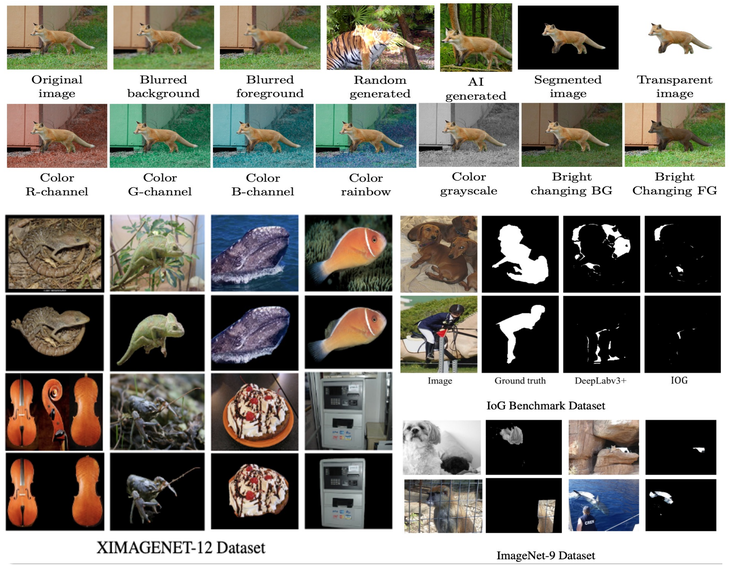
XIMAGENET-12: An Explainable Visual Benchmark Dataset for Model Robustness Evaluation

Qiang Li, Dan Zhang, Shengzhao Lei, Xun Zhao, Porawit Kamnoedboon, WeiWei Li, Junhao Dong, Shuyan Li

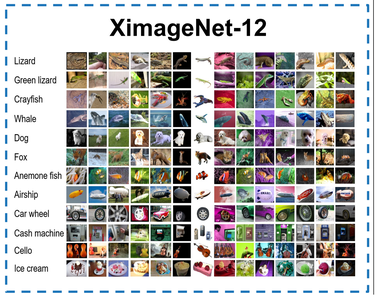
RWTH AACHEN UNIVERSITY · EPFL · University of Zurich[UZH] · accenture · NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE · UNIVERSITY OF CAMBRIDGE · UNIVERSITY OF AMSTERDAM · Shanghai Business School

CVPR SEATTLE, WA JUNE 17-21, 2024

Background: Despite the promising performance of existing visual models on public benchmarks, critically assessing their robustness for real-world applications remains an ongoing challenge. How should we evaluate the robustness performance of visual models when the background of the object changes? **What factors in the background matter?**



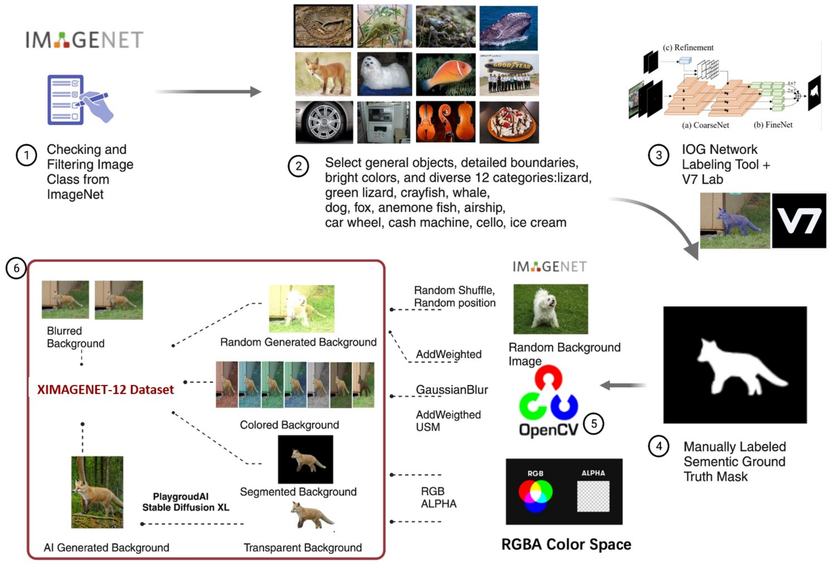
Motivation: Previous research has investigated the overarching phenomenon of contextual bias by training deep neural networks on foreground and background elements separately. These studies have shown that backgrounds can provide valuable visual hints or merely contribute to noise with SOTA visual models. However, they often fail to explore which factors lead to noise or signal results. Additionally, some inaccuracies in segmentation ground truth (GT) may lead to biased conclusions.



Contribution

❖ XImageNet-12

covers 12 categories, across 6 scenarios, including blur, randomly generated backgrounds, AI-generated backgrounds, segmented, transparent, and colored images, 15,410 manual semantic annotations, 12,248 GenAI Image, in total over 200K Image Benchmark. Has been downloaded over 100 times up-to-date in Kaggle.



❖ Quantitative Robustness Score Schema

the development of a quantitative robustness score, intended to measure models' generalization across these conditions.

Cross Scenarios

$$\sigma^2_{cross} = \frac{\sum_{i=1}^{n}(C(i)-\mu)^2}{n}$$

Same Scenarios

$$\sigma^2_{inner} = \frac{\sum_{i=1}^{n}(C'(i)-\mu)^2}{n}$$

Robustness Schema

$$S_{\text{robust}} = 1-(\sigma^2_{cross}+\sigma^2_{inner})$$

Here, μ means the best weight accuracy when the model is both trained and tested on the original scenario. C (i) means the model is trained on original images but tested on the i-th scenario. i 2 {0, 1,...,n} and n is the number of scenarios that we consider. C' (i) means that the model is both trained and tested on the i-th scenario.

Experiments & Analysis

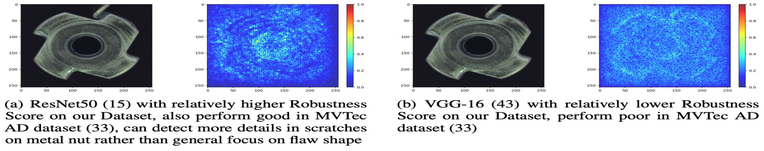
Main Results:

Table 1. Comparison of SOTA visual models with diverse scenarios. Here all the evaluation metrics are Top-1 Accuracy.

Pretrained Dataset	Model Name	Parameters (M)	Test Dataset (Top-1 Acc.)							
			Blur_bg	Blur_obj	Color_g	Color_b	Color_grey	Color_r	Rand_bg	Seg_img
ImageNet [7] (Original images) EX1	ResNet50 [11]	25.60	90.97%	88.17%	84.42%	86.98%	92.13%	89.03%	22.41%	68.55%
	VGG-16 [32]	138.4	89.92%	89.91%	78.64%	70.46%	81.48%	80.68%	24.58%	49.62%
	MobileNetV2 [29]	3.5	92.34%	88.52%	85.73%	88.67%	88.81%	89.33%	27.14%	66.43%
	EfficientNetB0 [34]	5.3	91.44%	90.86%	78.10%	82.45%	86.44%	83.65%	25.29%	53.56%
	EfficientNetB3 [34]	12.3	86.80%	84.53%	77.99%	81.22%	83.00%	83.85%	22.06%	69.67%
	DenseNet121 [13]	8.1	93.77%	88.92%	87.39%	87.33%	93.23%	88.21%	26.41%	69.67%
	ViT [8]	86.6	88.44%	90.77%	65.87%	62.82%	70.69%	66.53%	17.21%	49.01%
	Swin [22]	87.76	80.97%	81.57%	64.59%	65.91%	69.28%	64.41%	19.43%	44.57%
XImageNet-12 (*Scenarios) EX2	ResNet50 [11]	25.60	83.52%	80.24%	83.61%	84.45%	84.71%	80.40%	53.91%	85.76%
	VGG-16 [32]	138.4	74.85%	71.54%	74.18%	76.26%	77.58%	69.91%	70.25%	73.27%
	AlexNet [19]	61.1	81.60%	79.95%	81.96%	81.89%	81.31%	78.07%	46.29%	82.00%
	MobileNetV3 [12]	3.50	67.36%	67.88%	72.04%	74.25%	69.48%	64.79%	43.33%	78.85%
	DenseNet121 [13]	8,1	90.79%	86.57%	88.92%	89.96%	90.44%	87.37%	69.58%	91.60%
	ViT [8]	86.56	71.51%	70.21%	74.77%	75.96%	75.80%	71.14%	38.01%	78.69%
	Swin [22]	87.76	72.81%	75.02%	81.05%	81.96%	81.63%	76.42%	13.23%	80.64%

Variable	Estimate	P value	P value summary
Intercept	0.8986	< 0.0001	****
Model Name[EfficientNetB0 [34]]	-0.03444	0.0175	*
Model Name[EfficientNetB3 [34]]	-0.04111	0.0046	**
Model Name[DenseNet121 [13]]	0.01556	0.2821	ns
Model Name[MobileNetV2 [29]]	0.005556	0.7007	ns
Image Scenario[blur.background]	-0.0425	0.0288	*
Image Scenario[blur.object]	-0.07	0.0003	***
Image Scenario[image.g]	-0.1257	< 0.0001	****
Image Scenario[image.b]	-0.0985	< 0.0001	****
Image Scenario[image.grey]	**-0.06517**	0.0008	***
Image Scenario[image.r]	-0.087	< 0.0001	****
Image Scenario[Random Background with Real Environment]	**-0.7078**	< 0.0001	****
Image Scenario[Segmented_image]	-0.3012	< 0.0001	****
Image Class[1]	0.134	< 0.0001	****
Image Class[2]	-0.04867	0.0301	*
Image Class[3]	0.04	0.0745	ns
Image Class[4]	0.1004	< 0.0001	****
Image Class[5]	0.1333	< 0.0001	****
Image Class[6]	0.07667	0.0007	***
Image Class[7]	0.01044	0.6409	ns
Image Class[8]	0.09067	< 0.0001	****
Image Class[9]	0.09933	< 0.0001	****
Image Class[10]	0.1651	< 0.0001	****
Image Class[11]	-0.02244	0.3164	ns

Variable	Estimate	P value	P value summary
Intercept	**0.6672**	< 0.0001	****
Model Name[dpt.vit-b16 [8]]	**-0.1999**	< 0.0001	****
Model Name[upernet.swin [38]]	-0.2211	< 0.0001	****
Model Name[upernet.vit-b16 .ln.mln [38]]	-0.1695	< 0.0001	****
Model Name[pspnet.r50-d8 [46]]	-0.045	0.0106	*
Model Name[fpn.r50 [21]]	-0.2081	< 0.0001	****
Model Name[upernet.r50 [38]]	-0.05796	0.001	**
Image Scenario[blur.background]	0.01833	0.3576	ns
Image Scenario[blur.object]	-0.1571	< 0.0001	****
Image Scenario[image.g]	-0.07131	0.0004	***
Image Scenario[image.b]	-0.03952	0.0476	*
Image Scenario[image.grey]	-0.01929	0.3332	ns
Image Scenario[image.r]	-0.07702	0.0001	***
Image Scenario[segmented_image]	-0.08143	< 0.0001	****
Image Scenario[generated_background]	-0.1408	< 0.0001	****
Image Class[1]	0.07619	0.001	***
Image Class[2]	-0.06508	0.0048	**
Image Class[3]	0.05222	0.0234	*
Image Class[4]	0.08127	0.0004	***
Image Class[5]	0.2713	< 0.0001	****
Image Class[6]	0.3021	< 0.0001	****
Image Class[7]	0.1641	7.137	****
Image Class[8]	0.1548	6.73	****
Image Class[9]	0.2216	9.635	****
Image Class[10]	0.2689	11.69	****
Image Class[11]	0.04079	1.774	ns



(a) ResNet50 (15) with relatively higher Robustness Score on our Dataset, also perform good in MVTec AD dataset (33), can detect more details in scratches on metal nut rather than general focus on flaw shape

(b) VGG-16 (43) with relatively lower Robustness Score on our Dataset, perform poor in MVTec AD dataset (33)

Saliency Map Analysis on the MVTec AD Dataset

Findings:

- Different scenarios influence visual models in different degrees, and randomly substituting the background leads to the most severe performance drops.
- Models trained and tested with well-segmented foregrounds tend to perform well even if the backgrounds are missing.
- Our benchmark present a challenging task for state-of-the-art (SOTA) segmentation models as well, serving as an effective tool for measuring model performance in segmenting complex shapes or detecting detailed areas in AI-generated background images.