Do CLIPs Always Generalize Better than ImageNet Models?

Qizhou Wang*† Yong Lin*◦ Yongqiang Chen*‡
Ludwig Schmidt◇ Bo Han† Tong Zhang§

† TMLR Group, Hong Kong Baptist University
◦ The Hong Kong University of Science and Technology
‡ The Chinese University of Hong Kong
◇ University of Washington
§ University of Illinois Urbana-Champaign
https://counteranimal.github.io

Abstract

Large vision language models, such as CLIPs, have revolutionized modern machine learning. CLIPs have demonstrated great generalizability under distribution shifts, supported by an increasing body of literature. However, the evaluation datasets for CLIPs are variations primarily designed for ImageNet benchmarks, which may not fully reflect the extent to which CLIPs, e.g., pre-trained on LAION, robust to spurious correlations. To bridge the gap, we collect a real-world dataset called `CounterAnimal` that contains realistic spurious features found in animal photos. `CounterAnimal` consists of a) the `common` group: comprising animals on common backgrounds, and b) the `counter` group: including animals on unusual backgrounds. The performance drops from the `common` to `counter` groups quantify the reliance of models on spurious features (i.e., backgrounds) to predict the animals. We find that CLIPs trained on either LAION or the OpenAI data exhibit notable performance drops on the `counter` group. Surprisingly, we observe that single-modal models trained on ImageNet are more robust than CLIPs. We provide both theoretical and empirical explanations for why CLIPs still learn spurious features. Our findings suggest that distribution shifts remain an open problem for CLIPs, and one needs to be cautious about test setups when evaluating foundation models pre-trained on a significantly different scale and distribution.

1 Introduction

Large vision language models (LVLMs), which align images and text representations to comprehend general relationships within images across an unprecedented scale of real-world data, have presented a notable paradigm shift in modern machine learning. Many advanced LVLM works, such as CLIP (Radford et al., 2021), ALIGN (Jia et al., 2021), BASIC (Pham et al., 2023), Flamingo (Alayrac et al., 2022), and BLIP (Li et al., 2022), have exhibited remarkable performance across a wide range of vision and multimodal tasks, surpassing conventional ImageNet-trained models by a large margin (Deng et al., 2009). Henceforth, the huge success of LVLMs has drawn the focus of the community from ImageNet benchmarks to web-scale multi-modal datasets such as LAION (Schuhmann et al., 2022; Gadre et al., 2023).

A key signature of LVLMs is the impressive improvements in the effective robustness to various ImageNet-based distribution shifts, arising from the large-scale Contrastive Language Image Pre-training (CLIP) (Radford et al., 2021; Wortsman et al., 2022; Shi et al., 2023; Pham et al., 2023).[1] The performance boosts seem to demonstrate that CLIP resolves the distribution shifts and spark a rich discussion about the robustness of CLIP (Fang et al., 2022; Santurkar et al., 2023; Daunhawer et al., 2023; Xue et al., 2023; Mayilvahanan et al., 2023). However, *the elephant in the room* is that the adopted datasets to test the robustness of CLIP models are primarily designed for the ImageNet-based

[1]Following (Radford et al., 2021), we use CLIP as a name for the general training technique, not only their specific models.

Photos of **ice bears** in **snow** background (common, accu 97.62)

Photos of **ice bears** in **grass** background (counter, accu 70.91)

Figure 1. We showcase photos in `CounterAnimal` with the object label of `ice bear`, further separating into `common` and `counter` groups based on different backgrounds (i.e., `snow` and `grass`). The majority of `ice bears` has a `snow` background (i.e., `common`), although it is reasonable yet relatively less common to find `ice bears` in a grassy environment (i.e., `counter`). The performance drop in zero-shot inference by `CLIP-LAION400M-ViT/B/32` is significant, from the `common` group (97.62%) to the `counter` group (70.91%).

models (Radford et al., 2021; Wortsman et al., 2022; Shi et al., 2023), which may not fully reflect the cases of CLIP due to a variety of issues such as data contamination (Mayilvahanan et al., 2023). It poses an important research question:

Is there a benchmark that reflects the exact reliance on spurious features of CLIPs?

To answer the question, we collect a new dataset named `CounterAnimal`, which is tailored to quantify the robustness of CLIP against real-world spurious features. Figure 1 presents several examples of `CounterAnimal` where data are divided into two groups, a) the `common` group: animals in frequently appeared backgrounds and b) the `counter` group: animals in less commonly yet still plausibly appeared backgrounds. The `common` part captures some real-world biases that CLIPs-trained on web-scale data may naturally inherit. Therefore, by comparing the performance on the `common` group against that on the `counter` group, one can quantify to what extent CLIPs rely on spurious features.

The `CounterAnmial` dataset is created based on raw photos collected from iNaturalist[2]. The construction pipeline has 4 steps, a) data collection: querying iNaturalist with each animal class, using the animal names from the ImageNet-1K dataset (Deng et al., 2009). b) Data curation: manually cleaning low-quality photos that contain ambiguity, noise, or picture corruption. c) Background labeling: manually annotating photos with their respective backgrounds, selected from the label space of the candidate backgrounds. d) Spurious discovering: preserving classes and associated data based on the decrease in zero-shot performance when shifting the backgrounds. The resulting dataset ends up with 7,174 `common` photos and 5,926 `counter` photos, covering a total of 45 animal classes. `CLIP-LAION400M-ViT/B/32` (Schuhmann et al., 2022) is used as the referred CLIP in spurious discovering (cf., Appendix B for model naming rules).

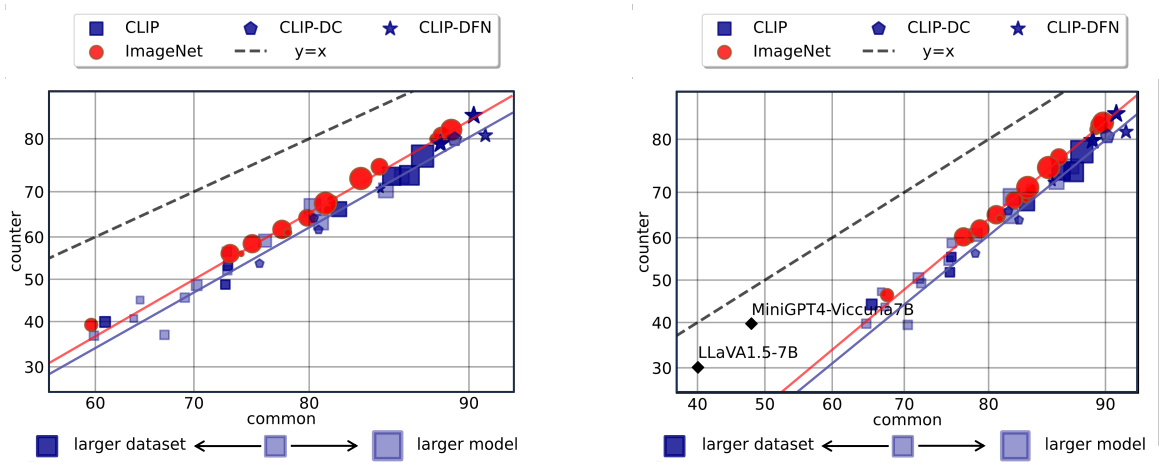
Table 1. Performance drops of `CLIP-LAION400M-ViT/B/32` on some example classes in the `CounterAnmial` dataset. "bkg" denotes the background label, "accu" (%) denotes the zero-shot accuracy (1 vs. 1000 setup), and "drop" (%) denotes the drop in accuracy between `common` and `counter` groups.

object label	common		counter		drop
	bkg	accu	bkg	accu	
ice bear	snow	97.62	grass	70.91	26.71
black swan	water	93.63	earth	68.87	24.76
flamingo	water	79.70	sky	55.45	24.25
vulture	sky	87.76	tree	41.84	45.92
dung beetle	earth	56.92	hand	17.02	39.90

We evaluate CLIP models on `CounterAnmial` with various backbones, e.g., ViT (Dosovitskiy et al., 2020), along with different pre-train datasets, e.g., LAION (Schuhmann et al., 2022). We also consider more advanced LVLMs like MiniGPT4 (Zhu et al., 2023) and LLaVA (Liu et al., 2023). We employ two evaluation setups for different families of models (cf., Appendix C), a) **1 vs. 1000 setup**: using the full ImageNet-1K class names as the candidate label space and b) **1 vs. 20 setup**: using the top-20 most confusing classes regarding `CLIP-LAION400M-ViT/B/32` as the candidate label space. We provide some of results in Table 1 and Figure 2, highlighting the key observations as follows:

CLIPs still learn spurious correlations. In Table 1, we observe a significant drop of `CLIP-LAION400M-ViT/B/32` checkpoint in accuracy from `common` to `counter` groups for these example classes. Furthermore, the observed biases

[2] https://www.inaturalist.org/observations



(a) **1 vs. 1000 setup** (full label space of ImageNet-1K)

(b) **1 vs. 20 setup** (20 most confusing labels per object class)

Figure 2. The `common` vs. `counter` performance (%) for CLIPs, ImageNet models, and more advanced LVLMs, i.e., MiniGPT4 and LLaVA. The size of markers indicates the backbone scales and the color shade of markers indicates the pre-train dataset scales. More detailed results can be found in Table 3-5, Appendix E and Figure 13-14. We highlight the CLIPs pre-trained on two high-quality datasets, i.e., DataComp (CLIP-DC) and Data Filtering Networks (CLIP-DFN). We linearly fit the trends for CLIPs (CLIP, CLIP-DC, and CLIP-DFN jointly) and ImageNet models, illustrating their effective robustness. We depict the perfect trend (i.e., $y = x$) where models do not learn any spurious correlation.

in `CLIP-LAION400M-ViT/B/32` also generalize to other models pre-trained on `LAION400M`, cf., Figure 5(a). Hence, CLIP may still learn the spurious features in `LAION400M`, which can be captured by `CounterAnimal`.

`CounterAnimal` reflects general biases in large-scale pre-train datasets. In Figure 2, we evaluate a wider range of CLIPs, alongside various backbones and pre-train datasets. We observe that other CLIP checkpoints also exhibit a non-trivial performance drop from `common` to `counter` groups. It implies that `CounterAnimal` characterizes some general spurious correlations that are commonly present in large-scale real-world multimodal datasets.

ImageNet models are more robust to spurious correlations captured by `CounterAnimal`. Figure 2 depicts the performance of ImageNet-trained models (colored in red). Compared with LVLMs (colored in blue), we find that ImageNet models exhibit better robustness to the presented spurious correlations. It is evident from Figure 2 by the superior performance of ImageNet models on the `counter` groups and similar `common` performance with respect to CLIPs. Our findings contrast with previous studies that evaluated distribution shifts using ImageNet variants (Radford et al., 2021; Shi et al., 2023), indicating that CLIP does not necessarily generalize better than ImageNet models.

Larger CLIPs are more robust. Shown also in Figure 2, we use the sizes and the color shades of the markers to indicate the scales of backbones and the pre-train datasets, respectively. Overall, larger CLIP backbones (i.e., larger markers) can improve the `counter` performance, implying that scaling up backbones may enhance robustness against spurious features. In contrast, increasing the scale of the pre-train dataset (i.e., darker markers) does not yield the same improvement, implying that collecting more data alone cannot rectify much bias, which provides a new understanding in addition to the data-centric perspective (Fang et al., 2022; Mayilvahanan et al., 2023).

CLIPs trained on high-quality data are more robust. We discern two classes of CLIPs, a) those pre-trained on high-quality datasets, i.e., DataComp (Gadre et al., 2023) (CLIP-DC) and Data Filtering Networks (Fang et al., 2023) (CLIP-DFN), and b) those pre-trained on other datasets that lack stringent curation (CLIP). We observe that CLIPs pre-trained on high-quality data generally exhibit superior robustness, suggesting that improving data quality is still a promising way against spurious features (Gadre et al., 2023).

CLIP objective may not offer additional robustness. Complementing to our empirical observations, we also provide theoretical explanations for why CLIPs learn spurious features. We conduct confirmatory experiments that finetune

Table 2. The object names in the `CounterAnimal` dataset as well as the background names of the `common` and `counter` groups. The full names of labels are presented following the fashion of the ImageNet-1K dataset.

ID	object label	common	counter	ID	object label	common	counter	ID	object label	common	counter
1	ostrich, struthio camelus	ground	water	2	brambling, Fringilla montifringilla	grass	sky	3	bulbul	sky	grass
4	water ouzel, dipper	water	ground	5	vulture	sky	tree	6	bullfrog, rana catesbeiana	water	ground
7	loggerhead, loggerhead turtle, caretta caretta	water	ground	8	box turtle, box tortoise	grass	earth	9	common iguana,iguana iguana iguana	earth	shrub
10	whiptail, whiptail lizard	earth	human	11	agama	rock	tree	12	african crocodile, nile crocodile, crocodylus niloticus	earth	grass
13	hognose snake, puff adder, sand viper	earth	grass	14	king snake kingsnake	earth	grass	15	garter snake grass snake	grass	earth
16	water snake	water	ground	17	harvestman, daddy longlegs, Phalangium opilio	shrub	rock	18	scorpion	indoor	outdoor
19	tarantula	sand	grass	20	centipede	indoor	grass	21	black grouse	grass	tree
22	ptarmigan	snow	grass	23	prairie chicken, prairie grouse, prairie fowl	grass	snow	24	sulphur-crested cockatoo, Kakatoe galerita, cacatua galerita	tree	grass
25	black swan, cygnus atratus	water	ground	26	echidna, spiny anteater, anteater	grass	tree	27	black stork ciconia nigra	grass	sky
28	flamingo	water	sky	29	bittern	grass	tree	30	pelican	water	sky
31	sea lion	sand	water	32	african hunting dog, hyena dog, cape hunting dog, lycaon pictus	grass	tree	33	hyena, hyaena	grass	road
34	red fox, vulpes vulpes	grass	road	35	arctic fox, white fox, alopex lagopus	snow	grass	36	jaguar, panther, Panthera onca, Felis onca	water	tree
37	lion, king of beasts, panthera leo	grass	tree	38	cheetah, chetah, acinonyx jubatus	grass	tree	39	ice bear, polar bear, ursus maritimus, thalarctos maritimus	snow	grass
40	dung beetle	earth	human	41	cicada, cicala	tree	human	42	beaver	water	grass
43	bighorn, bighorn sheep, cimarron	grass	rock	44	mink	grass	water	45	otter	water	tree

pre-trained CLIPs onto datasets with synthetic spurious features, further verifying that the CLIP objective can not offer additional robustness over standard single-modal supervised training.

Comparison with previous works. To the best of our knowledge, our work presents the first systematical data curation method as well as the benchmark to evaluate the robustness of CLIPs, which complements the literature on understanding CLIPs. Although there are several seminal works discussing the failure cases of LVLMs, they are limited in either the quality or the scale that may not fully reflect the robustness of CLIPs. Specifically, (Tong et al., 2023; Yang et al., 2023; Tong et al., 2024) leverage other single-modal or multi-modal models (Minderer et al., 2022; Oquab et al., 2023) to detect the learned shorcuts. However, the quality of the curated datasets highly relies on the robustness of the referred models. The detected spurious correlations by (Tong et al., 2023; Yang et al., 2023; Tong et al., 2024) are limited to specific classes or backbones, and may not systematically reflect the biases of the general large-scale multimodal pre-training datasets. In addition, our work also explains why LVLMs such as MiniGPT-4 (Chen et al., 2023; Zhu et al., 2023) or LLaVA (Liu et al., 2023) hallucinate objects (Li et al., 2023), as LVLMs typically incorporate the CLIP trained encoder to extract visual signals.

Moreover, our results challenge the previous beliefs inspired by the impressive robustness improvements evaluated on ImageNet variant test sets (Daunhawer et al., 2023; Santurkar et al., 2023; Zhang et al., 2023; Xue et al., 2023; Mayilvahanan et al., 2023). Our findings suggest that distribution shifts remain an open problem for CLIPs. We need to be cautious about test setups when evaluating foundation models pre-trained on a significantly different scale and distribution over the conventional ImageNet-based models.

2 Dataset Construction

We introduce the curation pipeline of our new dataset `CounterAnimal`, tailored for CLIPs to investigate spurious correlations. The pipeline consists of 4 steps as follows:

Data Collection. We query animal names listed in the ImageNet-1K dataset and collect raw data via the search interface of iNaturalist. We retrieve the latest 300-800 photos per class, organizing them based on query labels.

Data Curation. The collected raw samples are highly susceptible to various kinds of noises and ambiguities. Hence, we manually cleanse the low-quality samples that encounter any of the following situations:

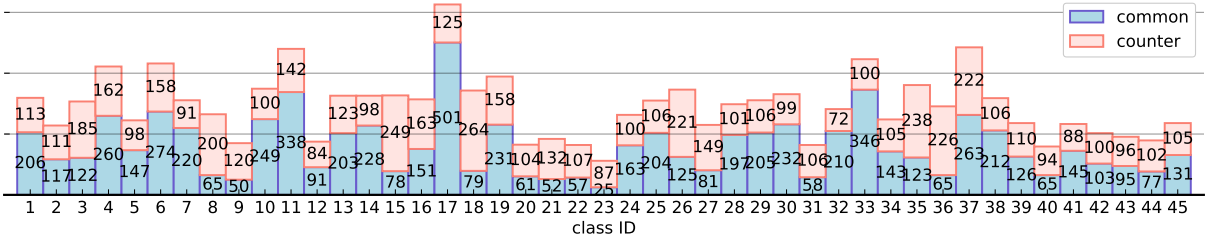


Figure 3. The data layout in the `CounterAnimal` dataset for `common` and `counter` groups across various classes. The horizontal axis denotes the class ids and the vertical axis denotes the numbers of photos for the `common` and `counter` groups, respectively.

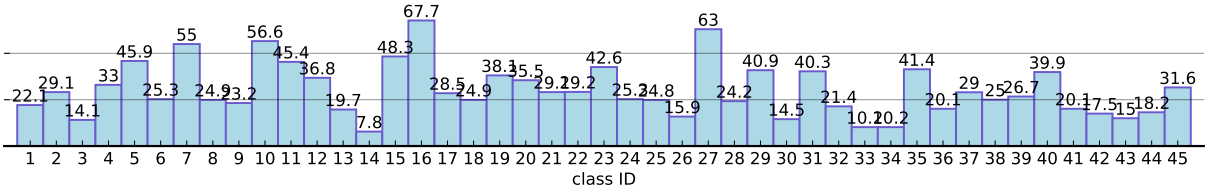


Figure 4. The 1 vs. 1000 drop (%) between the `common` and `counter` groups for `CLIP-LAION400M-ViT/B/32`. The horizontal axis denotes the class ids and the vertical axis denotes the percentage points of decline. A more detailed report can be found in Appendix E.

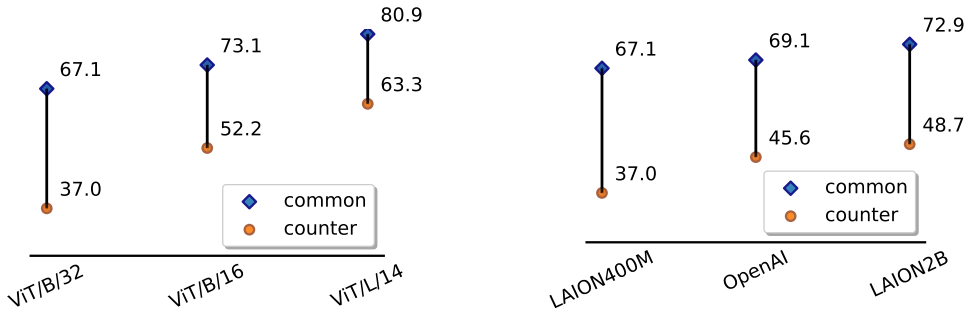
- **Label Noise**. It relates to the situations where photos that are not from the associated queried classes.
- **Feature Noise**. It relates to the situations where photos that contain severe feature corruptions.
- **Ambiguity**. It relates to the situations where photos that contain multiple object classes.
- **Clarity**. It relates to the situations where photos without animal objects at the major positions.

Background labeling. We consider a typical form of spurious features that the backgrounds of photos contain certain spurious features (Sagawa et al., 2020a). To identify data that exhibit such biases to CLIPs, we manually label the backgrounds of the curated data. The class space of the background labels is defined in the following:

`ground, water, earth, sand, snow, grass, human, sky, road, rock, shrub, indoor, tree, outdoor, road.`

Note that the class space of backgrounds is not entirely orthogonal with each other due to the inherent ambiguity of real-world backgrounds. Nevertheless, we try our best to discern the assigned background labels within each class.

Spurious Discovery. For each class, we quantify the impacts of spurious correlations to CLIPs by comparing their performances on samples across different backgrounds. We take those classes as containing spurious features onto which we observe a large decrease in accuracy when changing backgrounds. In realization, we adopt the checkpoint of `CLIP-LAION400M-ViT/B/32` for evaluation, where the prompt for its text encoder is "`A photo of <object label>`.", and the space of `<object label>` is the ImageNet-1K class names, i.e., the 1 vs. 1000 setup. Then, the classes where the zero-shot accuracies of the CLIP model vary by more than 5% when changing the background, are considered as the cases where CLIP learns the spurious features. The data associated with the preserved classes and backgrounds are used to create our final dataset. Photos with the highest CLIP accuracy are assigned to the `common` group, and those with the lowest CLIP accuracy are assigned to the `counter` group.



(a) LAION400M (b) ViT/B/32

Figure 5. Results for varying CLIP setups beyond CLIP-LAION400M-ViT/B/32: a) Fixing the pre-train dataset to be LAION400M and b) fixing the backbone to be ViT/B/32. The 1 vs. 1000 results are given.

3 The CounterAnimal Dataset

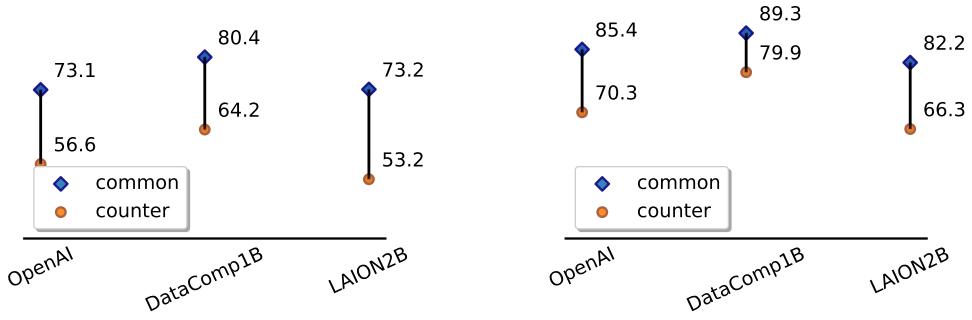
When CLIP models resort to the shortcut of data, the model performance will heavily correlate with the backgrounds presented in the common group. It will also be compromised when coming to the associated counter group. Our data organization offers a convenient way to evaluate spurious correlations (cf., Appendix A). We summarize the key properties of our dataset in Table 2 and Figure 3, covering detailed object/background names as well as data sizes for the common and counter groups. We also visualize the zero-shot performance gaps across different classes in Figure 4, employing CLIP-LAION400M-ViT/B/32 as our referred CLIP. Our preliminary results imply a heavy reliance for the CLIP models on the backgrounds, motivating a more comprehensive analysis in the following.

4 Evaluations

We evaluate a series of CLIPs on the CounterAnimal dataset for their zero-shot performance. For each class, we use the pre-defined prompt $\boldsymbol{x}_T^{\texttt{label}}$ of "A photo of <object label>." as in our data collection procedure. The prompt $\boldsymbol{x}_T^{\texttt{label}}$ goes through the text encoder g_T and the image $\boldsymbol{x}_I$ goes through the image encoder g_I, generating text and image embeddings, respectively. We compute the cosine similarity between them, i.e., $\texttt{CS}(\boldsymbol{x}_T^{\texttt{label}}, \boldsymbol{x}_I) = \frac{g_T(\boldsymbol{x}_T^{\texttt{label}}) \cdot g_I(\boldsymbol{x}_I)}{||g_T(\boldsymbol{x}_T^{\texttt{label}})|| \times ||g_I(\boldsymbol{x}_I)||}$, and classify the image $\boldsymbol{x}_I$ to the class that has the highest similarity, i.e., $\hat{y} = \arg\max_{\texttt{label} \in \mathcal{L}} \texttt{CS}(\boldsymbol{x}_T^{\texttt{label}}), \boldsymbol{x}_I)$. By default, we use the label space $\mathcal{L}$ of the ImageNet-1K dataset and report the top-1 accuracy, following the 1 vs. 1000 setup. Moreover, when involving more advanced LVLMs, we adopt the 1 vs. 20 setup where we employ the top-20 most confusing classes regarding CLIP-LAION400M-ViT/B/32 as the candidate label space. For the re-productivity, we adopt the pre-trained CLIP checkpoints from OpenCLIP (Cherti et al., 2023) and ImageNet-trained model checkpoints from the PyTorch repository. The model naming rule is in Appendix B and the evaluation details are further discussed in Appendix C.

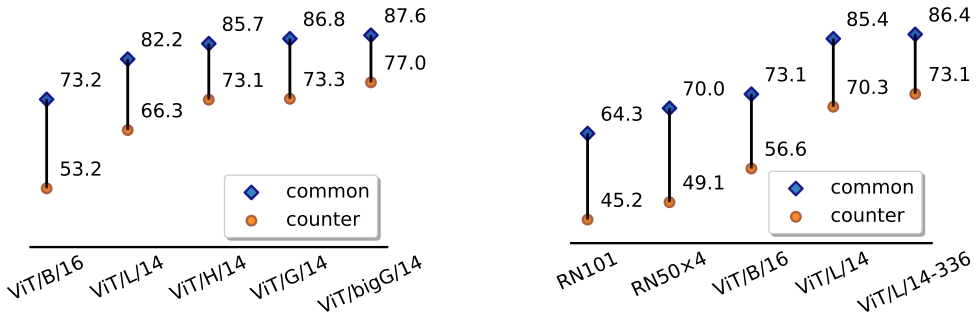
5 Experimental Analysis

Our experiments center on the evaluation and the analysis for CounterAnimal. In Section 5.1, we examine the generality of the captured spurious correlations. In Section 5.2, we explore the potential facets that affect the robustness of CLIPs. In Section 5.3, we extend the evaluation to a broader family of models from different training paradigms.



(a) ViT/B/16 (b) ViT/L/14

Figure 6: Comparison with different pre-train datasets and fixed backbones. The 1 vs. 1000 results are given.



(a) LAION2B (b) OpenAI Checkpoints

Figure 7: Results of CLIPs with different backbones and fixed pre-train datasets. The 1 vs. 1000 results are given.

5.1 Transferability of the Spurious Correlations

In Section 2, we discover spurious correlations using CLIP-LAION400M-ViT/B/32 and collect associated data to build the CounterAnimal dataset. A critical problem then arises: Is our dataset a general benchmark to examine the spurious correlation of CLIPs with other pre-train datasets and backbones? Hence, we examine whether biases in CounterAnimal can hinder the robustness of other CLIPs, considering two cases: a) Fixing the pre-train datasets while varying backbones and b) varying pre-train datasets while fixing the backbones.

Varying Backbones. We fix the pre-train dataset to be LAION400M and explore two other backbones within the ViT family (Dosovitskiy et al., 2020), i.e., ViT/B/16 and ViT/L/14. Their zero-shot results are depicted in Figure 5(a). As we can see, there remains a drop above 17 percentage points for both ViT/B/16 and ViT/L/14. It suggests that the CounterAnimal dataset captures some general spurious shifts that commonly present in LAION400M.

Varying Pre-train Datasets. We fix the backbone to be ViT/B/32 and consider other pre-train datasets. Here, we consider LAION2B (Schuhmann et al., 2022) and the closed-source dataset used by OpenAI (Radford et al., 2021). Their common and counter results are summarized in Figure 5(b). The spurious features degenerate zero-shot robustness of CLIPs trained on both LAION2B and by OpenAI. Therefore, our CounterAnimal dataset possesses some realistic shifts that are generally contained in the large-scale pre-training data, regardless of the backbones. More detailed class-wise results of Figure 5 can be found in Appendix E.

Table 3. The 1 vs. 1000 results for CLIP checkpoints on `CounterAnimal`. The pre-train datasets with high-quality data are marked by *.

backbone	pre-train dataset	common	counter	drop
RN50	OpenAI	64.02	40.70	23.32
RN101	OpenAI	64.27	45.15	19.12
RN50×4	OpenAI	70.02	49.07	20.95
RN50×16	OpenAI	76.43	59.13	17.30
RN50×64	OpenAI	80.25	66.77	13.48
ViT/B/16	LAION400M	73.11	52.17	20.94
ViT/B/16	OpenAI	73.08	56.56	16.52
ViT/B/16	DataComp1B*	80.36	64.24	16.12
ViT/B/16	LAION2B	73.18	53.18	20.00
ViT/B/16	DFN2B*	85.03	70.61	14.42
ViT/B/32	LAION400M	67.13	36.95	30.18
ViT/B/32	OpenAI	69.13	45.62	23.51
ViT/B/32	DataComp1B*	75.96	53.74	22.22
ViT/B/32	LAION2B	72.94	48.74	24.20
ViT/B/32-256	DataComp1B*	80.72	61.65	19.07
ViT/L/14	LAION400M	80.90	63.31	17.59
ViT/L/14	OpenAI	85.38	70.28	15.10
ViT/L/14	DataComp1B*	89.29	79.90	9.39
ViT/L/14	LAION2B	82.23	66.27	15.96
ViT/L/14	DFN2B*	90.77	80.55	10.22
ViT/L/14-336	OpenAI	86.36	73.14	13.21
ViT/H/14	LAION2B	85.74	73.13	12.61
ViT/H/14	DFN5B*	88.55	79.13	9.42
ViT/H/14-384	DFN5B*	90.23	83.67	6.56
ViT/G/14	LAION2B	86.81	73.32	13.49
ViT/bigG/14	LAION2B	87.57	76.96	10.61
ConvNext/B	LAION400M	59.85	36.77	23.08
ConvNext/BW	LAION2B	61.03	39.91	21.12

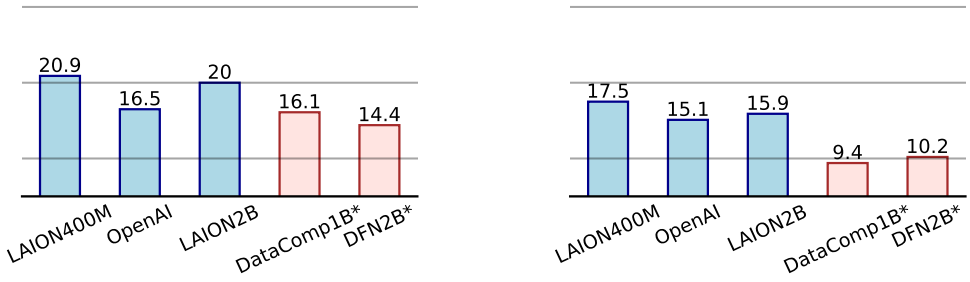
5.2 Scaling up May Relieve Spurious Correlations

We extend our evaluations to a wider range of CLIPs with different scales of parameters and pre-trained data. The main results are summarized in Table 3 and further depicted in Figure 2(a). Generally speaking, performance drops can be observed across all considered CLIP configurations, indicating that CLIPs in various scales still learn spurious features. More specifically, we investigate the influence of a) parameter and b) pre-training data scales in CLIPs to the sensitivity of spurious features. We exclude `ViT/B/32` and `LAION400M` in order to avoid the bias from data collection.

Scaling up Pre-training Data. To test the impacts of enlarging scales of pre-training datasets, we consider two CLIP backbones, namely, `ViT/B/16` and `ViT/L/14`, along with a series of pre-training datasets of increasing sizes. The results are provided in Figure 6. We observe that scaling up the data scale does not necessarily reduce the performance drop, suggesting that directly enlarging the scale of pre-training data alone cannot enhance robustness. One possible explanation is that larger datasets do not imply less biases, whereas CLIPs will learn the spurious correlations.

Scaling up CLIP Model Sizes. Similarly, we explore the connection between model scales and spurious correlations. In Figure 7, we consider two pre-train datasets, namely, `LAION2B` and the close-soured dataset from OpenAI, and the associated backbones in increasing scales. We observe a clear trend indicating that larger models exhibit more robustness against spurious correlations. It may tell us that larger models possess stronger generalization, making them less prone to the shortcuts of spurious features.

Data Quality Matters. Moreover, we observe that results obtained with DataComp and DFN trained CLIPs (marked by *) exhibit better performance and smaller drops across backbones, Figure 8 offers their comparisons. We notice



(a) ViT/B/16 (b) ViT/L/14

Figure 8. Comparisons of CLIPs pre-trained on unfiltered data and on high-quality data (marked by *). The 1 vs. 1000 results are given.

that these datasets have been stringently filtered and thus possess high-quality data. It may indicate that enhancing the quality of pre-train datasets can still be a promising way to improve robustness.

5.3 Evaluations for other Learning Paradigms

We extend our evaluations to broader families of models, including ImageNet-1K supervised models and more advanced LVLMs, i.e., MiniGPT4 and LLaVA.

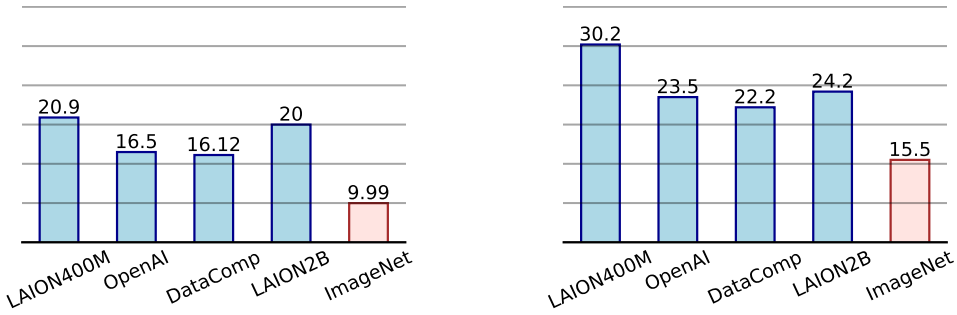
ImageNet Models. We first extend our evaluations to include ImageNet-trained models. The results are summarized in Table 4. Figure 9 illustrates the accuracy drops of various CLIP models, in comparison to ImageNet-trained models. Surprisingly, we find that ImageNet models are more robust to spurious features in the CounterAnimal dataset. This finding may contradict the common belief (Radford et al., 2021; Shi et al., 2023) that the CLIP models tend to be more robust to spurious correlations than single-modal supervised learning.

We conjecture that our CounterAnimal dataset represents the biases in real-world multi-modal training scenarios. However, such biases may not be so influential to ImageNet models. It indicates that spurious correlations in large-scale multimodal data are distinct from that of the ImageNet scenarios which are widely used in conventional single-modal supervised learning. These findings highlight the importance of our proposed dataset, which is especially suitable for studying spurious correlations in large-scale vision-language pre-training.

Table 4. The 1 vs. 1000 performance of the CounterAnimal dataset for ImageNet-trained models.

backbone	common	counter	drop
AlexNet	59.56	39.24	20.31
VGG11	73.37	56.12	17.25
VGG13	75.33	58.43	16.90
VGG19	77.84	61.74	16.10
RN18	74.36	56.07	18.29
RN34	78.31	61.01	17.30
RN50	81.44	66.07	15.37
RN101	81.76	68.18	13.57
ViT/B/16	84.97	74.98	9.99
ViT/B/32	79.84	64.36	15.48
ViT/L/16	83.74	72.69	11.05
ViT/L/32	81.23	67.54	13.69
ConvNext/S	88.27	79.97	8.30
ConvNext/B	88.60	80.53	8.07
ConvNext/L	89.12	81.47	7.65

Advanced LVLMs. We further evaluate for more advanced LVLMs, which align CLIP visual encoders with advanced large language models like Vicuna (Zheng et al., 2023). To reduce inference costs, our evaluation follows the 1 vs. 20 setup. We summarize their results in Table 5, along with the 1 vs. 20 results for several CLIP models (cf., Appendix E for more results). We further depict the full results in Figure 2(b). As we can see, these advanced LVLMs have lower performance yet smaller drops, but the spurious features in CounterAnimal still impact them.



(a) `ViT/B/16` (b) `ViT/B/32`

Figure 9. Comparison between CLIPs and ImageNet models. We consider two backbones: `ViT/B/16` and `ViT/B/32`. The horizontal axis denotes pre-train or supervised datasets. The vertical axis denotes the accuracy drop. The 1 vs. 1000 results are given.

6 Understanding Why CLIPs Rely on Spurious Features

To better understand the observed phenomena, we present a theoretical analysis of why CLIPs rely on spurious features. We begin by establishing the setup for analyzing multimodal contrastive learning following (Xue et al., 2023).

Definition 6.1 (Multimodal Dataset)**.** *Consider n image-text pairs $\{(\boldsymbol{x}_I^i, \boldsymbol{x}_T^i)\}_{i=1}^n$, both image $\boldsymbol{x}_I^i$ and text $\boldsymbol{x}_T^i$ are generated from the latent factor $\boldsymbol{z}_i$, where $\boldsymbol{z} = [z_{inv}, z_{spu}] \in \mathbb{R}^2$ is composed of an invariant feature $z_{inv} \sim \mathcal{N}(\mu_{inv} y, \sigma_{inv}^2)$ and a spurious feature $z_{spu} \sim \mathcal{N}(\mu_{spu} a, \sigma_{spu}^2)$ with $\Pr(a = y) = p_{spu}$ otherwise $a = -y$. y is the label uniformly drawn from $\{-1, 1\}$. The training data $\mathcal{D}^{tr}$ is drawn with $\frac{1}{2} \leq p_{spu} \leq 1$ and OOD data $\mathcal{D}^*$ is drawn with a $p_{spu} = \frac{1}{2}$.*

We employ two linear encoders: $g_I : \mathbb{R}^{d_I} \to \mathbb{R}^h$ for the image modality and $g_T : \mathbb{R}^{d_T} \to \mathbb{R}^h$ for the text modality, implemented as $g_I(\boldsymbol{x}_I) = \boldsymbol{W}_I \boldsymbol{x}_I$ and $g_T(\boldsymbol{x}_T) = \boldsymbol{W}_T \boldsymbol{x}_T$ with $\boldsymbol{W}_I \in \mathbb{R}^{h \times d_I}$ and $\boldsymbol{W}_T \in \mathbb{R}^{h \times d_T}$. The encoders are trained through the linearized contrastive loss (Nakada et al., 2023; Xue et al., 2023) that mimics the CLIP dynamics:

$$\begin{aligned} \mathcal{L}_{\text{CLIP}} = & \frac{1}{2n(n-1)} \sum_i \sum_{j \neq i} (s_{ij} - s_{ii}) \\ & + \frac{1}{2n(n-1)} \sum_i \sum_{j \neq i} (s_{ji} - s_{ii}) + \frac{\rho}{2} ||\boldsymbol{W}_I^T \boldsymbol{W}_T||_F^2, \end{aligned} \tag{1}$$

where $s_{ij} = g_I(\boldsymbol{x}_I^i)^T g_T(\boldsymbol{x}_T^j)$ is the similarity with respect to the i-th image and j-th text representations. Once the CLIP (g_I, g_T) has been trained, the performance will be measured in a zero-shot manner by matching the most similar caption with the corresponding object name filled in, such as "`a photo of <object label>`" (Radford et al., 2021). Intuitively, once the model focuses more on invariant features, it will have a better zero-shot classification accuracy across different distributions. Nevertheless, in the following theorem, we justify that CLIP remains learning to use spurious features, aligned with previous observations.

Theorem 6.2. *Given a multimodal dataset (Def. 6.1) with suitable variance in the features $\sigma_{inv} = \Theta(1) > \sigma_{spu}$, and spurious features with a large spurious correlation $p_{spu} = 1 - o(1)$, an overparameterized CLIP model where $n = \omega(1), d_M = \Omega(n)$ and $d_T = \Omega(n)$, if the spurious features (e.g., backgrounds of the image) takes up a relatively large amount of the image $\mu_{spu} \geq \frac{\sigma_{inv}^2 + 2}{2} \geq \mu_{inv} = 1$, then with a high probability of at least $1 - O(\frac{1}{poly(n)}) = 1 - o(1)$, the CLIP model achieves a large error in zero-shot accuracy in the OOD test data where $a \neq y$:*

$$Err(g_I, g_T) \geq 1 - \Phi(\kappa_1) - o(1),$$

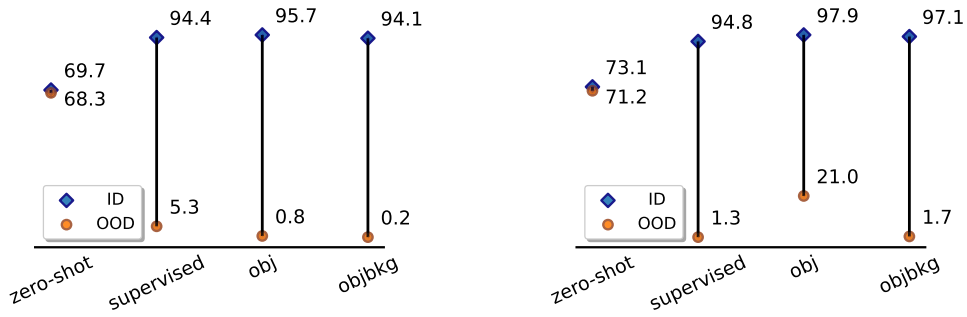
and a small error in the OOD test data where $a = y$:

$$Acc(g_I, g_T) \geq 1 - \Phi(\kappa_2) - o(1),$$

t

Table 5. The 1 vs. 20 performance of `CounterAnimal` for advanced LVLMs and several CLIPs. More results of CLIPs and ImageNet models under the 1 vs. 20 setups are in Appendix E.

LVLMs	common	counter	drop
MiniGPT4-Viccuna7B	47.99	39.73	8.26
LLaVA1.5-7B	40.06	30.09	9.97
CLIP-LAION400M-ViT/L/14	80.90	63.31	17.59
CLIP-OpenAI-ViT/L/14	85.38	70.28	15.10
CLIP-DataComp1B-ViT/L/14	89.29	79.90	9.39
CLIP-LAION2B-ViT/L/14	82.23	66.27	15.96
CLIP-DFN2B-ViT/L/14	90.77	80.55	10.22



(a) `OpenAI-RN50` (b) `OpenAI-ViT/B/16`

Figure 10: CLIP performance on ColoredCOCO dataset.

where $\kappa_1 = \frac{\sigma_{inv}^2+2-2\mu_{spu}p_{spu}}{\sqrt{(1+\sigma_{inv}^2)^2\sigma_{inv}^2+(2\mu_{spu}p_{spu}-1)^2\sigma_{spu}^2}}$, $\kappa_2 = \frac{-2\mu_{spu}p_{spu}-\sigma_{inv}^2}{\sqrt{(1+\sigma_{inv}^2)^2\sigma_{inv}^2+(2\mu_{spu}p_{spu}-1)^2\sigma_{spu}^2}}$ *and* Φ *denotes the CDF of a standard normal distribution.*

We leave more theoretical details as well as the proof to Appendix D due to space limit. Intuitively, Theorem 6.2 implies that, once there exists a relatively strong correlation between the object captions and the parts of image backgrounds, CLIP will learn to align the backgrounds, i.e., spurious features, with object captions. Although our theory discusses a simplistic case of one invariant and one spurious feature, there could exist more features describing the objects and even more features describing the backgrounds. CLIP will fail to robustly align the visual features of objects to its captions, once there exists a spurious correlation between any of the background features with the object caption.

To verify our theory, we construct multimodal datasets named ColoredCOCO following (Ahmed et al., 2021). It contains 9 classes and the spurious correlation in the training part is 80%, i.e., each class has a correlation of 80% to a specific biased color and 20% uniformly correlates to 10 different randomly chosen colors, cf., Figure 11. The OOD datasets are built with classes randomly correlating to other 8 biased colors. We consider two prompts with different descriptiveness: a) `obj`: "`a photo of <object label>`" and b) `objbkg`: "`a photo of <object label> in <color label> background`", with either objects or both objects and backgrounds.

We tune the pre-trained CLIP models using the CLIP objective, which has been shown to be most robust to distribution shifts (Goyal et al., 2023). In addition, we also incorporate the baseline of full fine-tuning with a new MLP onto the image encoder using ERM objective. As shown in Figure 10, fine-tuning with CLIP objective based on neither of the prompts provides any non-trivial robustness against the vanilla full fine-tuning. The results further verify our theory. Nevertheless, the degraded robustness of CLIPs could also be caused by the weak language understanding capability of the BERT encoder in the CLIPs. To this end, we also conduct additional experiments with a perfect language encoder setting. The results are given in Appendix D.4. Nevertheless, we find that CLIPs still perform similarly to ERM and are prone to distribution shifts even with perfect captions.

7 Conclusion

This paper investigates the robustness of LVLMs to spurious correlations, with a focus on CLIPs. Since previous evaluation benchmarks of CLIPs are mainly for ImageNet training distribution, we introduce a new dataset `CounterAnimal` that characterizes the natural spurious correlations between animals and backgrounds. We then evaluate different CLIPs upon `CounterAnimal`. Our results suggest a surprising finding that CLIPs still depend on spurious features for zero-shot predictions, of which the robustness is even lower than conventional ImageNet models. It challenges the prevailing belief for the strong robustness of CLIPs, calling for a sober look of LVLMs. Moreover, we observe that one can enhance CLIP robustness by, e.g., increasing backbone scales and improving pre-train data quality. We also present a theoretical analysis for reasons of CLIPs that learn bias. Overall, we hope our work can inspire subsequent studies for the robustness, enhancing the reliability of LVLMs in the open world.

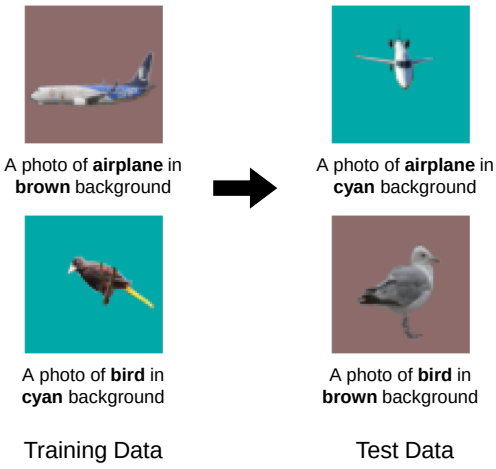


Figure 11. Illustrative examples in COLOREDCOCO dataset.

Impact Statements

For the research community, we challenge the prevailing opinions within the community regarding the robustness of CLIPs against spurious correlations. Our comprehensive analysis suggests that CLIPs might not exhibit better robustness over conventional training paradigms. Therefore, we argue that the current community may overestimate the robustness of CLIPs, and subsequent works will be necessary to address this critical issue. Moreover, our collected dataset, specifically crafted for CLIPs, can serve as a real-world benchmark to evaluate the robustness of CLIPs. It is poised to be meaningful for the subsequent works to understand and enhance CLIPs. For the real-world applications, the understanding of spurious correlations for CLIPs is also critical. We raise the practical concerns when deploying CLIPs, which pertain to fairness and potential biases that may arise from inherent spurious collations. We also present general strategies and theoretical analysis to understanding spurious correlations in CLIPs, which may motivate subsequent works to further enhance CLIPs in real-world applications.

References

Ahmed, F., Bengio, Y., van Seijen, H., and Courville, A. Systematic generalisation with group invariant predictions. In *ICLR*, 2021. (Cited on pages 11 and 19)

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022. (Cited on page 1)

Chen, J., Zhu, D., Shen, X., Li, X., Liu, Z., Zhang, P., Krishnamoorthi, R., Chandra, V., Xiong, Y., and Elhoseiny, M. Minigpt-v2: large language model as a unified interface for vision-language multi-task learning. *arXiv preprint arXiv:2310.09478*, 2023. (Cited on pages 4 and 16)

Cherti, M., Beaumont, R., Wightman, R., Wortsman, M., Ilharco, G., Gordon, C., Schuhmann, C., Schmidt, L., and Jitsev, J. Reproducible scaling laws for contrastive language-image learning. In *CVPR*, 2023. (Cited on page 6)

Daunhawer, I., Bizeul, A., Palumbo, E., Marx, A., and Vogt, J. E. Identifiability results for multimodal contrastive learning. In *ICLR*, 2023. (Cited on pages 1 and 4)

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. (Cited on pages 1 and 2)

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. (Cited on pages 2 and 7)

Fang, A., Ilharco, G., Wortsman, M., Wan, Y., Shankar, V., Dave, A., and Schmidt, L. Data determines distributional robustness in contrastive language image pre-training (CLIP). In *ICML*, 2022. (Cited on pages 1 and 3)

Fang, A., Jose, A. M., Jain, A., Schmidt, L., Toshev, A., and Shankar, V. Data filtering networks. *arXiv preprint arXiv:2309.17425*, 2023. (Cited on page 3)

Gadre, S. Y., Ilharco, G., Fang, A., Hayase, J., Smyrnis, G., Nguyen, T., Marten, R., Wortsman, M., Ghosh, D., Zhang, J., et al. Datacomp: In search of the next generation of multimodal datasets. *arXiv preprint arXiv:2304.14108*, 2023. (Cited on pages 1 and 3)

Goyal, S., Kumar, A., Garg, S., Kolter, Z., and Raghunathan, A. Finetune like you pretrain: Improved finetuning of zero-shot vision models. In *CVPR*, 2023. (Cited on page 11)

Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, Q., Sung, Y.-H., Li, Z., and Duerig, T. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. (Cited on page 1)

Li, J., Li, D., Xiong, C., and Hoi, S. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. (Cited on page 1)

Li, Y., Du, Y., Zhou, K., Wang, J., Zhao, W. X., and Wen, J. Evaluating object hallucination in large vision-language models. *arXiv preprint*, arXiv:2305.10355, 2023. (Cited on pages 4 and 17)

Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. In *NeurIPS*, 2023. (Cited on pages 2, 4 and 16)

Mayilvahanan, P., Wiedemer, T., Rusak, E., Bethge, M., and Brendel, W. Does CLIP's generalization performance mainly stem from high train-test similarity? In *NeurIPS 2023 Workshop on Distribution Shifts: New Frontiers with Foundation Models*, 2023. (Cited on pages 1, 2, 3 and 4)

Minderer, M., Gritsenko, A. A., Stone, A., Neumann, M., Weissenborn, D., Dosovitskiy, A., Mahendran, A., Arnab, A., Dehghani, M., Shen, Z., Wang, X., Zhai, X., Kipf, T., and Houlsby, N. Simple open-vocabulary object detection. In *ECCV*, 2022. (Cited on page 4)

Nakada, R., Gulluk, H. I., Deng, Z., Ji, W., Zou, J., and Zhang, L. Understanding multimodal contrastive learning and incorporating unpaired data. In *AISTAT*, 2023. (Cited on pages 10 and 17)

Oquab, M., Darcet, T., Moutakanni, T., Vo, H. V., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Howes, R., Huang, P.-Y., Xu, H., Sharma, V., Li, S.-W., Galuba, W., Rabbat, M., Assran, M., Ballas, N., Synnaeve, G., Misra, I., Jegou, H., Mairal, J., Labatut, P., Joulin, A., and Bojanowski, P. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. (Cited on page 4)

Pham, H., Dai, Z., Ghiasi, G., Kawaguchi, K., Liu, H., Yu, A. W., Yu, J., Chen, Y.-T., Luong, M.-T., Wu, Y., et al. Combined scaling for zero-shot transfer learning. *Neurocomputing*, 555:126658, 2023. (Cited on page 1)

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. (Cited on pages 1, 2, 3, 7, 9, 10, 17 and 22)

Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. Distributionally robust neural networks. In *ICLR*, 2020a. (Cited on page 5)

Sagawa, S., Raghunathan, A., Koh, P. W., and Liang, P. An investigation of why overparameterization exacerbates spurious correlations. In *ICML*, 2020b. (Cited on page 17)

Santurkar, S., Dubois, Y., Taori, R., Liang, P., and Hashimoto, T. Is a caption worth a thousand images? a study on representation learning. In *ICLR*, 2023. (Cited on pages 1 and 4)

Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al. Laion-5b: An open large-scale dataset for training next generation image-text models. In *NeurIPS*, 2022. (Cited on pages 1, 2 and 7)

Shi, Z., Carlini, N., Balashankar, A., Schmidt, L., Hsieh, C.-J., Beutel, A., and Qin, Y. Effective robustness against natural distribution shifts for models with different training data. In *NeurIPS*, 2023. (Cited on pages 1, 2, 3 and 9)

Tong, S., Jones, E., and Steinhardt, J. Mass-producing failures of multimodal systems with language models. *arXiv preprint arXiv:2306.12105*, 2023. (Cited on page 4)

Tong, S., Liu, Z., Zhai, Y., Ma, Y., LeCun, Y., and Xie, S. Eyes wide shut? exploring the visual shortcomings of multimodal llms. *arXiv preprint arXiv:2401.06209*, 2024. (Cited on page 4)

Wortsman, M., Ilharco, G., Kim, J. W., Li, M., Kornblith, S., Roelofs, R., Lopes, R. G., Hajishirzi, H., Farhadi, A., Namkoong, H., et al. Robust fine-tuning of zero-shot models. In *CVPR*, 2022. (Cited on pages 1 and 2)

Xue, Y., Joshi, S., Nguyen, D., and Mirzasoleiman, B. Understanding the robustness of multi-modal contrastive learning to distribution shift. *arXiv preprint arXiv:2310.04971*, 2023. (Cited on pages 1, 4, 10, 17 and 18)

Yang, Y., Nushi, B., Palangi, H., and Mirzasoleiman, B. Mitigating spurious correlations in multi-modal models during fine-tuning. *arXiv preprint arXiv:2304.03916*, 2023. (Cited on page 4)

Zhang, Q., Wang, Y., and Wang, Y. On the generalization of multi-modal contrastive learning. In *ICML*, 2023. (Cited on page 4)

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*, 2023. (Cited on page 9)

Zhu, D., Chen, J., Shen, X., Li, X., and Elhoseiny, M. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. (Cited on pages 2, 4 and 16)

A Dataset Composition

We release our dataset `CounterAnimal` structured as follows:

```
CounterAnimal
├── ostrich
│   ├── common-ground
│   │   ├── figure1.jpeg
│   │   ├── figure2.jpg
│   │   └── ...
│   └── counter-water
│       ├── figure1.jpeg
│       ├── figure2.jpg
│       └── ...
├── brambling
│   ├── common-green
│   │   ├── figure1.jpeg
│   │   ├── figure2.jpg
│   │   └── ...
│   └── counter-sky
│       ├── figure1.jpeg
│       ├── figure2.jpg
│       └── ...
└── ...
```

Overall, the `CounterAnimal` dataset is organized by the object names. The data therein are further separated into two parts, i.e., the `common` and `counter` groups, where the background name is also provided for each sub-directory. By evaluating accuracy with respect to the `common` and `counter` groups, one can quantify the impacts of spurious correlations.

B CLIP Naming Rules

For the CLIP checkpoints, we adopt the naming rule of "`CLIP-<dataset>-<backbone>`", where `<dataset>` is the name of pre-train datasets and `<backbone>` is the name of backbone models. For example, `CLIP-LAION400M-ViT/B/32` indicates the `ViT/B/32` model CLIP-trained on `LAION400M`. Different training setups are considered in OpenCLIP, and the version of the adopted checkpoints are summarized in Table 6. Moreover, in Table 9, we consider results of checkpoints beyond the adopted ones.

C Experimental Configurations

In this section, we provide more details about our experimental configurations.

C.1 Candidate Label Space

We consider two different label space of candidate labels: a) Using the full ImageNet-1K class names and b) using the top-20 most confusing classes for more computing-intensive models like MiniGPT4. It leads the following two evaluation setups, i.e., the 1 vs. 1000 setup and the 1 vs. 20 setup.

1 vs. 1000 Setup. As a default option, we use the full label space of the ImageNet-1K dataset, which is suitable given that object labels for `CounterAnimal` all belong to that of the ImageNet-1K dataset. Furthermore, this choice also reflects a more realistic situation in the open world, where we have a vast number of candidate labels and the failure cases of LVLMs are common.

1 vs. 20 Setup. To suit more advanced LVLMs of which the inference costs are much higher than CLIPs, we constrain the sizes of candidate label space for each class. Specifically, based on `CLIP-LAION400M-ViT/B/32`, we select the top-20 most confusing labels, which is calculated by the average cosine similarity for both the `common` and `counter` groups, i.e., $\mathcal{L}^{20}_{\texttt{label}} = \texttt{top20}_{\texttt{label}\in\mathcal{L}} \sum_{i\in\mathcal{I}_{\texttt{label}}} \texttt{CS}(\boldsymbol{x}_T^{\texttt{label}}, x)$. Then, $\mathcal{L}^{20}_{\texttt{label}}$ is the candidate label space for data belonging to `label`.

Table 6: Adopted versions of CLIP checkpoints employed in our main experiments.

backbone	pre-train dataset	checkpoint
ViT/B/16	LAION400M	E31
ViT/B/16	LAION2B	S34B B88K
ViT/B/16	DataComp1B	XL S13B B90K
ViT/B/32	LAION400M	E31
ViT/B/32	LAION2B	S34B B79K
ViT/B/32	DataComp1B	XL S13B B90K
ViT/L/14	LAION400M	E31
ViT/L/14	LAION2B	S32B B82K
ViT/L/14	DataComp1B	XL S32B B82K
ViT/H/14	LAION2B	S32B B79K
ViT/G/14	LAION2B	S34B B88K
ViT/bigG/14	LAION2B	S34B B160K
ConvNext/B	LAION400M	S13B B51K
ConvNext/BW	LAION2B	S13B B82K

C.2 Evaluation Metrics

Now, we discuss the evaluation metrics. They are applied to the `common` and `counter` groups separately when testing robustness.

Class-wise Accuracy. We are interested in the effects of spurious features for each class. Therefore, we calculate the prediction accuracy specifically for photos within each class. It can be referred to as the class-wise accuracy, which is given by

$$\texttt{ACC}(\texttt{label}) = \frac{1}{|\mathcal{I}_{\texttt{label}}|} \sum_{i \in \mathcal{I}_{\texttt{label}}} \mathbf{1}\{\hat{y}_i = \texttt{label}\},$$

where $\mathcal{I}_{\texttt{label}}$ is the indices of photos belonging to `label`. The class-wise accuracy reflects the class-level model reliability against spurious correlations.

Average Accuracy. Upon the class-wise accuracy, we can calculate the average performance of models, i.e.,

$$\texttt{ACC} = \frac{1}{|\mathcal{L}|} \sum_{\texttt{label} \in \mathcal{L}} \texttt{ACC}(\texttt{label}).$$

Compared to the conventional average accuracy, i.e., $\frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \mathbf{1}\{\hat{y}_i = \texttt{gt}\}$ with $\mathcal{I}$ the image indices and `gt` the true labels, our definition of the average accuracy further offsets the impact of class imbalance. We default to using the average accuracy, and further present the results without class balancing in Tables 11-12 for CLIPs and ImageNet-trained models.

Accuracy Drop. To quantify the spurious correlations that make CLIPs fail, we measure the performance drop when moving from the `common` to `counter` groups. At the class level, the accuracy drop is defined as the class-wise accuracy of `common` minuses that of `counter`. At the dataset level, it is the average value for the class-level accuracy drop.

C.3 Evaluation details of MiniGPT-4 and LLaVA-1.5

To evaluate LVLMs with a backend of large language models, we follow the common practice in the literature to construct questions and prompt the LVLMs (Zhu et al., 2023; Liu et al., 2023). Specifically, we construct the question:

```
What is the main object in the image?
```

and then calculate the language modeling loss with respect to the answer:

```
A <object name>
```

for each ImageNet class name. Meanwhile, we also tried another question prompt that is widely used in training MiniGPT-4 and LLaVA (Chen et al., 2023; Liu et al., 2023):

```
Describe this image in detail.
```

while the performance will generically decrease.

In addition, when we switch to the object-centric evaluation protocol as (Li et al., 2023):

```
Is there a <object name> in the image?
```

or

```
Is this image a photo of <object name>?
```

and evaluate the answer with `Yes` for each class, we observe a severe performance decrease as the LVLMs easily hallucinate the objects. If we strictly follow the evaluation metrics of (Li et al., 2023) by simply fetching the answers instead of comparing the losses, there exist lots of hallucinated objects by LVLMs in our dataset.

D Theoretical Understanding of CLIP's Robustness to Spurious Features

We provide a more detailed setup and analysis in complementary to Section 6.

D.1 Detailed Theoretical Setup

We begin by introducing more details about the data generation process following the literature (Sagawa et al., 2020b; Nakada et al., 2023; Xue et al., 2023).

Definition D.1 (Multimodal Dataset)**.** *Consider n image-text pairs $\{(\boldsymbol{x}_I^i, \boldsymbol{x}_T^i)\}_{i=1}^n$, both image $\boldsymbol{x}_I^i \in \mathbb{R}^{d_I}$ and text $\boldsymbol{x}_T^i \in \mathbb{R}^{d_T}$ are generated from the underlying latent factor $\boldsymbol{z}_i \in \mathbb{R}^l$. The samples are generated as follows:*

- *$\boldsymbol{z} = [z_{inv}, z_{spu}] \in \mathbb{R}^2$ is composed of a invariant feature $z_{inv} \sim \mathcal{N}(\mu_{inv} y, \sigma_{inv}^2)$ and a spurious feature $z_{spu} \sim \mathcal{N}(\mu_{spu} a, \sigma_{spu}^2)$ with $\Pr(a = y) = p_{spu}$ otherwise $a = -y$, y is the label uniformly drawn from $\{-1, 1\}$, $\mathcal{D}^{tr}$ is drawn with $1/2 \leq p_{spu} \leq 1$ while the OOD test data $\mathcal{D}^*$ is drawn uniformly with $p_{spu} = 1/2$.*
- *Given $\boldsymbol{z}$, the $\boldsymbol{x}$ at modality M is generated via $\boldsymbol{x}_M = \boldsymbol{D}_M \boldsymbol{\mu}_{\boldsymbol{M}}(\boldsymbol{z}) + \xi_M$, with $\boldsymbol{D}_M \in \mathbb{R}^{d_M \times l}$ and $\xi_M \sim \mathcal{N}(0, \sigma_\xi^2 / d_m \boldsymbol{I}_{d_m})$. The matrix $\boldsymbol{D}_M \in \mathbb{R}^{d_m \times l}$ with $d_m > l$ is a matrix with orthonormal columns which can be considered as a dictionary matrix.*

With the definition, we can write every $\boldsymbol{z}^i = \begin{bmatrix} y^i + \eta_{1,i} \\ \mu_{\text{spu}} p_{\text{spu}} + \eta_{2,i} \end{bmatrix}$ where $\eta_{1,i}, \eta_{2,i}$ are two Gaussian variables in the definition.

CLIP Training. We employ two linear encoders $g_I : \mathbb{R}^{d_I} \to \mathbb{R}^h$ for the image modality and $g_T : \mathbb{R}^{d_T} \to \mathbb{R}^h$ for the text modality, implemented as $g^I(\boldsymbol{x}_I) = \boldsymbol{W}_I \boldsymbol{x}_I$ and $g_T(\boldsymbol{x}_T) = \boldsymbol{W}_T \boldsymbol{x}_T$ with $\boldsymbol{W}_I \in \mathbb{R}^{h \times d_I}$ and $\boldsymbol{W}_T \in \mathbb{R}^{h \times d_T}$, respectively. The encoders are trained through the linearized contrastive loss (Nakada et al., 2023; Xue et al., 2023) that mimics CLIP training dynamics:

$$\begin{aligned} \mathcal{L}_{\text{CLIP}} = & \frac{1}{2n(n-1)} \sum_i \sum_{j \neq i} (s_{ij} - s_{ii}) \\ & + \frac{1}{2n(n-1)} \sum_i \sum_{j \neq i} (s_{ji} - s_{ii}) + \frac{\rho}{2} ||\boldsymbol{W}_I^T \boldsymbol{W}_T||_F^2, \end{aligned} \tag{2}$$

where $s_{ij} = g_I(\boldsymbol{x}_I^i)^T g_T(\boldsymbol{x}_T^j)$ is the similarity with respect to the i-th image and j-th text representations, and $||\boldsymbol{W}_I^T \boldsymbol{W}_T||_F^2$ is the a regularization term with $\rho > 0$.

Zero-shot Inference. Once the CLIP model (g_I, g_T) is trained, the performance will be measured in a zero-shot manner by matching the most similar caption such as `'a photo of {object name}'` across different `object name` as class names. Meanwhile, one could also leverage several prompts and leverage the average text embeddings across the available prompts to facilitate the evaluation (Radford et al., 2021). The prompt with respect to y could be modeled as $\boldsymbol{p}_y = \boldsymbol{D}_T \mathbb{E}[\boldsymbol{z}^t | y]$, where $\boldsymbol{D}_T$ is the prompt transformation matrix. Then, the zero-shot accuracy of CLIP could be formalized as follows:

$$\text{Acc}(g_I, g_T) = \mathbb{E}_{(\boldsymbol{x}, y)}[\mathbb{1}(\arg\max_{\hat{y}} g_I(\boldsymbol{x}_I)^T g_T(\boldsymbol{p}_{\hat{y}}), y)], \tag{3}$$

while the error is $\text{Err}(g_I, g_T) = 1 - \text{Acc}(g_I, g_T)$. Intuitively, once the model extracts more of the invariant features, it will have a better zero-shot classification accuracy across different distributions.

D.2 Proof for Theorem 6.2

Theorem D.2 (Restatement of Theorem 6.2)**.** *Given a multimodal dataset (Def. D.1) with suitable variance in the features $\sigma_{inv} = \Theta(1) > \sigma_{spu}$, and spurious features with a large spurious correlation $p_{spu} = 1 - o(1)$, an overparameterized CLIP model where $n = \omega(1), d_M = \Omega(n)$ and $d_T = \Omega(n)$, if the spurious features (e.g., backgrounds of the image) takes up a relatively large amount of the image $\mu_{spu} \geq \frac{\sigma_{inv}^2+2}{2} \geq \mu_{inv} = 1$, then with a high probability of at least $1 - O(\frac{1}{poly(n)}) = 1 - o(1)$, the CLIP model achieves a large error in zero-shot accuracy in the OOD test data where $a \neq y$:*

$$Err(g_I, g_T) \geq 1 - \Phi(\kappa_1) - o(1),$$

and a small error in the OOD test data where $a = y$:

$$Acc(g_I, g_T) \geq 1 - \Phi(\kappa_2) - o(1),$$

where $\kappa_1 = \frac{\sigma_{inv}^2+2-2\mu_{spu}p_{spu}}{\sqrt{(1+\sigma_{inv}^2)^2\sigma_{inv}^2+(2\mu_{spu}p_{spu}-1)^2\sigma_{spu}^2}}$, $\kappa_2 = \frac{-2\mu_{spu}p_{spu}-\sigma_{inv}^2}{\sqrt{(1+\sigma_{inv}^2)^2\sigma_{inv}^2+(2\mu_{spu}p_{spu}-1)^2\sigma_{spu}^2}}$ and Φ denotes the CDF of a standard normal distribution.

Proof. We will introduce some useful lemmas to help with our proof.

Lemma D.3 (Xue et al. (2023))**.** *The minimizer of linearized CLIP loss $\boldsymbol{W}_I^{*T}\boldsymbol{W}_T^*$ satisfies the following with a probability of at least $1 - O(\frac{1}{poly(n)})$ such that,*

$$||\boldsymbol{W}_I^{*T}\boldsymbol{W}_T^* - \frac{1}{\rho}\boldsymbol{D}_I \begin{bmatrix} 1+\sigma_{inv}^2 & 2\mu_{spu}p_{spu}-1 \\ 2\mu_{spu}p_{spu}-1 & 1+\sigma_{spu}^2 \end{bmatrix} \boldsymbol{D}_T^T||_2 \leq \frac{1}{\rho}O(\sqrt{\epsilon_0}),$$

where $\epsilon_0 = O(\sqrt{\frac{\log n}{n}})$.

Intuitively, the lemma indicates the importance of the training distribution, that the minimizer of CLIP will converge to the data characteristics of the latent features of the training distribution.

Then, consider the case where the model is inferred onto a test sample with $y = 1, a = -1$. Then, with the aforementioned lemma, we have

$$\begin{aligned} |\boldsymbol{x}_I^T\boldsymbol{W}_I^*\boldsymbol{W}_T^*\boldsymbol{x}_T^{\hat{y}} - \frac{1}{\rho}\boldsymbol{x}_I^T\boldsymbol{D}_I \begin{bmatrix} 1+\sigma_{\text{inv}}^2 & 2\mu_{\text{spu}}p_{\text{spu}}-1 \\ 2\mu_{\text{spu}}p_{\text{spu}}-1 & 1+\sigma_{\text{spu}}^2 \end{bmatrix} \boldsymbol{D}_T^T\boldsymbol{x}_T^{\hat{y}}||_2 &\leq ||\boldsymbol{x}_I||||\boldsymbol{x}_T^{\hat{y}}||\frac{1}{\rho}O(\sqrt{\epsilon_0}) \\ &\leq \frac{1}{\rho}O(\sqrt{\epsilon_0}\log n). \end{aligned} \tag{4}$$

Then, notice that

$$\frac{1}{\rho}\boldsymbol{x}_I^T\boldsymbol{D}_I \begin{bmatrix} 1+\sigma_{\text{inv}}^2 & 2\mu_{\text{spu}}p_{\text{spu}}-1 \\ 2\mu_{\text{spu}}p_{\text{spu}}-1 & 1+\sigma_{\text{spu}}^2 \end{bmatrix} \boldsymbol{D}_T^T\boldsymbol{x}_T^{\hat{y}} = \hat{y}((1+\eta_1)(1+\sigma_{\text{inv}}^2) + (-1+\eta_2)(2\mu_{\text{spu}}p_{\text{spu}}-1). \tag{5}$$

When CLIP makes an incorrect prediction, we have

$$\boldsymbol{x}_I^T\boldsymbol{W}_I^*\boldsymbol{W}_T^*\boldsymbol{x}_T^{\hat{y}=1} < \boldsymbol{x}_I^T\boldsymbol{W}_I^*\boldsymbol{W}_T^*\boldsymbol{x}_T^{\hat{y}=-1}.$$

Then, we have

$$\begin{aligned} &\frac{1}{\rho}\boldsymbol{x}_I^T\boldsymbol{D}_I \begin{bmatrix} 1+\sigma_{\text{inv}}^2 & 2\mu_{\text{spu}}p_{\text{spu}}-1 \\ 2\mu_{\text{spu}}p_{\text{spu}}-1 & 1+\sigma_{\text{spu}}^2 \end{bmatrix} \boldsymbol{D}_T^T\boldsymbol{x}_T^{\hat{y}=1} - \frac{1}{\rho}O(\sqrt{\epsilon_0}\log n) < \\ &\frac{1}{\rho}\boldsymbol{x}_I^T\boldsymbol{D}_I \begin{bmatrix} 1+\sigma_{\text{inv}}^2 & 2\mu_{\text{spu}}p_{\text{spu}}-1 \\ 2\mu_{\text{spu}}p_{\text{spu}}-1 & 1+\sigma_{\text{spu}}^2 \end{bmatrix} \boldsymbol{D}_T^T\boldsymbol{x}_T^{\hat{y}=-1} - \frac{1}{\rho}O(\sqrt{\epsilon_0}\log n), \end{aligned} \tag{6}$$

with Eq. 5 plugged in, denote $\epsilon_1 = O(\sqrt{\epsilon_0}\log n)$, we further have

$$-2\left[(1+\eta_1)(1+\sigma_{\text{inv}}^2) + (-1+\eta_2)(2\mu_{\text{spu}}p_{\text{spu}}-1) - \epsilon_1\right] > 0. \tag{7}$$

Since $\eta_1(1+\sigma_{\text{inv}}^2) + \eta_2(2\mu_{\text{spu}}p_{\text{spu}}-1)$ is a Gaussian variable follows the distribution of

$$\eta_1(1+\sigma_{\text{inv}}^2) + \eta_2(2\mu_{\text{spu}}p_{\text{spu}}-1) \sim \mathcal{N}(0, (1+\sigma_{\text{inv}}^2)^2\sigma_{\text{inv}}^2 + (2\mu_{\text{spu}}p_{\text{spu}}-1)^2\sigma_{\text{spu}}^2),$$

Table 7: Comparison between CLIPs and standard supervised learning on `ColoredCOO`

backbone	pre-train dataset	approach	in-distribution	out-of-distribution	drop
RN50	OpenAI	zero shot	69.67	68.33	1.34
RN50	OpenAI	obj	95.67	0.78	94.89
RN50	OpenAI	objbkg	94.11	0.22	93.89
RN50	OpenAI	supervised	94.44	5.33	89.11
ViT16	OpenAI	zero shot	73.11	71.22	1.89
ViT16	OpenAI	obj	97.89	21	76.89
ViT16	OpenAI	objbkg	97.11	1.67	95.44
ViT16	OpenAI	supervised	94.78	1.33	93.45

then, we have

$$\begin{aligned}
&\Pr(-2\left[(1+\eta_1)(1+\sigma_{\text{inv}}^2)+(-1+\eta_2)(2\mu_{\text{spu}}p_{\text{spu}}-1)-\epsilon_1\right]>0)\\
&=\Pr_{v\sim\mathcal{N}(0,1)}(v>\frac{\sigma_{\text{inv}}^2+2-2\mu_{\text{spu}}p_{\text{spu}}+\epsilon_1}{\sqrt{(1+\sigma_{\text{inv}}^2)^2\sigma_{\text{inv}}^2+(2\mu_{\text{spu}}p_{\text{spu}}-1)^2\sigma_{\text{spu}}^2}})\\
&=1-\Phi(\frac{\sigma_{\text{inv}}^2+2-2\mu_{\text{spu}}p_{\text{spu}}+\epsilon_1}{\sqrt{(1+\sigma_{\text{inv}}^2)^2\sigma_{\text{inv}}^2+(2\mu_{\text{spu}}p_{\text{spu}}-1)^2\sigma_{\text{spu}}^2}}),
\end{aligned} \tag{8}$$

where Φ is the CDF of the standard Gaussian distribution. Then, it suffices to know that the $\text{Err}_{y=1,a=-1}$ is lower bounded by $\Phi(\frac{\sigma_{\text{inv}}^2+2-2\mu_{\text{spu}}p_{\text{spu}}+\epsilon_1}{\sqrt{(1+\sigma_{\text{inv}}^2)^2\sigma_{\text{inv}}^2+(2\mu_{\text{spu}}p_{\text{spu}}-1)^2\sigma_{\text{spu}}^2}})$, which also applies to the case $y=-1, a=1$.

Similarly, given the case $y=a$, as the model fits the spurious feature, we could derive the lower bound for its Acc by leveraging the spurious features as $\Phi(\frac{-2\mu_{\text{spu}}p_{\text{spu}}-\sigma_{\text{inv}}^2}{\sqrt{(1+\sigma_{\text{inv}}^2)^2\sigma_{\text{inv}}^2+(2\mu_{\text{spu}}p_{\text{spu}}-1)^2\sigma_{\text{spu}}^2}})$. □

D.3 More Details on `ColoredCOO` Experiments

To further validate our theoretical results, we construct the `ColoredCOO` dataset following (Ahmed et al., 2021). More specifically, `ColoredCOO` is constructed as follows:

- The dataset contains 9 classes of COCO objects. The spurious correlation in the trainset is 80% such that each class has a correlation of 80% to a specific biased color and 20% uniformly correlates to 10 sufficiently different randomly chosen colors.
- The OOD test sets are constructed with classes randomly correlating to 8 biased colors different from the one correlated in the training set.

Then, we further generate two prompts for each sample:

1. `obj`: `a photo of <object label>`;
2. `objbkg`: `a photo of <object label> in <color label> background`

We tune the pre-trained CLIP models using the CLIP objective based on the OpenCLIP library. We consider the learning rate of $\{1e-3, 1e-4, 1e-5\}$, with a weight decay of $\{1e-1, 1e-3, 1e-5\}$, and a warmup of $\{0, 1000\}$ steps. We select the model according to the best in-distribution test performance. The detailed results are given in Table 7. As we can see, the CLIPs finetuned using either the CLIP objective or the standard supervised training both exhibit high sensitivity to the spurious features.

D.4 More Details on `MultiColoredMNIST` Experiments

One possible explanation for the failure of CLIP objective in `ColoredCOCO` is that, the language encoder of the CLIP models may not understand the captions well. Therefore, we further construct a new setup called `MultiColoredMNIST`, where each image contains only the digit information from MNIST dataset and the color information. Therefore, we can directly derive the one hot encoding for all of the useful factors in the dataset.

Data. We consider a multi-class classification setting with a number of classes no less than 2. The objects are the

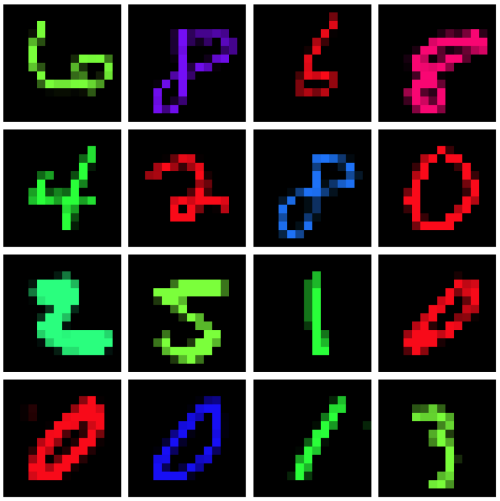


Figure 12: Examples of `MultiColoredMNIST` dataset.

- Training data: Fix two class (0/1) and color (r/g), they are spurious correlated by a correlation p_{spu}. The invariant feature's correlation with labels is p_{inv}.
- Test data (Rand): All classes and the colors are randomly correlated, given k class, $p_{\text{spu}} = 1/k$.
- Test data (Rev): All classes and the colors are reversely correlated, p_{spu} is 10% 0/1 classes and $1/k$ for others.

In Figure 12, we give some examples for the `MultiColoredMnist` dataset.

Experimental setting. We compare the standard supervised training and CLIP. To avoid noises or information loss in encoding language modality, we consider the perfect language supervision for a single model.

Given a batch of image and caption representations $\{(\boldsymbol{h}^{x_i}, \boldsymbol{h}^{c_i})\}_i^B$, for a image-caption pair, the CLIP objective aims to

$$\max(\boldsymbol{M}_x \boldsymbol{h}^{x_i} \cdot \boldsymbol{M}_c \boldsymbol{h}^{c_i}) - (\boldsymbol{M}_x \boldsymbol{h}^{x_j} \cdot \boldsymbol{M}_c \boldsymbol{h}^{c_j}), \forall i \neq j, \tag{9}$$

where $\boldsymbol{M}_x \in \mathbb{R}^{d \times h_x}$ and $\boldsymbol{M}_c^{d \times h_c}$ are the learnable projection layers for image and caption representations. Assuming the perfect language encoding as the one-hot encoding for all possible object and background appearance $\boldsymbol{h}^{c_i} \in [0,1]^{|\mathcal{O}|+|\mathcal{B}|}$, and $\boldsymbol{M}_c$ can simply be an identity matrix, then Eq. 9 can be considered as a classification task for objects and backgrounds respectively:

$$\max \text{CE}(\boldsymbol{M}_c^T \boldsymbol{M}_x \boldsymbol{h}^{x_i}, \boldsymbol{h}^{c_i}), \tag{10}$$

where the labels are simply the one-hot encodings of the objects and the backgrounds, and the classifier is $\boldsymbol{M}_c^T \boldsymbol{M}_x$. For the `MultiColoredMNIST` task where there is only one object and background (i.e., color), to implement Eq. 10, we only need to construct an additional classification head for the background.

Given the aforementioned setup, we conduct experiments comparing CLIP-based contrastive learning to the standard supervised learning. The results are given in Table 8. As we can see, both contrastive learning and supervised learning perform similarly across different numbers of classes and bias degrees.

Table 8: Comparison of standard supervised learning and contrastive learning on `MultiColoredMNIST` dataset.

# classes	# samples	p_{inv}	p_{spu}	train method	class 0/1 (Rand)	class 0/1 (Rev.)	rest class
2	10,610	0.9	0.75	Contrastive	87.42±0.79	81.87±1.86	n/a
2	10,610	0.9	0.75	Supervised	86.44±0.90	80.22±1.73	n/a
2	10,610	0.9	0.9	Contrastive	71.56±1.79	50.08±3.97	n/a
2	10,610	0.9	0.9	Supervised	71.62±1.58	50.13±3.24	n/a
2	10,610	0.75	0.75	Contrastive	65.06±2.21	43.18±3.78	n/a
2	10,610	0.75	0.75	Supervised	65.01±1.68	43.76±3.44	n/a
2	10,610	0.75	0.9	Contrastive	53.73±1.08	16.42±1.74	n/a
2	10,610	0.75	0.9	Supervised	53.89±0.96	17.14±1.88	n/a
3	15,578	0.9	0.75	Contrastive	85.86±0.70	81.88±0.52	88.33±1.48
3	15,578	0.9	0.75	Supervised	85.03±1.25	79.20±1.91	88.03±1.10
3	15,578	0.9	0.9	Contrastive	69.05±2.26	45.55±4.52	88.60±1.20
3	15,578	0.9	0.9	Supervised	68.29±1.37	44.74±3.50	88.43±0.89
3	15,578	0.75	0.75	Contrastive	61.57±2.86	37.76±2.81	68.84±3.53
3	15,578	0.75	0.75	Supervised	59.51±2.28	36.66±2.06	68.75±2.58
3	15,578	0.75	0.9	Contrastive	42.47±2.48	7.08±1.10	71.07±3.01
3	15,578	0.75	0.9	Supervised	41.60±1.67	8.18±0.95	71.89±1.55
5	25,538	0.9	0.75	Contrastive	86.06±0.56	82.41±0.77	88.30±0.39
5	25,538	0.9	0.75	Supervised	85.60±0.74	80.99±0.99	87.76±0.57
5	25,538	0.9	0.9	Contrastive	71.78±0.77	44.66±4.02	88.15±0.42
5	25,538	0.9	0.9	Supervised	70.73±1.41	43.47±4.01	87.80±0.59
5	25,538	0.75	0.75	Contrastive	61.15±1.10	33.97±3.70	71.88±0.79
5	25,538	0.75	0.75	Supervised	57.69±1.29	33.66±3.18	68.75±0.91
5	25,538	0.75	0.9	Contrastive	35.37±1.70	4.60±0.45	72.47±0.58
5	25,538	0.75	0.9	Supervised	34.82±1.97	5.44±0.70	69.38±0.59
6	30,044	0.9	0.75	Contrastive	85.76±0.74	81.87±1.41	86.58±0.54
6	30,044	0.9	0.75	Supervised	85.84±0.81	81.81±1.27	86.29±0.47
6	30,044	0.9	0.9	Contrastive	70.99±2.39	40.07±10.53	86.57±0.49
6	30,044	0.9	0.9	Supervised	70.97±2.45	40.63±9.81	86.25±0.52
6	30,044	0.75	0.75	Contrastive	62.05±1.18	32.70±4.50	70.76±0.40
6	30,044	0.75	0.75	Supervised	59.49±1.26	33.94±3.69	67.91±0.81
6	30,044	0.75	0.9	Contrastive	38.96±2.55	4.71±0.56	70.65±0.40
6	30,044	0.75	0.9	Supervised	35.85±2.27	4.87±0.71	68.36±0.91
8	40,170	0.9	0.75	Contrastive	84.81±0.86	80.54±1.27	86.43±0.40
8	40,170	0.9	0.75	Supervised	85.49±0.67	81.47±1.08	86.78±0.39
8	40,170	0.9	0.9	Contrastive	71.75±1.65	39.85±8.81	86.34±0.36
8	40,170	0.9	0.9	Supervised	72.82±1.37	41.36±7.19	86.78±0.39
8	40,170	0.75	0.75	Contrastive	63.73±1.96	31.46±7.20	71.08±0.57
8	40,170	0.75	0.75	Supervised	62.22±2.00	33.12±6.54	70.58±0.63
8	40,170	0.75	0.9	Contrastive	43.91±2.36	5.11±0.68	70.76±0.60
8	40,170	0.75	0.9	Supervised	40.39±2.82	5.28±0.92	70.43±0.64
10	50,000	0.9	0.75	Contrastive	84.52±0.77	80.42±1.70	85.19±0.27
10	50,000	0.9	0.75	Supervised	85.10±0.67	81.83±0.97	86.11±0.15
10	50,000	0.9	0.9	Contrastive	73.79±1.43	48.02±5.50	85.18±0.34
10	50,000	0.9	0.9	Supervised	74.97±1.69	52.09±5.72	85.96±0.24
10	50,000	0.75	0.75	Contrastive	65.31±1.43	32.31±6.73	69.67±0.53
10	50,000	0.75	0.75	Supervised	66.00±1.52	36.35±5.59	70.27±0.30
10	50,000	0.75	0.9	Contrastive	48.03±1.56	5.53±1.25	69.13±0.47
10	50,000	0.75	0.9	Supervised	46.83±1.33	5.72±1.35	69.92±0.37

Table 9: The 1 vs. 1000 performance with other versions of CLIP checkpoints in OpenCLIP.

backbone	pre-train dataset	checkpoint	common	counter	drop
ViT/B/16	LAION400M	E31	73.11	52.17	20.94
ViT/B/16	LAION400M	E32	73.59	52.53	21.06
ViT/B/16	DataComp1B	XL S13B B90K	80.36	64.24	16.12
ViT/B/16	DataComp1B	L S1B B8K	65.80	44.14	21.66
ViT/B/32	LAION400M	E31	67.13	36.95	30.18
ViT/B/32	LAION400M	E32	67.13	36.98	30.15
ViT/B/32	LAION2B	E16	71.32	47.21	24.11
ViT/B/32	LAION2B	S34B B79K	72.94	48.74	24.20
ViT/B/32	DataComp1B	XL S13B B90K	75.96	53.74	22.22
ViT/B/32	DataComp1B	M S128M B4K	25.91	11.65	14.26
ViT/B/32	DataComp	S S13M B4K	0.02	0.01	0.01
ViT/L/14	LAION400M	E31	80.90	63.31	17.59
ViT/L/14	LAION400M	E32	81.11	63.87	17.24
ViT/G/14	LAION2B	S12B B42K	83.72	68.46	15.26
ViT/G/14	LAION2B	S34B B88K	86.81	73.32	13.49

E More Results

We present more results for the evaluations on CounterAnimal, supplementing our analysis of CLIPs under spurious correlations.

Other Versions of Pre-train Datasets. OpenCLIP provides other CLIP checkpoints beyond our adopted ones. Table 9 summarizes the results of CLIPs similar to Table 3 while using different versions of checkpoints. As we can see, the performance for both common and counter is very stable across varying versions, except for DataComp1B. The reason is that their various checkpoints use subsets of DataComp1B, where XL indicates the fully DataComp1B, L indicates a 140M subset, M indicates a 14M subset, and S indicates a 1.4M subset.

Results of OpenAI Prompts. We further consider the prompt setups following OpenAI CLIP (Radford et al., 2021), using average text embeddings over 80 predefined prompts as the final text embeddings. The results are summarized in Table 10. As we can see, the average performance for both the common and counter groups generally improves 1 to 3 percentage points over the results of our simpler prompt. However, our main conclusion remains unchanged: the ImageNet models generally exhibit better performance and smaller drops. Another interesting finding is that when evaluating with CLIP-LAION400M-ViT/B/32 (the CLIP checkpoint employed in our data collection), the performance drop with OpenAI prompts is not as high as that of our simple prompt used in Table 3. It indicates that our curation procedure mainly overfit the adopted prompt instead of the particular CLIP checkpoint.

Average Performance without Balancing. We by default adopt the balanced average accuracy to offset the impacts of class imbalance. In Tables 11-12, we further summarize the results without class balance, following $\frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}}\mathbf{1}\{\hat{y}_i = \texttt{gt}\}$. As we can see, the performance drop remain obvious and similar conclusions can be drawn as the balanced results: a) Backbone scales are more important for spurious robustness than pre-train dataset scales, and b) ImageNet models are more reliable when facing spurious features in CounterAnimal.

1 vs. 20 Results for CLIPs and ImageNet Models. We adopt the 1 vs. 20 setup for the evaluations of more advanced LVLMs in Table 5. For fair comparison, we further summarize the 1 vs. 20 results for CLIPs in Table 13 and for ImageNet models in Table 14. As we can see, there do not exist a significant change in performance drop compared to 1 vs. 1000 results, indicating that mistakes made by CLIPs are relatively concentrated. As in Figure 2, we also depict the common vs. counter performance for various learning setups with their names, following the 1 vs. 1000 setup in Figure 13 and 1 vs. 20 setups in Figure 14.

Class-wise Results. In Tables 15-16, we summarize the detailed results of the class-wise accuracy for the main results in Figure 5. We further depict the drop in accuracy in Figure 15. Generally speaking, the spurious features found in CLIP-LAION400M-ViT/B/32 can also fail other CLIP setups, and the general trends of decline are preserved class-wise. However, there are some cases where the drop in accuracy between common and counter is negative, e.g., for data in class ID 33 and 42. It means that for these cases, our collection pipeline may have a large overfit to the adopted CLIP setup, i.e., CLIP-LAION400M-ViT/B/32.

Table 10. The 1 vs. 1000 performance using prompts of OpenAI CLIP. The pre-train datasets with high-quality data are marked by *.

backbone	pre-train dataset	common	counter	drop
RN50	OpenAI	64.55	44.20	20.35
RN101	OpenAI	64.81	46.30	18.51
RN50×4	OpenAI	69.62	53.68	15.93
RN50×16	OpenAI	84.78	72.13	12.65
RN50×64	OpenAI	84.33	72.02	12.31
ViT/B/16	LAION400M	76.20	58.17	18.18
ViT/B/16	OpenAI	76.58	60.58	16.00
ViT/B/16	DataComp1B*	82.85	69.74	13.11
ViT/B/16	LAION2B	74.08	58.18	15.90
ViT/B/16	DFN2B*	85.20	74.33	10.87
ViT/B/32	LAION400M	66.68	43.22	23.46
ViT/B/32	OpenAI	67.23	47.11	20.12
ViT/B/32	DataComp1B*	76.00	59.23	16.77
ViT/B/32	LAION2B	70.25	50.00	20.25
ViT/B/32-256	DataComp1B*	79.77	64.20	15.57
ViT/L/14	LAION400M	81.22	65.31	15.91
ViT/L/14	OpenAI	85.76	73.23	12.53
ViT/L/14	DataComp1B*	89.56	81.21	8.35
ViT/L/14	LAION2B	83.43	69.44	13.99
ViT/L/14	DFN2B*	90.45	82.28	8.17
ViT/L/14-336	OpenAI	86.45	76.30	10.15
ViT/H/14	LAION2B	86.11	75.30	10.81
ViT/H/14	DFN5B*	91.33	85.20	6.13
ViT/H/14-384	DFN5B*	92.20	88.01	4.19
ViT/G/14	LAION2B	87.17	77.20	10.97
ViT/bigG/14	LAION2B	87.57	76.96	10.61
ConvNext/B	LAION400M	60.20	44.15	16.05
ConvNext/BW	LAION2B	63.33	46.11	17.22

Table 11. The 1 vs. 1000 performance on CounterAnimal for CLIPs, evaluating based on the accuracy without balancing. The pre-train datasets with high-quality data are marked by *.

backbone	pre-train dataset	common	counter	drop
RN50	OpenAI	64.59	38.40	26.19
RN101	OpenAI	64.18	43.99	20.19
RN50×4	OpenAI	70.76	46.91	23.85
RN50×16	OpenAI	77.26	58.97	18.29
RN50×64	OpenAI	82.88	62.84	20.04
ViT/B/16	LAION400M	75.58	48.46	27.12
ViT/B/16	OpenAI	73.94	53.93	20.01
ViT/B/16	DataComp1B*	81.83	61.47	20.36
ViT/B/16	LAION2B	74.97	51.20	23.77
ViT/B/16	DFN2B*	86.10	67.95	18.14
ViT/B/32	LAION400M	69.02	33.94	35.08
ViT/B/32	OpenAI	68.84	44.17	24.67
ViT/B/32	DataComp1B*	78.16	51.50	26.66
ViT/B/32	LAION2B	74.23	46.36	27.87
ViT/B/32-256	DataComp1B*	82.38	58.56	23.82
ViT/L/14	LAION400M	81.06	61.68	19.38
ViT/L/14	OpenAI	85.29	69.25	16.04
ViT/L/14	DataComp1B*	90.79	77.28	13.51
ViT/L/14	LAION2B	83.47	62.33	21.14
ViT/L/14	DFN2B*	91.81	78.10	13.71
ViT/L/14-336	OpenAI	86.40	72.40	14.00
ViT/H/14	LAION2B	87.10	69.84	17.26
ViT/H/14	DFN5B*	90.36	76.19	14.17
ViT/H/14-384	DFN5B*	92.29	80.95	11.34
ViT/G/14	LAION2B	88.09	69.96	18.13
ViT/bigG/14	LAION2B	88.47	73.45	15.02
ConvNext/B	LAION400M	60.16	34.27	25.89
ConvNext/BW	LAION2B	60.65	38.64	22.01

Table 12. The 1 vs. 1000 performance on `CounterAnimal` for ImageNet models, evaluating based on the accuracy without balancing.

backbone	common	counter	drop
AlexNet	62.33	37.20	25.12
VGG11	75.92	53.35	22.57
VGG13	77.23	55.58	21.65
VGG19	79.40	58.93	20.47
RN18	76.46	52.79	23.67
RN34	80.38	57.80	22.58
RN50	83.52	62.97	20.54
RN101	83.58	64.74	18.84
ViT/B/16	86.97	71.62	15.35
ViT/B/32	82.03	61.71	20.32
ViT/L/16	85.96	70.21	15.75
ViT/L/32	82.89	64.64	18.25
ConvNext/S	89.88	76.61	13.27
ConvNext/B	90.27	77.51	12.76
ConvNext/L	90.67	78.34	12.33

Table 13. The 1 versus 20 performance on `CounterAnimal` for CLIPs. The pre-train datasets with high-quality data are marked by *.

backbone	pre-train dataset	common	counter	drop
RN50	OpenAI	67.41	43.63	23.78
RN101	OpenAI	66.92	47.23	19.69
RN50×4	OpenAI	71.82	50.50	21.32
RN50×16	OpenAI	78.60	60.63	17.97
RN50×64	OpenAI	82.33	69.05	13.28
ViT/B/16	LAION400M	75.51	54.59	20.92
ViT/B/16	OpenAI	75.89	58.74	17.15
ViT/B/16	DataComp1B*	82.02	66.02	16.00
ViT/B/16	LAION2B	75.85	55.48	20.37
ViT/B/16	DFN2B*	86.04	72.13	13.91
ViT/B/32	LAION400M	70.46	39.44	31.02
ViT/B/32	OpenAI	72.17	49.25	22.92
ViT/B/32	DataComp1B*	78.58	56.32	22.26
ViT/B/32	LAION2B	75.68	51.86	23.82
ViT/B/32-256	DataComp1B*	83.05	63.98	19.07
ViT/L/14	LAION400M	82.27	64.89	17.38
ViT/L/14	OpenAI	86.38	72.12	14.26
ViT/L/14	DataComp1B*	90.13	80.46	9.67
ViT/L/14	LAION2B	83.81	67.68	16.13
ViT/L/14	DFN2B*	91.29	81.23	10.05
ViT/L/14-336	OpenAI	87.56	75.16	12.40
ViT/H/14	LAION2B	86.75	74.29	12.46
ViT/H/14	DFN5B*	89.13	79.79	9.35
ViT/H/14-384	DFN5B*	90.70	84.00	6.70
ViT/G/14	LAION2B	87.74	74.11	13.63
ViT/bigG/14	LAION2B	88.35	77.85	10.50
ConvNext/B	LAION400M	64.85	39.71	25.14
ConvNext/BW	LAION2B	65.61	44.21	21.40

Table 14: The 1 versus 20 performance on `CounterAnimal` for ImageNet models.

backbone	common	counter	drop
AlexNet	67.71	46.43	21.29
VGG11	77.25	60.19	17.06
VGG13	79.07	62.02	17.04
VGG19	80.80	65.19	15.61
RN18	78.11	59.47	18.64
RN34	81.14	64.32	16.82
RN50	83.72	68.60	15.29
RN101	84.13	70.77	13.37
ViT/B/16	86.57	76.88	9.69
ViT/B/32	82.56	68.30	14.26
ViT/L/16	85.71	74.94	10.77
ViT/L/32	83.86	71.00	12.86
ConvNext/S	89.31	81.61	7.69
ConvNext/B	89.58	82.32	7.26
ConvNext/L	89.84	82.67	7.17

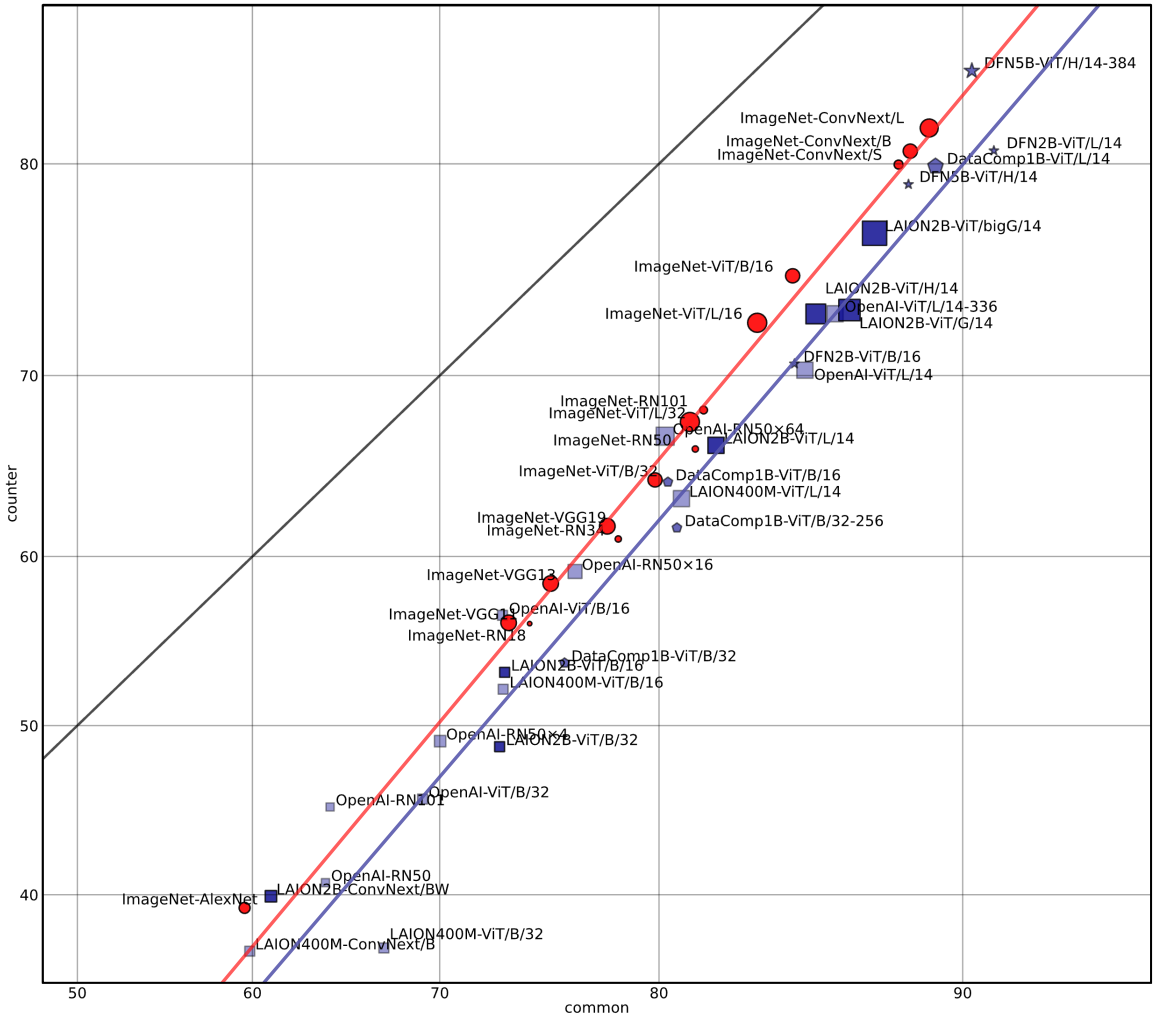


Figure 13. The `common` vs. `counter` performance (%) for CLIPs and ImageNet models, following the 1 vs. 1000 setup. We also present the model setups for each `common-counter` result pair.

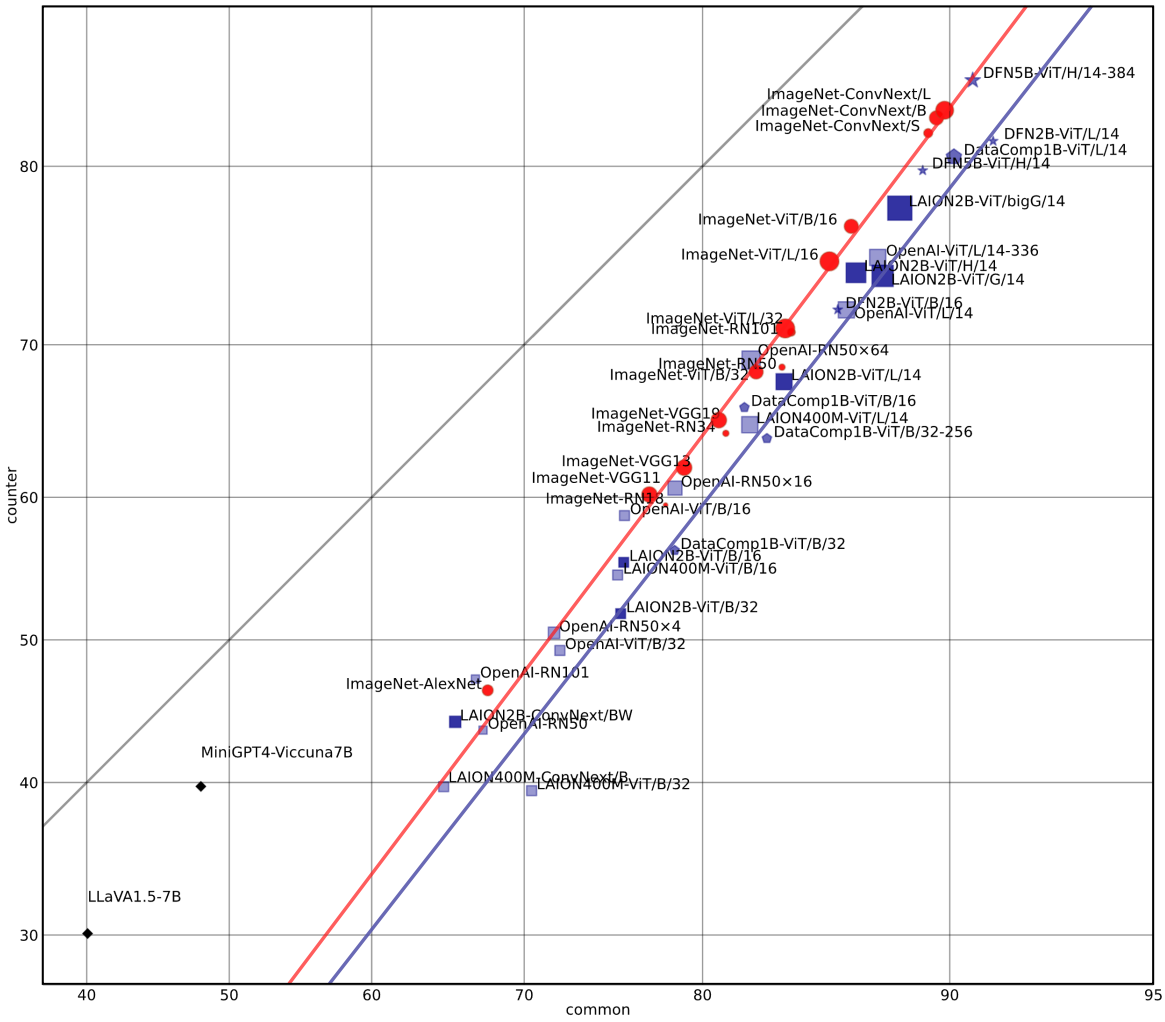


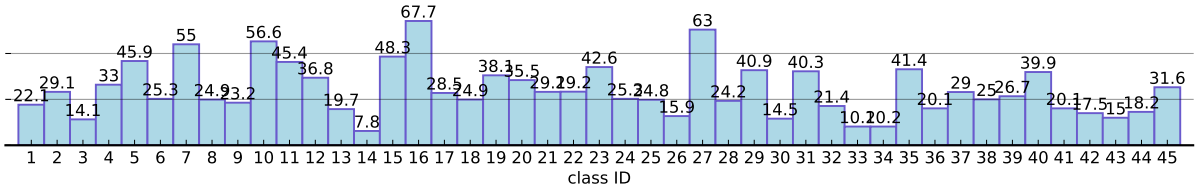
Figure 14. The `common` vs. `counter` performance (%) for CLIPs, ImageNet models, and more advanced LVLMs, following the 1 vs. 20 setup. We also present the model setups for each `common-counter` result pair.

Table 15. Class-wise 1 vs. 1000 performance on `CounterAnimal` for different backbones CLIP-trained on `LAION400M`.

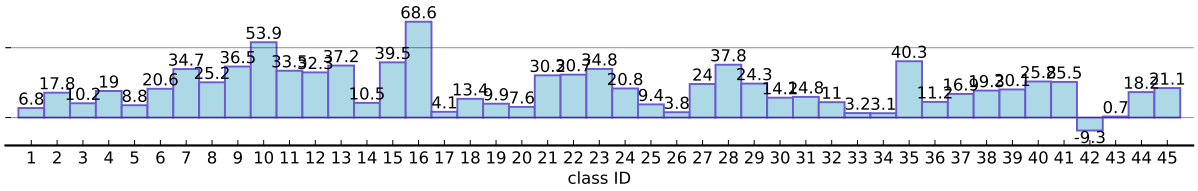
class ID	CLIP-LAION400M-ViT/B/16			CLIP-LAION400M-ViT/B/32			CLIP-LAION400M-ViT/L/14		
	common	counter	drop	common	counter	drop	common	counter	drop
1	71.36	64.60	6.76	79.61	57.52	22.09	93.20	91.15	2.05
2	87.18	69.37	17.81	78.63	49.55	29.08	94.02	75.68	18.34
3	18.85	8.65	10.20	28.69	14.59	14.09	14.75	7.57	7.19
4	90.00	70.99	19.01	81.15	48.15	33.01	94.23	90.12	4.11
5	76.19	67.35	8.84	87.76	41.84	45.92	97.96	82.65	15.31
6	88.32	67.72	20.60	73.36	48.10	25.26	83.94	68.35	15.59
7	78.64	43.96	34.68	73.64	18.68	54.96	81.36	69.23	12.13
8	69.23	44.00	25.23	73.85	49.00	24.85	87.69	74.00	13.69
9	74.00	37.50	36.50	54.00	30.83	23.17	54.00	39.17	14.83
10	79.92	26.00	53.92	60.64	4.00	56.64	69.48	13.00	56.48
11	62.43	28.87	33.55	74.26	28.87	45.39	60.95	42.96	17.99
12	83.52	51.19	32.33	72.53	35.71	36.81	89.01	72.62	16.39
13	64.04	26.83	37.21	22.17	2.44	19.73	17.24	7.32	9.92
14	63.60	53.06	10.54	32.46	22.45	10.01	64.04	44.90	19.14
15	61.54	22.09	39.45	67.95	19.68	48.27	85.90	18.47	67.42
16	82.12	13.50	68.62	68.87	1.23	67.65	88.08	50.92	37.16
17	56.09	52.00	4.09	48.50	20.00	28.50	77.25	52.80	24.45
18	68.35	54.92	13.43	29.11	4.17	24.95	87.34	69.32	18.02
19	83.98	74.05	9.93	81.82	43.67	38.15	91.34	70.89	20.46
20	67.21	59.62	7.60	55.74	20.19	35.55	75.41	70.19	5.22
21	67.31	37.12	30.19	73.08	43.94	29.14	71.15	56.82	14.34
22	87.72	57.01	30.71	96.49	67.29	29.20	100.00	80.37	19.63
23	85.33	50.57	34.75	59.85	17.24	42.60	83.78	41.38	42.40
24	98.77	78.00	20.77	88.34	63.00	25.34	98.77	95.00	3.77
25	98.04	88.68	9.36	93.63	68.87	24.76	99.02	86.79	12.23
26	5.60	1.81	3.79	20.00	4.07	15.93	43.20	8.60	34.60
27	86.42	62.42	24.00	77.78	14.77	63.01	85.19	78.52	6.66
28	65.48	27.72	37.76	79.70	55.45	24.25	91.37	82.18	9.19
29	92.20	67.92	24.27	80.49	39.62	40.87	95.12	83.02	12.10
30	96.98	82.83	14.15	86.21	71.72	14.49	99.14	93.94	5.20
31	93.10	78.30	14.80	82.76	42.45	40.31	94.83	94.34	0.49
32	95.71	84.72	10.99	85.24	63.89	21.35	98.57	97.22	1.35
33	83.24	80.00	3.24	92.20	82.00	10.20	86.42	80.00	6.42
34	65.03	61.90	3.13	69.23	59.05	10.18	76.92	71.43	5.49
35	76.42	36.13	40.29	67.48	26.05	41.43	88.62	61.34	27.27
36	16.92	5.75	11.17	33.85	13.72	20.13	83.08	67.70	15.38
37	79.47	62.61	16.86	74.90	45.95	28.96	93.16	82.43	10.72
38	96.70	77.36	19.34	80.66	55.66	25.00	98.11	83.02	15.09
39	99.21	79.09	20.12	97.62	70.91	26.71	100.00	90.91	9.09
40	49.23	23.40	25.83	56.92	17.02	39.90	58.46	14.89	43.57
41	86.90	61.36	25.53	68.97	48.86	20.10	80.69	56.82	23.87
42	75.73	85.00	-9.27	84.47	67.00	17.47	90.29	93.00	-2.71
43	67.37	66.67	0.70	37.89	22.92	14.98	64.21	60.42	3.79
44	22.08	3.92	18.16	18.18	0.00	18.18	72.73	24.51	48.22
45	72.52	51.43	21.09	72.52	40.95	31.57	80.92	53.33	27.58

Table 16: Class-wise 1 vs. 1000 performance on `CounterAnimal` for `ViT/B/32` CLIP-trained on different datasets.

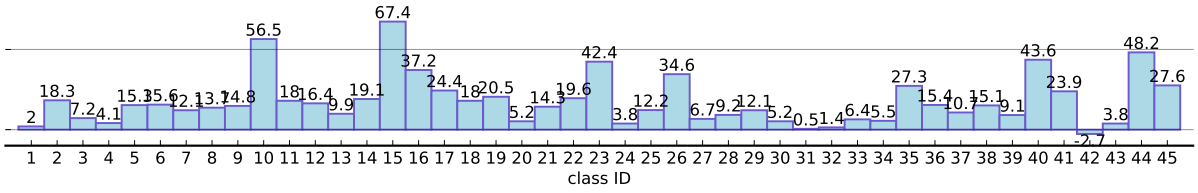
class ID	CLIP-LAION2B-ViT/B/32			CLIP-LAION400M-ViT/B/32			CLIP-OpenAI-ViT/B/32		
	common	counter	drop	common	counter	drop	common	counter	drop
1	86.41	79.65	6.76	79.61	57.52	22.09	81.55	66.37	15.18
2	86.32	72.97	13.35	78.63	49.55	29.08	85.47	58.56	26.91
3	10.66	9.73	0.93	28.69	14.59	14.09	18.85	12.43	6.42
4	91.54	74.69	16.85	81.15	48.15	33.01	77.69	38.27	39.42
5	75.51	55.10	20.41	87.76	41.84	45.92	61.22	38.78	22.45
6	83.58	64.56	19.02	73.36	48.10	25.26	77.74	65.82	11.91
7	72.27	29.67	42.60	73.64	18.68	54.96	88.64	67.03	21.60
8	92.31	77.50	14.81	73.85	49.00	24.85	87.69	72.50	15.19
9	44.00	25.00	19.00	54.00	30.83	23.17	70.00	35.83	34.17
10	87.55	43.00	44.55	60.64	4.00	56.64	67.87	15.00	52.87
11	68.64	45.07	23.57	74.26	28.87	45.39	53.85	11.27	42.58
12	82.42	41.67	40.75	72.53	35.71	36.81	78.02	61.90	16.12
13	31.53	11.38	20.14	22.17	2.44	19.73	38.92	16.26	22.66
14	60.09	47.96	12.13	32.46	22.45	10.01	17.98	18.37	-0.38
15	71.79	22.89	48.90	67.95	19.68	48.27	75.64	31.33	44.32
16	71.52	15.95	55.57	68.87	1.23	67.65	72.85	8.59	64.26
17	61.08	36.00	25.08	48.50	20.00	28.50	54.29	32.80	21.49
18	67.09	39.77	27.41	29.11	4.17	24.95	74.68	25.76	48.93
19	68.40	60.76	7.64	81.82	43.67	38.15	77.49	51.27	26.22
20	73.77	54.81	18.96	55.74	20.19	35.55	70.49	34.62	35.88
21	69.23	31.82	37.41	73.08	43.94	29.14	67.31	27.27	40.03
22	92.98	62.62	30.37	96.49	67.29	29.20	89.47	53.27	36.20
23	64.86	37.93	26.93	59.85	17.24	42.60	60.62	32.18	28.43
24	95.09	71.00	24.09	88.34	63.00	25.34	95.09	85.00	10.09
25	91.67	57.55	34.12	93.63	68.87	24.76	96.57	83.96	12.61
26	13.60	0.45	13.15	20.00	4.07	15.93	15.20	0.45	14.75
27	66.67	48.32	18.34	77.78	14.77	63.01	77.78	69.13	8.65
28	68.53	49.50	19.02	79.70	55.45	24.25	63.45	16.83	46.62
29	85.37	53.77	31.59	80.49	39.62	40.87	78.54	46.23	32.31
30	93.10	61.62	31.49	86.21	71.72	14.49	82.76	30.30	52.46
31	86.21	63.21	23.00	82.76	42.45	40.31	91.38	68.87	22.51
32	95.71	84.72	10.99	85.24	63.89	21.35	88.57	72.22	16.35
33	85.84	80.00	5.84	92.20	82.00	10.20	76.59	80.00	-3.41
34	49.65	36.19	13.46	69.23	59.05	10.18	69.93	56.19	13.74
35	78.05	21.85	56.20	67.48	26.05	41.43	73.17	48.74	24.43
36	61.54	42.92	18.62	33.85	13.72	20.13	58.46	46.46	12.00
37	83.27	51.80	31.47	74.90	45.95	28.96	85.55	62.22	19.34
38	92.92	68.87	24.06	80.66	55.66	25.00	78.77	47.17	31.60
39	96.03	76.36	19.67	97.62	70.91	26.71	100.00	93.64	16.36
40	64.62	34.04	30.57	56.92	17.02	39.90	56.92	25.53	31.39
41	86.21	54.55	31.66	68.97	48.86	20.10	86.21	84.09	2.12
42	72.82	69.00	3.82	84.47	67.00	17.47	71.84	71.00	0.84
43	67.37	59.38	7.99	37.89	22.92	14.98	69.47	62.50	6.97
44	63.64	19.61	44.03	18.18	0.00	18.18	10.39	2.94	7.45
45	70.99	48.57	22.42	72.52	40.95	31.57	35.88	30.48	5.40



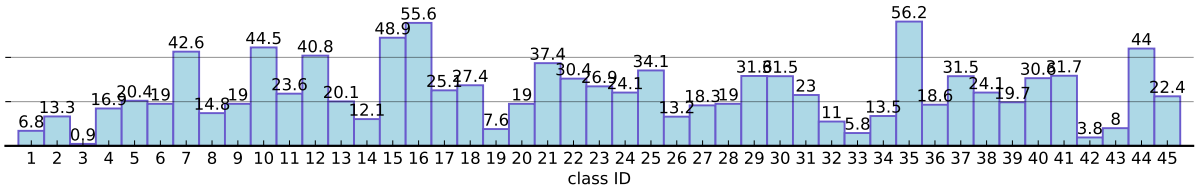
(a) CLIP-LAION400M-ViT/B/32



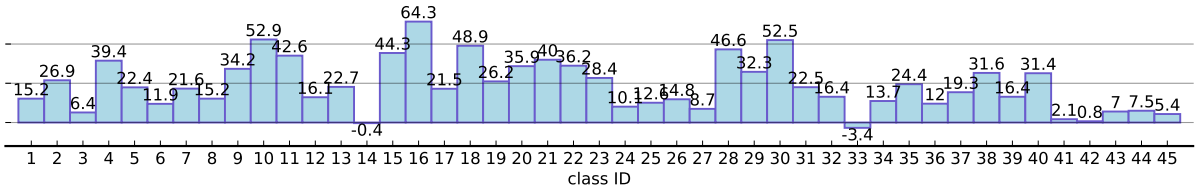
(b) CLIP-LAION400M-ViT/B/16



(c) CLIP-LAION400M-ViT/L/14



(d) CLIP-LAION2B-ViT/B/32



(e) CLIP-OpenAI-ViT/B/32

Figure 15. The performance drop (%) between common to counter on varying CLIP setups. The horizontal axis denotes the class ids and the vertical axis denotes the class-wise accuracy drop.