SIMPLESPOT and Evaluating Systemic Errors using Synthetic Image Datasets

Anonymous Authors[1]

Abstract

We introduce SynthSpot, a framework for generating synthetic datasets to use for evaluating methods for discovering blindspots (*i.e.,* systemic errors) in image classifiers, and SimpleSpot, a method for discovering such blindspots. We use SynthSpot to run controlled studies of how various factors influence blindspot discovery method performance. Our experimental results reveal several important shortcomings of existing methods, such as their relatively poor performance in settings with multiple model blindspots and their sensitivity to hyper-parameters. Further, we find that SimpleSpot is competitive with existing methods, which has promising implications for developing an interactive tool based on it.

1. Introduction

A growing body of research has established that machine learning models with high test performance often make systemic errors (Buolamwini and Gebru, 2018; Chung et al., 2019; Oakden-Rayner et al., 2020; Ribeiro et al., 2020; Singla et al., 2021; Ribeiro and Lundberg, 2022; Eyuboglu et al., 2022), which occur when the model performs significantly worse on a coherent (*i.e.,* semantically meaningful) subset of the data. For example, past works (Winkler et al., 2019; Mahmood et al., 2021) have demonstrated that models trained to diagnose skin cancer from dermoscopic images rely on spurious artifacts (such as the presence of a surgical skin marker that some dermatologists use to mark lesions) to make predictions. As a result, these models have different performance on images with or without those spurious artifacts. More broadly, being able to discover systemic errors is critical for many different reasons, including detecting algorithmic bias (Buolamwini and Gebru, 2018; Cabrera et al., 2019) or sensitivity to distribution shift (Sagawa et al., 2020; Singh et al., 2020).

While there is a rich history of work that studies how to find systemic errors in settings where there is useful metadata for defining coherent subsets of the data (Buolamwini and Gebru, 2018; Chung et al., 2019; Cabrera et al., 2019; Singh et al., 2020), finding systemic errors is much harder in settings without such metadata. For example, in many image classification settings we do not a-priori know which images may or may not have some artifact (such as a surgical skin marker) present. To address this challenge, several Blindspot Discovery Methods (BDMs)[1], such as (Sohoni et al., 2020; Singla et al., 2021; d'Eon et al., 2021; Eyuboglu et al., 2022), have been proposed to discover blindspots (*i.e.,* systemic errors) in settings without useful metadata.

While the goals and technical approaches of these BDMs are well-defined, there is no standardized approach for evaluating them. Evaluating the hypothesized blindspots returned by BDMs is fundamentally challenging because it is unclear how to measure their coherence and because we do not have the complete set of true blindspots to compare them against. One approach to address these challenges is to compare the hypothesized blindspots to a subset of the true blindspots that have either been artificially induced or identified in prior work (Sohoni et al., 2020; Eyuboglu et al., 2022). While this is a promising direction, it makes it difficult to measure a BDM's recall or false positive rate or to identify various factors that influence a BDM's performance.

To address these challenges, we propose SynthSpot which builds on the ideas from (Kim et al., 2022) by generating synthetic datasets with varying degrees of complexity and then training models on those datasets to have specific blindspots of varying types. This allows us to measure a BDM's recall and false positive rate, because we know the complete set of true blindspots for each model, and to measure how various factors, such as the number of blindspots in the model, influence a BDM's performance.

We use SynthSpot to conduct a evaluation of 3 recent BDMs and compare them to a new approach that we introduce, called SimpleSpot Our evaluation reveals several insights about BDM performance: performance degrades quickly as the number of blindspots in a model increases, performance depends on the type of features that define

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the SCIS workshop.

[1]Past works have used other terms such as discovering failure modes, unknown systemic errors, or slices. We chose "blindspot" because we believe it to be more descriptive and concise.

a blindspot, and that performance is very sensitive to the BDM's hyper-parameters (which are challenging to tune).

Interestingly, we observe that `SimpleSpot` consistently outperforms existing BDMs and is comparatively robust to hyper-parameter choice. Because `SimpleSpot` utilizes a 2-dimensional representation, our suggests that it could be be useful in an interactive setting. We plan to explore this direction using real datasets in future work.[2]

2. Synthetic Evaluation Framework

`SynthSpot` builds on ideas from (Kim et al., 2022) to generate synthetic datasets of varying complexity and to train models with specific blindspots on those datasets.

Dataset Definition. Each dataset can be described using *semantic features* that define the possible types of images that it contains. Datasets that have a larger number of features have a larger variety of images and therefore are more complex. For example, a very simple dataset may only contain images with blue squares while a more complicated dataset may also contain images with orange rectangles, striped circles, small text, and dotted backgrounds.

Blindspot Definition. Each blindspot is defined using a subset of the semantic features that define its associated dataset. Similar to how a dataset with more features is more complex, a blindspot that is defined using more features is more specific. For example, the "blue squares without circles" blindspot includes all such images, regardless of whether or not they have other objects or what the background looks like, and is defined with three features (the presence of a square, the color of the square, and the absence of a circle).

Training a Model with Specific Blindspots. For each dataset and set of blindspots, we train a ResNet-18 (He et al., 2016) to predict whether or not a square is present in the image. To do this, we generate data where the label for each image in the training and validation sets has the correct label if and only if it does not belong to any of the blindspots (the test set images are always correctly labeled).

Generating Diverse Experimental Configurations. Since our goal is to study how various factors influence BDM performance, we generate a diverse set of *experimental configurations* (*i.e.,* dataset, blindspots, and model triplets). To do this, we randomize the features that define a dataset (both the number of them and what they actually are) as well as the blindspots (the number of them, the number of features that define them, and what those features actually are). Note that, while generating the blindspots for a dataset, we make sure that they are uniquely identifiable.

[2]All code will be released at https://github.com/user/repo

3. Evaluation Metrics

A BDM returns an *ordered list* of hypothesized blindspots, $\hat{\mathbf{\Psi}} : [\hat{\Psi}_k]_{k=1}^{K}$, sorted by decreasing importance. In our experiments, we have access to the model's true blindspots, $\mathbf{\Psi} : \{\Psi_m\}_{m=1}^{M}$. Our goal is to measure *how well* $\hat{\mathbf{\Psi}}$ captures $\mathbf{\Psi}$. Note that the $\hat{\Psi}_k$ and Ψ_m are sets of images.

Blindspot Precision. We start by checking if $\hat{\Psi}_k$ is a subset of Ψ_m. If it is, we know that the model under-performs on $\hat{\Psi}_k$ and that $\hat{\Psi}_k$ is coherent. We measure this using the precision of $\hat{\Psi}_k$ with respect to Ψ_m:

$$\text{BP}(\hat{\Psi}_k, \Psi_m) = \frac{|\hat{\Psi}_k \cap \Psi_m|}{|\hat{\Psi}_k|} \tag{1}$$

Then, we say that $\hat{\Psi}_k$ *belongs to* Ψ_m if $\text{BP}(\hat{\Psi}_k, \Psi_m) > \lambda_p$.

However, $\hat{\Psi}_k$ can belong to Ψ_m without capturing the same information as Ψ_m. For example, $\hat{\Psi}_k$ could be "striped blue squares" while Ψ_m could be "striped squares, that are blue or orange". Because this excessive specificity could result in the user arriving at insufficiently general conclusions, we need to incorporate some notion of recall into the evaluation.

Blindspot Recall. One approach to measure recall is to calculate the proportion of Ψ_m that $\hat{\Psi}_k$ covers *individually*:

$$\text{BR}_{\text{naive}}(\hat{\Psi}_k, \Psi_m) = \frac{|\hat{\Psi}_k \cap \Psi_m|}{|\Psi_m|} \tag{2}$$

We relax this definition by allowing Ψ_m to be covered by the union of *multiple* of the $\hat{\Psi}_k$ that belong to it:

$$\text{BR}(\hat{\mathbf{\Psi}}, \Psi_m) = \frac{\left| \left(\bigcup_{\hat{\Psi}_k : \text{BP}(\hat{\Psi}_k, \Psi_m) > \lambda_p} \hat{\Psi}_k \right) \cap \Psi_m \right|}{|\Psi_m|} \tag{3}$$

Then, we say that $\hat{\mathbf{\Psi}}$ *covers* Ψ_m if $\text{BR}(\hat{\mathbf{\Psi}}, \Psi_m) > \lambda_r$.

We use this relaxation because "striped blue squares" and "striped orange squares" both belong to and collectively cover "striped squares, that are blue or orange". So a user could use them to arrive at the correct conclusion.

Discovery Rate (DR). We define the *Discovery Rate* of $\hat{\mathbf{\Psi}}$ and $\mathbf{\Psi}$ as the fraction of the Ψ_m that are covered by $\hat{\mathbf{\Psi}}$.

False Positive Rate (FPR). If the DR is non-zero, we define the *false positive rate* of $\hat{\mathbf{\Psi}}$ and $\mathbf{\Psi}$ as the fraction of the $\hat{\Psi}_k$ that do not belong to any of the $\mathbf{\Psi}$.[3] If the DR is zero, we exclude the FPR from our analysis.

[3]While calculating the DR, we may only need the top-u items of $\hat{\mathbf{\Psi}}$. As a result, we only calculate the FPR over those top-u items. This prevents this metric from being overly pessimistic when we intentionally pick K too large in our experiments.

Method	DR	FPR
Barlow	0.43 (0.04)	0.03 (0.01)
Spotlight	0.79 (0.03)	0.09 (0.01)
Domino	0.64 (0.04)	0.07 (0.01)
SimpleSpot	**0.85 (0.03)**	**0.03 (0.01)**

Table 1. Average BDM discovery rate and false positive rate and their standard errors across 100 experimental configurations.

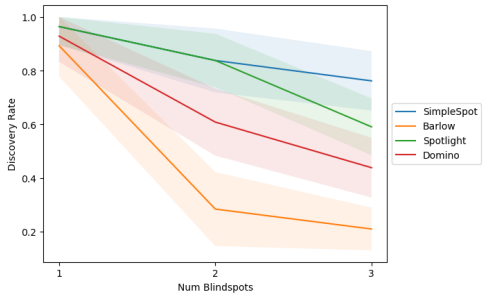

Figure 1. Average BDM DR (and 95% confidence intervals indicated by the shaded regions) for experimental configurations that have 1, 2, and 3 true blindspots.

4. SimpleSpot

SimpleSpot starts from the representation defined by the model's penultimate layer and uses scvis (Ding et al., 2018) to learn a 2-d embedding of that representation. This 2-d embedding is then normalized so each dimension has range [0, 1] and a 3rd dimension, consisting of $w *$ ModelConfidence(x, y), is appended. This 3-d input is then passed to a Gaussian Mixture Model clustering algorithm where the number of clusters is chosen using the Bayesian Information Criterion. The clusters are then sorted by the product of their error rate and the number of errors in them. w is a hyper-parameter that controls the relative importance of the 2-d embedding of an image and the model's confidence in its prediction for that image for the clustering.

5. Experiments

We use 100 experimental configurations generated by SynthSpot to study BDM performance. For each BDM, we use a held-out set of 20 configurations to select hyper-parameters. We use $\lambda_p = \lambda_r = 0.8$ for our metrics.

Overall Results. Table 1 shows the DR and FPR results averaged across all 100 experimental configurations. We observe that, in comparison to other methods, SimpleSpot has the highest DR, on average finding 85% of the true blindspots per experimental configuration. SimpleSpot and Barlow have a lower FPR than Spotlight and Domino.

The number of blindspots matters. In Figure 1, we plot the average DR for experimental configurations with 1, 2, and 3 true blindspots. Average DR decreases for all methods as the number of true model blindspots increases, albeit at different rates. Further, FPR increases as this happens as well (not shown). This insight particularly significant because past evaluations have almost exclusively focused on settings that consider only one true blindspot.

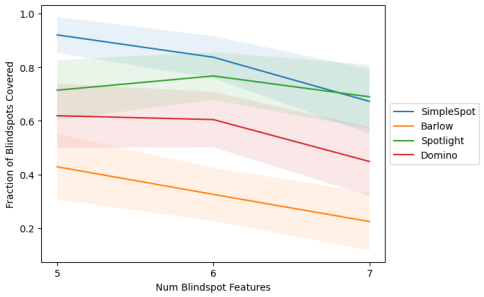

Figure 2. The fraction of true blindspots covered, based on all of the individual blindspots from the 100 experimental configurations, for blindspots defined using 5, 6, and 7 features.

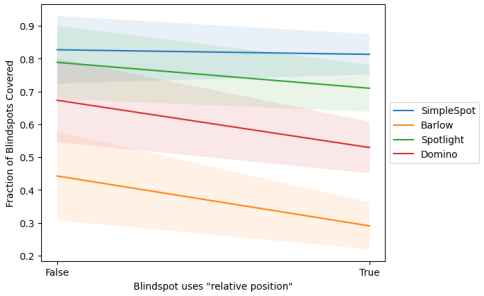

Figure 3. The fraction of true blindspots covered that either use (Right) or do not use (Left) the "relative position" feature.

The specificity of blindspots matters. In Figure 2, we plot the fraction of true blindspots covered for blindspots defined using 5, 6, and 7 features. With the exception of Spotlight, all of these methods are less able to find more specific/smaller blindspots.

The features that define a blindspot matter. In Figure 3, we plot the fraction of true blindspots covered for blindspots that either are or are not defined using the "relative position" feature; this feature is an indicator for whether the square in the image is located above its horizontal center-line. With the exception of SimpleSpot, all of these these are less able to find blindspots defined using this feature. This suggests that the types of features used to define a blindspot (*e.g.,* the presence of spurious objects, color or texture information, background information, or location information) may influence BDM performance.

There should be more discussion on hyper-parameter

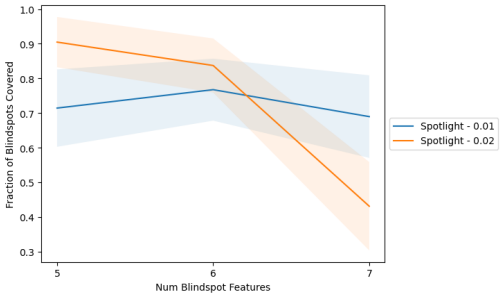

Figure 4. The fraction of true blindspots covered for blindspots defined using $5, 6$, and 7 features. Despite the fact that these hyper-parameter choices perform nearly identically on average, they behave differently in different settings.

tuning. In Figure 4, we observe that two hyper-parameter settings that perform nearly identically on average can exhibit very different performance in different settings. This suggests that their may not be a single best hyper-parameter choice to discover all of the blindspots in a single model, which could contain multiple blindspots of different specificity or size. In conjunction with the general sensitivity that these methods have to their hyper-parameters, this suggests that there should be more discussion on hyper-parameter tuning (especially since this is much easier in our controlled setting than in real applications).

6. Related Work

Finding Blindspots. Many methods have been proposed to help users discover blindspots across a wide set of applications. We focus on methods that make the least restrictive assumptions and that work for image classification models. Specifically, we evaluate the methods from Singla et al. (2021); d'Eon et al. (2021); Eyuboglu et al. (2022) because they do not assume any of the following:

- Access to metadata (Kim et al., 2018; Buolamwini and Gebru, 2018; Chung et al., 2019; Singh et al., 2020).
- Access to tools that manipulate the data (Shetty et al., 2019; Singla et al., 2020; Xiao et al., 2021; Leclerc et al., 2021; Bharadhwaj et al., 2021)
- Any specific model structure (Alvarez-Melis and Jaakkola, 2018; Koh et al., 2020).
- Any specific model training process (Higgins et al., 2017; Tsipras et al., 2019; Wong et al., 2021)
- A human in the loop, either through an interactive interface (Cabrera et al., 2019; Balayn et al., 2022; Ribeiro and Lundberg, 2022) or inspecting explanations (Yeh et al., 2020; Adebayo et al., 2021).

The methods we evaluate share a common approach to discovering blindspots: (1) they use a, potentially separate, *representation model* to get an embedding for each point, and (2) learn a *blindspot membership model* to predict if a point belongs to a particular blindspot given its embedding.

Quantitative Evaluations of BDMs. One evaluation approach focuses on measuring properties of the hypothesized blindspots, such as their error rate or size (Singla et al., 2021; d'Eon et al., 2021). However, while these properties are important, they do not capture whether the hypothesized blindspots are coherent.

Another approach is to compare the hypothesized blindspots to a subset of the true blindspots that have either been artificially induced or identified in prior work (Sohoni et al., 2020; Eyuboglu et al., 2022). This approach is similar to `SynthSpot`, with a few key differences:

- Past works use unrealistic definitions of what it means to discover a true blindspot that either consider only precision (Eyuboglu et al., 2022) or consider both precision and recall but use thresholds for precision that are too lenient (Sohoni et al., 2020). Further, neither allow hypothesized blindspots to be combined.
- Without access to the complete set of true blindspots, they cannot measure DR or FPR.
- They do not isolate factors that influence a BDM's performance (*e.g.,* number of blindspots, the specificity of those blindspots, or the features that define the blindspots).

7. Conclusion

We proposed `SynthSpot`, a synthetic evaluation framework for BDMs, and ran controlled studies of how various factors influence BDM performance. While running a fully synthetic evaluation has several limitations (*e.g.,* it remains a question for follow-up work to see if our observed trends generalize to settings with real image data), our evaluation yields fundamental insights about when and why different BDMs are less effective. More broadly, our work is an important step towards formalizing a more rigorous and complete set of desiderata for BDMs.

Our experimental results have many interesting implications for future work. First, it is important to evaluate BDMs in settings with models that have multiple blindspots, as this setting is more realistic and more challenging. Second, finding a way to tune BDM hyper-parameters in realistic settings is an open challenge with significant practical importance.

Finally, we are intrigued by the result that `SimpleSpot`, which learns a GMM on a 2-d embedding, performs competitively with prior methods, which use significantly higher dimensional embeddings. In future work, we hope to evaluate `SimpleSpot` on real image data to see if this result generalizes. If it does, this has promising implications for the development of an interactive blindspot discovery tool by replacing the GMM with a human in the loop.

References

Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91. PMLR, 2018.

Yeounoh Chung, Tim Kraska, Neoklis Polyzotis, Ki Hyun Tae, and Steven Euijong Whang. Slice finder: Automated data slicing for model validation. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, pages 1550–1553. IEEE, 2019.

Luke Oakden-Rayner, Jared Dunnmon, Gustavo Carneiro, and Christopher Ré. Hidden stratification causes clinically meaningful failures in machine learning for medical imaging. In *Proceedings of the ACM conference on health, inference, and learning*, pages 151–159, 2020.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of nlp models with checklist. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, 2020.

Sahil Singla, Besmira Nushi, Shital Shah, Ece Kamar, and Eric Horvitz. Understanding failures of deep networks via robust feature extraction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12853–12862, 2021.

Marco Tulio Ribeiro and Scott Lundberg. Adaptive testing and debugging of nlp models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3253–3267, 2022.

Sabri Eyuboglu, Maya Varma, Khaled Kamal Saab, Jean-Benoit Delbrouck, Christopher Lee-Messer, Jared Dunnmon, James Zou, and Christopher Re. Domino: Discovering systematic errors with cross-modal embeddings. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=FPCMqjI0jXN.

Julia K Winkler, Christine Fink, Ferdinand Toberer, Alexander Enk, Teresa Deinlein, Rainer Hofmann-Wellenhof, Luc Thomas, Aimilios Lallas, Andreas Blum, Wilhelm Stolz, et al. Association between surgical skin markings in dermoscopic images and diagnostic performance of a deep learning convolutional neural network for melanoma recognition. *JAMA dermatology*, 155(10):1135–1141, 2019.

Usman Mahmood, Robik Shrestha, David DB Bates, Lorenzo Mannelli, Giuseppe Corrias, Yusuf Emre Erdi, and Christopher Kanan. Detecting spurious correlations with sanity tests for artificial intelligence guided radiology systems. *Frontiers in digital health*, page 85, 2021.

Ángel Alexander Cabrera, Will Epperson, Fred Hohman, Minsuk Kahng, Jamie Morgenstern, and Duen Horng Chau. Fairvis: Visual analytics for discovering intersectional bias in machine learning. In *2019 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 46–56. IEEE, 2019.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=ryxGuJrFvS.

Krishna Kumar Singh, Dhruv Mahajan, Kristen Grauman, Yong Jae Lee, Matt Feiszli, and Deepti Ghadiyaram. Don't judge an object by its context: Learning to overcome contextual bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11070–11078, 2020.

Nimit Sohoni, Jared Dunnmon, Geoffrey Angus, Albert Gu, and Christopher Ré. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. *Advances in Neural Information Processing Systems*, 33:19339–19352, 2020.

Greg d'Eon, Jason d'Eon, James R Wright, and Kevin Leyton-Brown. The spotlight: A general method for discovering systematic errors in deep learning models. *arXiv preprint arXiv:2107.00758*, 2021.

Joon Sik Kim, Gregory Plumb, and Ameet Talwalkar. Sanity simulations for saliency methods. *ICML*, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Jiarui Ding, Anne Condon, and Sohrab P Shah. Interpretable dimensionality reduction of single cell transcriptome data with deep generative models. *Nature communications*, 9 (1):1–13, 2018.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pages 2668–2677. PMLR, 2018.

Rakshith Shetty, Bernt Schiele, and Mario Fritz. Not using the car to see the sidewalk – quantifying and controlling the effects of context in classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Sumedha Singla, Brian Pollack, Junxiang Chen, and Kayhan Batmanghelich. Explanation by progressive exaggeration. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=H1xFWgrFPS.

Kai Yuanqing Xiao, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. Noise or signal: The role of image backgrounds in object recognition. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=gl3D-xY7wLq.

Guillaume Leclerc, Hadi Salman, Andrew Ilyas, Sai Vemprala, Logan Engstrom, Vibhav Vineet, Kai Xiao, Pengchuan Zhang, Shibani Santurkar, Greg Yang, et al. 3db: A framework for debugging computer vision models. *arXiv preprint arXiv:2106.03805*, 2021.

Homanga Bharadhwaj, De-An Huang, Chaowei Xiao, Anima Anandkumar, and Animesh Garg. Auditing ai models for verified deployment under semantic specifications. *arXiv preprint arXiv:2109.12456*, 2021.

David Alvarez-Melis and Tommi S Jaakkola. Towards robust interpretability with self-explaining neural networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 7786–7795, 2018.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International Conference on Machine Learning*, pages 5338–5348. PMLR, 2020.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=SyxAb30cY7.

Eric Wong, Shibani Santurkar, and Aleksander Madry. Leveraging sparse linear layers for debuggable deep networks. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 11205–11216. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/wong21b.html.

Agathe Balayn, Natasa Rikalo, Christoph Lofi, Jie Yang, and Alessandro Bozzon. How can explainability methods be used to support bug identification in computer vision models? In *CHI Conference on Human Factors in Computing Systems*, pages 1–16, 2022.

Chih-Kuan Yeh, Been Kim, Sercan Arik, Chun-Liang Li, Tomas Pfister, and Pradeep Ravikumar. On completeness-aware concept-based explanations in deep neural networks. *Advances in Neural Information Processing Systems*, 33:20554–20565, 2020.

Julius Adebayo, Michael Muelly, Harold Abelson, and Been Kim. Post hoc explanations may be ineffective for detecting unknown spurious correlation. In *Proceedings of the 2022 International Conference on Learning Representations*, 2021.