019), similar to (Kar et al., 2022).

Lighting corruptions change scene illumination by modifying the original illumination. As ImageNet does not have full scene geometry information, it is not possible to perform ray-tracing. Thus, we only consider a *flash* corruption, where illumination decreases with increasing depth.

Noise corruptions arise from imperfect camera sensors. For *low-light noise*, we decreased the pixel intensities and added Poisson-Gaussian distributed noise to reflect the low-light imaging setting (Foi et al., 2008). *ISO noise* also follows a Poisson-Gaussian distribution, with a fixed photon noise (modeled by a Poisson), and varying electronic noise (modeled by a Gaussian). We also included *color quantization* as another corruption that reduces the bit depth of the RGB image. Only this subset of our corruptions is not based on 3D information.

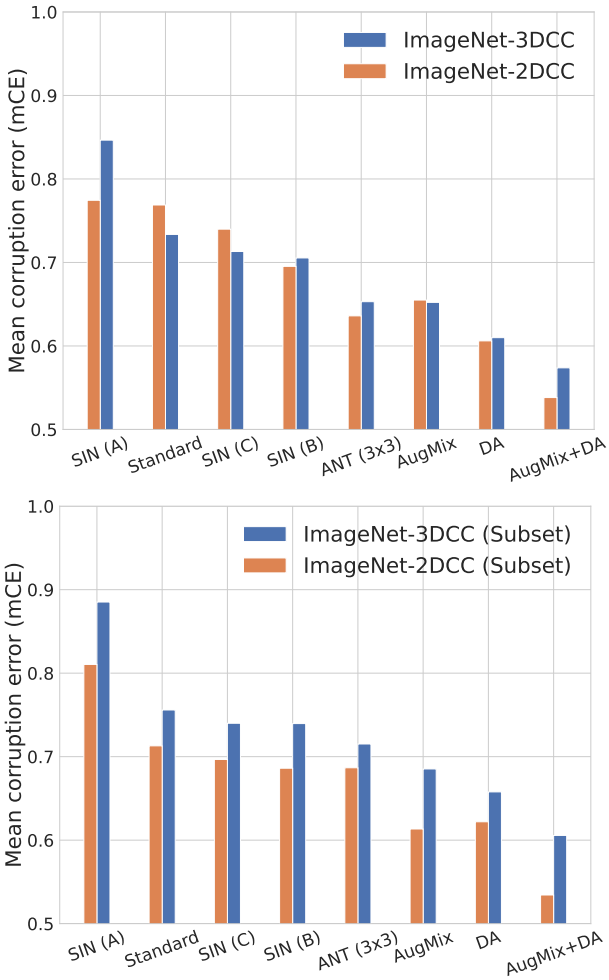

Figure 4. **Robustness on ImageNet-3DCC and ImageNet-2DCC.** Errors on ImageNet validation images corrupted by 3DCC and 2DCC are computed for the models in robustness leaderboards (Hendrycks & Dietterich, 2019; Croce et al., 2020). **Top:** mCEs are computed over all the corruptions. **Bottom:** mCEs are computed for a subset of corruptions that exists in both benchmarks (e.g. 2D defocus blur vs its 3D version). See the text (Sec. 3) for details.

2.2. Dataset and Evaluation Criteria

We applied the corruptions on 50k ImageNet validation images. For all the corruptions, we follow the protocol in ImageNet-2DCC and define 5 shift intensities. We also calibrate the shift intensities so that the average SSIM (Wang et al., 2004) values of images for an ImageNet-3DCC corruption is similar to its counterpart in 2DCC. For the corruptions that do not have a counterpart in 2DCC, we adjust the distortion parameters to increase shift intensity while staying in a similar SSIM range as the others. The dataset can be accessed from the project page. Figure 3 shows example

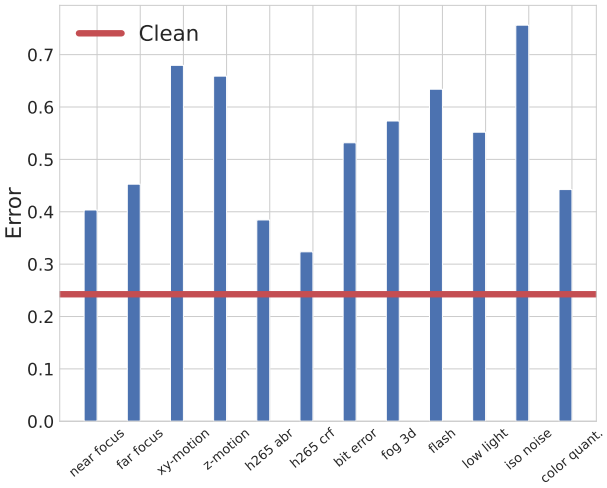

Figure 5. **Detailed breakdown of performance on ImageNet-3DCC.** The benchmark can expose models' sensitivity to a wide range of corruptions. We show this for a standard ResNet-50 model from PyTorch model zoo by averaging errors over five shift intensities. The red line denotes the performance on clean data.

corruptions with different shift intensities.

As evaluation criteria, we follow 2DCC for compatibility and compute mean corruption error (mCE) by dividing the models errors by AlexNet (Krizhevsky et al., 2012) errors and averaging over corruptions.

3. Experiments

Models evaluated: We evaluate the robust ImageNet models (Geirhos et al., 2018; Rusak et al., 2020; Hendrycks et al., 2019; 2021) from RobustBench (Croce et al., 2020) and ImageNet-2DCC (Hendrycks & Dietterich, 2019) leaderboards. We directly use the provided weights, i.e. no training or fine-tuning is performed.

As shown in Fig. 4 (top), the performance of models degrade significantly, including those with diverse augmentations. Thus, ImageNet-3DCC can serve as a challenging benchmark for object recognition task. As expected, the general trends are similar between the two benchmarks as 2D and 3D corruptions are not completely disjoint. A similar observation was also made in (Mintun et al., 2021) even when the corruptions are *designed to be dissimilar* to 2DCC. Still, there are notable differences that can be informative during model development by exposing trends and vulnerabilities that are not captured by 2DCC. For example, ANT (Rusak et al., 2020) has better mCE on 2DCC compared to AugMix (Hendrycks et al., 2019), while they perform similarly on 3DCC. Likewise, combining DeepAugment (Hendrycks et al., 2021) with AugMix improved the performance on 2DCC significantly more than 3DCC.

To further understand the differences, we consider a subset of corruptions that exists in both benchmarks (e.g. 2D defocus blur vs its 3D version), namely *near focus*, *far focus*, *xy-motion blur*, *z-motion blur*, *fog 3d*, and *flash* from ImageNet-3DCC and *defocus blur*, *motion blur*, *zoom blur*, *fog* and *brightness* from 2DCC. We then compute the mCEs only on these subsets. The results shown in Fig. 4 (bottom) further reflects the differences where **1.** all models have consistently higher normalized errors (mCEs) on 3D corruptions compared to their 2DCC counterparts and **2.** certain models, e.g. AugMix and AugMix+DA, face a larger drop in performance on 3DCC compared to the other models, indicating that AugMix may be biased towards 2DCC.

Finally, in Fig. 5, we provide a detailed breakdown of performance on ImageNet-3DCC for a standardly trained ResNet-50 (He et al., 2016) from PyTorch (Paszke et al., 2019) model zoo. The performance degrades significantly from the performance on clean data, while some corruptions yield more severe errors than the others, e.g. *xy-motion blur* vs *h265 crf*. Examining this non-uniformity in performance could be informative to design better robustness mechanisms, e.g. targeted data augmentation, depending on the practical setting of interest.

4. Conclusion

We introduce ImageNet-3DCC to test model robustness against real-world distribution shifts, particularly those centered around 3D. Experiments demonstrate that ImageNet-3DCC is a challenging benchmark that exposes model vulnerabilities under real-world plausible corruptions that are not captured by 2D corruptions. We believe incorporating 3D information into benchmarking opens up a promising direction for robustness research.

References

Barsky, B. A. and Kosloff, T. J. Algorithms for rendering depth of field effects in computer graphics. In *Proceedings of the 12th WSEAS international conference on Computers*, volume 2008. World Scientific and Engineering Academy and Society (WSEAS), 2008.

Croce, F., Andriushchenko, M., Sehwag, V., Debenedetti, E., Flammarion, N., Chiang, M., Mittal, P., and Hein, M. Robustbench: a standardized adversarial robustness benchmark. *arXiv preprint arXiv:2010.09670*, 2020.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. Ieee, 2009.

Dodge, S. and Karam, L. A study and comparison of human and deep learning recognition performance under visual

distortions. In *2017 26th International Conference on Computer Communication and Networks (ICCCN)*, pp. 1–7. IEEE, 2017.

Eftekhar, A., Sax, A., Malik, J., and Zamir, A. Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3d scans. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10786–10796, 2021.

Fattal, R. Single image dehazing. *ACM transactions on graphics (TOG)*, 27(3):1–9, 2008.

Foi, A., Trimeche, M., Katkovnik, V., and Egiazarian, K. Practical poissonian-gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing*, 17(10):1737–1754, 2008.

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018.

Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. Shortcut learning in deep neural networks. *arXiv preprint arXiv:2004.07780*, 2020.

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.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

Hendrycks, D., Mu, N., Cubuk, E. D., Zoph, B., Gilmer, J., and Lakshminarayanan, B. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*, 2019.

Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8340–8349, 2021.

Jo, J. and Bengio, Y. Measuring the tendency of cnns to learn surface statistical regularities. *arXiv preprint arXiv:1711.11561*, 2017.

Kar, O. F., Yeo, T., Atanov, A., and Zamir, A. 3d common corruptions and data augmentation. *CVPR*, 2022.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25: 1097–1105, 2012.

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.

Mintun, E., Kirillov, A., and Xie, S. On interaction between augmentations and corruptions in natural corruption robustness. *arXiv preprint arXiv:2102.11273*, 2021.

Niklaus, S., Mai, L., Yang, J., and Liu, F. 3d ken burns effect from a single image. *ACM Transactions on Graphics (TOG)*, 38(6):1–15, 2019.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pp. 8024–8035, 2019.

Ranftl, R., Bochkovskiy, A., and Koltun, V. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12179–12188, 2021.

Rusak, E., Schott, L., Zimmermann, R. S., Bitterwolf, J., Bringmann, O., Bethge, M., and Brendel, W. A simple way to make neural networks robust against diverse image corruptions. In *European Conference on Computer Vision*, pp. 53–69. Springer, 2020.

Sakaridis, C., Dai, D., and Van Gool, L. Semantic foggy scene understanding with synthetic data. *International Journal of Computer Vision*, 126(9):973–992, 2018.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Von Bernuth, A., Volk, G., and Bringmann, O. Simulating photo-realistic snow and fog on existing images for enhanced cnn training and evaluation. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 41–46. IEEE, 2019.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

Yi, C., Yang, S., Li, H., Tan, Y.-p., and Kot, A. Benchmarking the robustness of spatial-temporal models against corruptions. *arXiv preprint arXiv:2110.06513*, 2021.

Zamir, A., Sax, A., Yeo, T., Kar, O., Cheerla, N., Suri, R., Cao, Z., Malik, J., and Guibas, L. Robust learning through cross-task consistency. *arXiv preprint arXiv:2006.04096*, 2020.

Zamir, A. R., Sax, A., Shen, W., Guibas, L. J., Malik, J., and Savarese, S. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3712–3722, 2018.