VL*Slice*: Interactive Vision-and-Language Slice Discovery

Eric Slyman
Oregon State University
Corvallis, OR, USA
slymane@oregonstate.edu

Minsuk Kahng
Google Research
Atlanta, GA, USA
kahng@google.com

Stefan Lee
Oregon State University
Corvallis, OR, USA
leestef@oregonstate.edu

Abstract

Recent work in vision-and-language demonstrates that large-scale pretraining can learn generalizable models that are efficiently transferable to downstream tasks. While this may improve dataset-scale aggregate metrics, analyzing performance around hand-crafted subgroups targeting specific bias dimensions reveals systemic undesirable behaviors. However, this subgroup analysis is frequently stalled by annotation efforts, which require extensive time and resources to collect the necessary data. Prior art attempts to automatically discover subgroups to circumvent these constraints but typically leverages model behavior on existing task-specific annotations and rapidly degrades on more complex inputs beyond "tabular" data, none of which study vision-and-language models. This paper presents VLSlice*, an interactive system enabling user-guided discovery of coherent representation-level subgroups with consistent visiolinguistic behavior, denoted as vision-and-language slices, from unlabeled image sets. We show that* VLSlice *enables users to quickly generate diverse high-coherency slices in a user study (n=22) and release the tool publicly*[1].

1. Introduction

Large-scale vision-and-language models trained on curated [25, 9, 36] and web-scrapped [29, 18, 8] data have led to significant improvements over task-specific models when transferred to downstream tasks in terms of aggregate metrics. However, researchers probing these models on hand-curated datasets have revealed problematic behaviors and well-known biases [30, 33] learned during pretraining – *e.g.* biases with respect to perceived gender, skin tone,[2] and occupation. These biases can lead to disparate representational performance for population subgroups, resulting in poor prediction quality for downstream applications such as image captioning [15, 37] and search [28].

In the standard paradigm for bias analysis in vision-and-language models, researchers query and analyze a set of images that potentially exhibit bias. They often select a subject population of interest, some specific subgroups of those subjects to analyze, and a bias dimension to measure against the model [30, 33]. For example, Ross *et al.* [30] choose images of people as their subject population, label these based on perceived gender and skin tone categories, and measure model-predicted affinities between the labeled image subsets and text describing occupations or (un)pleasantness.

To effectively support analysis of such image sets for researchers, the set of images returned for the subject population subsets should be **large**, **coherent**, and **representative** – *i.e.* containing enough images to make statistically significant statements, capturing a well-defined visual concept, and covering the full diversity of visual presentation for the selected concept rather than an arbitrary subset. Without these, the biases may simply be noise (~~large~~), be obscured by effects from images outside the intended subject group being included (~~coherent~~), or be the result of some intersectional bias captured in the subset that is not consistent across the whole expression of the visual concept (~~representative~~).

Collecting and labeling appropriate image sets that fulfill these properties can be an arduous task. Despite this, manual annotation of static datasets along predefined subgroup and bias dimensions is the standard practice [19, 17]. This data collection methodology is expensive to perform – effectively limiting broad bias-auditing to high-resource institutions. Further, the one-off nature of this labeling process limits the scope of testing to pre-identified biases and does not account for how concepts may shift in visual expression or cultural convention over time.

Several methods have been proposed to automatically discover biased "*slices*" of data which share similar input attributes and exhibit consistent responses from machine learning models [10, 31, 14, 11, 32, 21]. These Slice Discovery Methods (SDMs) have typically been deployed in

[1] https://github.com/slymane/vlslice

[2] We use the term 'skin tone' rather than 'race' as race is a socially constructed identity that can span a range of phenotypic features.

tabular input settings where individual input dimensions are semantically meaningful. While some recent work has explored extending SDMs to more complex inputs like images [14, 11, 32, 21], these require task-specific annotations to evaluate the model – making them unsuitable to auditing general vision-and-language alignment models.

To improve this workflow, we propose VL*Slice*, an interactive system to discover vision-and-language slices from unlabeled collections of images. VL*Slice* consists of four primary stages of user-driven interaction with a vision-and-language alignment model of interest as depicted in the system overview in Fig. 1. Ⓐ First, users write a query defining a subject population of interest (*e.g.*, "*person*", "*car*") and bias dimension to measure (*e.g.*, "*intelligent*", "*fast*") which is submitted to VL*Slice* to select from a large set of unlabeled images down to a subset of subject-relevant images, then cluster those images by visual similarity and alignment with the bias dimension. Ⓑ Second, users are displayed the clusters generated by VL*Slice* and can search, filter, and sort those clusters in (un)directed searches to identify and capture candidate slices (*e.g.*, "*people wearing suits*", "*red cars*"). Ⓒ Next, users interact with *VLSlice* in a loop viewing recommended similar and counterfactual clusters to their slice to gather more coherent and representative samples. Ⓓ Finally, users can view a plot that shows the relationship between the slice they formed and the bias term across the entire subject population of interest, validating if biased model behavior is demonstrated in the slice.

We demonstrate that VL*Slice* enables users to quickly generate diverse high-coherency slices in a between-subjects user study (n=22), contrasting with a control interface mimicking a linear unguided image search. From this study, we present both qualitative and quantitative support, and discuss emergent user interaction paradigms. We choose to study CLIP [29] as a representative model of contemporary methods in large-scale pretraining and self-supervision for image-text alignment.

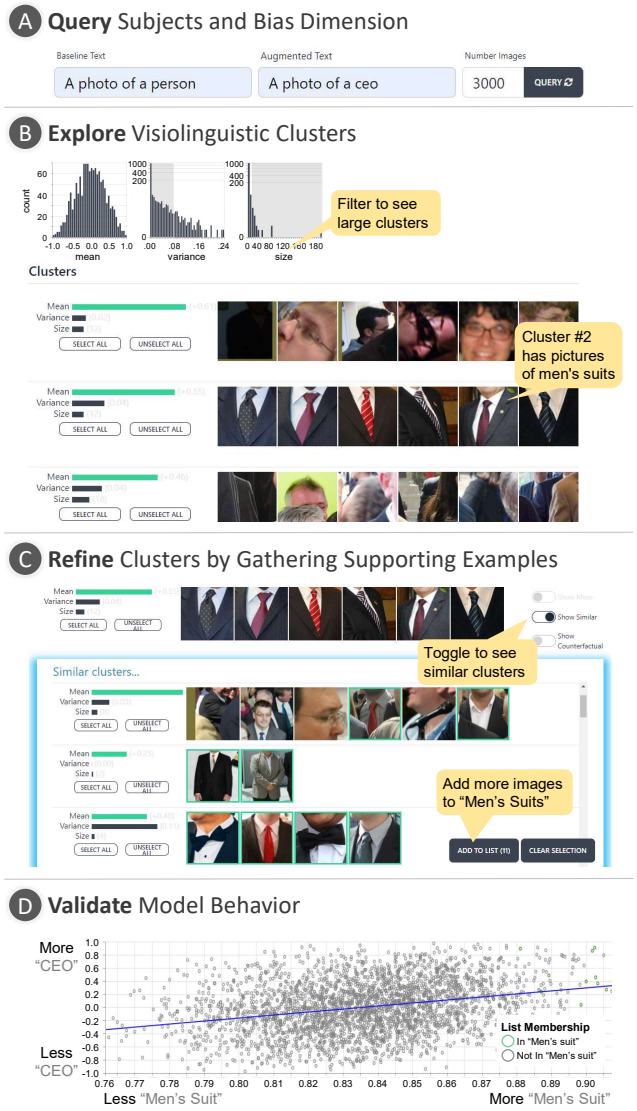

Figure 1: An example user workflow with VL*Slice*. The user workflow begins with writing a **(A) Query** to the model then **(B) Explor**ing the resulting visiolinguistic clusters to find interesting candidates to begin building a slice from. Once users identify a hypothesis, they can **(C) Refine** the clusters by gathering additional samples in a human-in-the-loop manner with VL*Slice* recommending similar and counterfactual examples to add to the clusters. Finally, users can **(D) Validate** the bias behavior of the model on this slice.

2. Related Work

Vision and Language Bias. Both vision and language models are independently known to harbor biases leading to representational harm. For example, gender and skin tone [4] in vision systems, associating racial minorities with animals [13], gendering professions in language models [3], and a litany of others documented in [2, 34]. Multimodal vision-and-language models are not exempt from these tendencies [26, 15, 28] and may in fact exacerbate them [33].

Many modern, high-performing vision-and-language alignment models are pretrained on scraped internet data [29, 18, 8] – a strategy that improves performance but has been shown to teach models "*misogyny, pornography, and malignant stereotypes*" [1] across multiple studies [1, 30, 33]. As discussed in Sec. 1, studying these biases often requires labelling thousands of images for specific preconceived subject groups and bias dimensions (*e.g.* gender [33] and skin tone [30, 16], or emotion [27]). Our proposed interactive method allows for more open-ended exploration of bias and reduces the burden of collecting relevant subject group image sets.

Slice Discovery & Exploration. Slice discovery methods (SDMs) attempt to find critical subgroups (or *slices*) of data with common input properties and consistent predictions (often, mis-predictions) with respect to some model. Much of this work is developed for tabular data settings where slices are defined based on categorical (e.g., gender, occupation) [10, 31] and numerical attributes (e.g., age, duration in days) [6]. In these settings, identifying semantically coherent subsets is more straightforward than in high-dimensional perceptual data like images where individual input dimensions are non-semantic. Once slices are discovered, they can be interactively inspected to understand model behavior. For the case of slices that contain many mis-predicted examples, they can be leveraged to improve model performance, such as by augmenting the training set with additional examples within the slice [7, 35].

Recent SDM work explores "*unstructred*" data like images, where each data item is not necessarily associated with structured attributes [14, 11, 32, 21]. Commonly, these approaches attempt to extract semantically meaningful clusters from their high-dimensional embedding representations. For instance, Spotlight [11] identifies contiguous representation space regions that contain data items with high loss. Domino [14] additionally leverages a joint input-and-language embedding space to generate natural language descriptions of the extracted slices after input-space clustering, which can potentially aid analysis. VL*Slice*, in contrast, examines the relationship between two modalities rather than the nature of an input and a model's task specific predictions. We note that the visiolinguistic relationship captured by image-text alignment may be viewed as an image classification task with an extremely large and sparse label space (all natural language strings). Under this setting, VL*Slice* examines a sparsely labeled many-to-many relationship between inputs where an image (caption) may match many captions (images) but only a single relationship is known. No SDMs exist that examine relationships of this noisy multimodal nature.

Moreover, automated SDMs for unstructured data have critical limitations. Because of the nature of unsupervised methods, extracted clusters cannot perfectly align with semantically meaningful concepts, bias, and human knowledge [23]. Therefore, results from automated methods need to be inspected and refined by human users (e.g., merge, split, add) to identify slices that capture the necessary concepts while meeting the desired properties such as coherency and representativeness. VL*Slice* mitigates the noise in multimodal relationships and improves alignment to semantically meaningful concepts by providing tooling for human-in-the-loop slice discovery and refinement.

The field of *visual analytics* has developed methods and tools that leverage the power of humans in data analysis [20]. Visual analytics tools frequently serve to help users explore noisy slices returned by SDMs. FairVis [6] allows users to discover subgroups that exhibit bias by interactively analyzing the clustering of tabular data. Zhao *et al.* [39] enables users to train a binary classifier for each slice through active learning, based on their analysis of the initial clustering of image patches. While most existing work targets a single modality, Cabrera *et al.* [5] target image captioning tasks. In contrast to their focus on human's comprehensive sensemaking of non-grouped image sets, VL*Slice* aims to use human inputs minimally by letting human users start their analysis from clustered results.

3. VL*Slice*

We propose VL*Slice*, an interactive system to discover and build slices from large-scale unlabeled image sets with respect to a vision-and-language (ViL) model of interest. Specifically, our methodology is designed to test ViL models which produce image-text alignment scores and support clustering in the image feature space – a common feature of many popular models (*e.g.* CLIP [29]).

The following subsections describe the technical details of VL*Slice* and provide a running example of how they support the user workflow depicted in Fig. 1. The workflow is roughly split into four parts – Ⓐ query specification (Sec. 3.1), Ⓑ exploration of presented clusters (Sec. 3.2), Ⓒ iterative slice refinement (Sec. 3.3), and Ⓓ validation of the observed phenomenon (Sec. 3.4).

3.1. Caption-based Querying

Users must first write a query defining the domain of subjects (*e.g.*, people, cars, houses) and bias dimension (*e.g.*, CEO-like, fast, pleasant) they are interested in evaluating by specifying the baseline and augmented captions in the VL*Slice* interface, denoted respectively as C_b and C_a. The interface will then define a *working set* $\mathcal{I}_w$ of relevant images by filtering a large unlabelled image set to just the k images most aligned with the baseline caption. The choice of k is left to the user and is a trade-off between precision and recall of subjects captured within the working set. We discuss this trade-off further in Sec. 7.

Running Example (Fig. 1 Ⓐ). The user enters "*A photo of a person*" as the baseline caption C_b with k=3000 to restrict the working set to 3000 "*people*"-images and "*A photo of a CEO*" as the augmented caption C_a to define a "*CEO*"-ness bias dimension to explore.

Measuring Affinity with the Augmented Caption. Without loss of generality, we can consider vision-and-language affinity models to be functions $f(I, C)$ that compute some score reflecting if caption C describes the contents of the image I. A natural approach for measuring the model's predicted affinity between each working set image $I_i \in \mathcal{I}_w$ and the bias dimension is then to compute the *augmented*

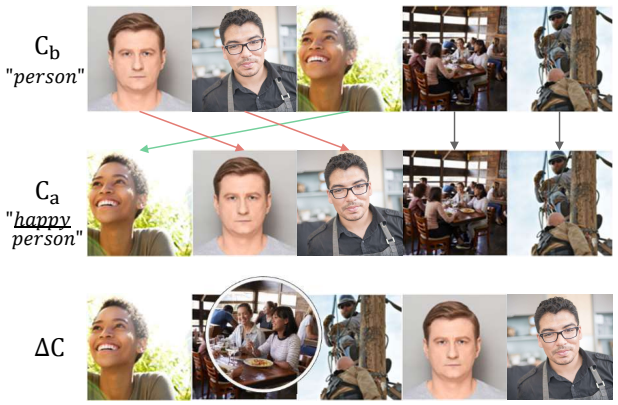

Figure 2: Sample rankings by baseline caption (C_b = "*person*"), augmented caption (C_a = "*happy person*"), and ΔC, with highest on the left. The change in percentile from C_b to C_a is shown with green arrows for positive changes, red arrows for negative, and gray arrows for neutral. We enlarge the photo of people with smiling faces eating a meal. The rank of this photo does not change from C_b to C_a (4th), but increases (2nd) under ΔC. Captions are prepended with "*A photo of a* ____" in practice.

caption similarity $S_i^a = f(I_i, C_a)$. However, our initial experiments suggest this is not always sufficient.

In Fig. 2, we consider cases where the augmented caption *extends* the baseline caption, *e.g.* "*A photo of a person*" vs. "*A photo of a happy person*". In these cases, well-framed canonical images of the subject may retain higher scores under the augmented caption despite actually *reducing* in magnitude compared to their affinity to the baseline caption – due solely to a strong alignment with the subject. In this instance, we would like to disentangle the model-of-interest's learned notions of "*person*" from that of "*happy*" which we wish to examine.

To ameliorate this effect, we consider a measure based on the change in similarity from the baseline to augmented caption. Analogous to the augmented caption similarity S_i^a, let $S_i^b = f(I_i, C_b)$ be the baseline caption similarity for image I_i. With S^a and S^b denoting the empirical distributions of these similarities over the working set $\mathcal{I}_w$, let $P_i^a = P(S^a \leq S_i^a)$ and $P_i^b = P(S^b \leq S_i^b)$ be the empirical percentile rank of caption similarity for image I_i for the augmented and baseline captions respectively.

We can then define the *change in augmented caption percentile* ΔC for image I_i as

$$\Delta C(I_i) = P_i^a - P_i^b. \quad (1)$$

Intuitively, images that report *greater* affinity with the augmented caption than the baseline caption relative to other working set images will achieve a higher ΔC. As shown in Fig. 2, this results in reduced effect of canonical images. Further, this score is bounded between ± 1 and avoids forcing users to reason about absolute changes in affinity magnitude which may vary greatly in scale for different settings.

3.2. Exploring Visiolinguistic Clusters

To assist users in identifying groups of related images with consistent affinity to the augmented caption, we display the working set images as clusters rather than a long list of individual images. These accelerate users' ability to find examples, prime them for identifying shared visual features, and provides a convenient bootstrap for forming slices.

We form clusters using standard agglomerative clustering with average linkage stopping at distance threshold $dt = 0.2$. To capture both visual similarity and bias-effect consistency, we define the linkage affinity between two images I_i and I_j as a combination of their visual dissimilarity D_{img} and the difference between their ΔC's. Given a visual encoder $\Phi : I \rightarrow \mathbb{R}^d$ from our model-of-interest, we compute the visual cosine distance as

$$D_{img}(I_i, I_j) = 1 - \Phi(I_i)^T \Phi(I_j) / \|\Phi(I_i)\| \|\Phi(I_j)\|, \quad (2)$$

the ΔC-based affinity consistency distance as

$$D_{\Delta C}(I_i, I_j) = |\Delta C(I_i) - \Delta C(I_j)|, \quad (3)$$

and the overall linkage affinity between I_i and I_j as

$$D(I_i, I_j) = a * D_{img}(I_i, I_j) + (1 - a) * D_{\Delta C}(I_i, I_j) \quad (4)$$

where a controls the trade-off between clustering by visual similarity and affinity consistency. We set $a = 0.95$.

By default, clusters are displayed in descending order by mean ΔC. Each cluster is displayed with a set of sample images along with numeric attributes for cluster size and mean / variance of ΔC within the cluster. Additionally, histograms for these numeric attributes are displayed and users may filter the displayed clusters by specifying ranges of attribute values. Users may change the ordering of clusters by ranking on different attributes or by making a directed search over clusters with any arbitrary text (*e.g.*, "*glasses*"). In this case, clusters are re-ranked by average image-text similarity from the model-of-interest. To construct slices, users create lists of images according to the properties enumerated in Sec. 1. Users may select individual image instances or entire clusters, then add-to an existing or new slice. Upon slice creation, users are prompted to provide a name according to the captured visual feature(s) and may change that name later as the slice is refined.

Running Example (Fig. 1 Ⓑ). The user interacts with the histogram filters to find large clusters with low variance in ΔC to drill-down on high-quality clusters likely to prompt some bias hypothesis. Then, they browse the sorted clusters to discover a visual concept corresponding to men's suits.

3.3. User-guided Cluster Refinement

Once users have identified a slice they would like to explore by capturing some example images, our interface provides tooling to assist with discovering additional examples to improve the size of the sample and representation of the visual concept. Specifically, users may request *similar* and *counterfactual* clusters for a slice. We measure cluster similarity based on the cosine between cluster centroids. When *similar* clusters are requested for a slice, we simply rank clusters by visual similarity to the slice and display the nearest 50 clusters. For *counterfactual* clusters, we first filter to clusters which have mean ΔC's with opposing sign as the slice and then sort by visual similarity.

Both similar and counterfactual clusters are updated reactivity as the user gathers samples (and thus change the slice centroid embedding), allowing users to refine and extend their slice in an iterative process. All clusters which contain at least one image already captured by the user are filtered out to avoid displaying samples which have already been considered for that slice. These tools help users find visually similar images to expand their slices and help to guard against users selecting non-representative subsets of the intended visual concept that happen to display bias (as demonstrated in in Fig. 3).

Running Example (Fig. 1 C). From the cluster displaying images of men's suits, the user requests similar clusters and begins adding more images of suits to their slice to expand the slice's coverage of the men's suit visual concept.

3.4. Validating Model Behavior

Once users have formed a slice, they can examine the mean ΔC to draw some conclusions about the model-of-interest's bias. However, it is useful to consider the trend over a larger set of images to determine if it is likely to hold beyond the slice. To achieve this, users can request a correlation scatter plot that charts visual similarity to the slice centroid against ΔC for each image in the working set. Observing a strong linear relationship in this plot provides additional evidence that the visual concept captured in the slice has a consistent effect on caption affinity. Further, this plot is useful for identifying outlier instances with unexpected behavior (*e.g.* high similarity but different ΔC) which users may add or remove from their slice.

Running Example (Fig. 1 D). The user requests a correlation plot to validate the bias behavior – observing a positive slope where visual similarity to the "*men's suit*" slice is predictive of higher "*CEO*"-ness under the model-of-interest.

4. VL*Slice* User Study

To evaluate VL*Slice*, we conduct a user study comparing it with a baseline interface, ListSort, representative of typical strategies used to explore model behavior.

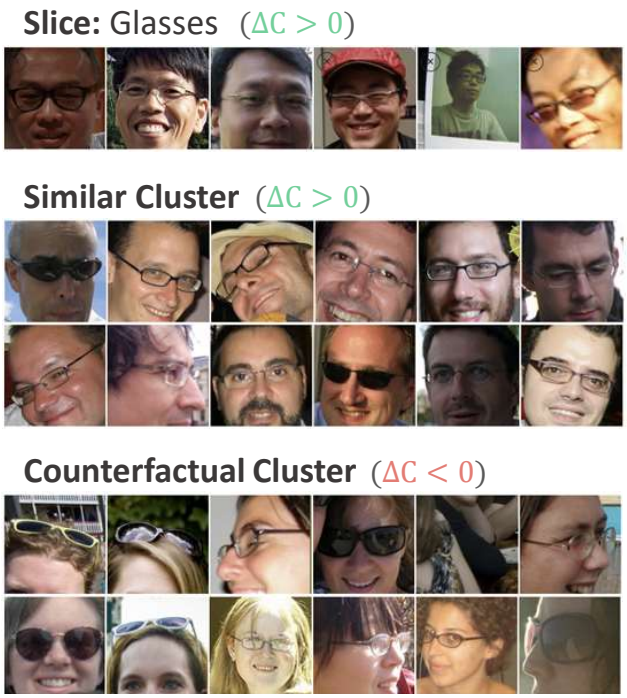

Figure 3: Similar and counterfactual clusters for a slice capturing an unintended subset of "*glasses*" for the query C_b = "*A photo of a person*", C_a = "*A photo of a CEO.*" While the similar clusters display additional masculine presenting glasses-wearers with positive ΔC, counterfactuals help escape this region by displaying a cluster of feminine presenting glasses-wearers with opposing negative ΔC.

Baseline: ListSort. The ListSort baseline interface uses the same query inputs and list construction as VL*Slice*, but several key components are removed. The interface simply sorts images by their change in augmented caption percentile (ΔC), then displays those ranked images to the user without any clustering or human-in-the-loop interactions. The user does not have the ability to view similar or counterfactual images, cannot do additional sorting or filtering, and cannot view the correlation scatter plot. We chose this interface as the baseline because there is no existing tool designed for the task VL*Slice* supports, and we design the ListSort interface as a representation of the current workflow of ML practitioners. They often simply visualize a sorted list of results to evaluate behavior or use a search engine to manually gather samples for population subsets [30].

4.1. Image Data and Model

Data. We use OpenImages [22] images as our base image set. As our user study focuses on objects and entities rather than scenes, we extract bounding images around annotated objects in the dataset. We apply non-maximum suppression to the ground-truth boxes to reduce redundant overlap-

ping detections, then extract a square box with side length $1.1 * max(bbox_h,\ bbox_w)$ centered on the detection to capture both the detection and its context. Detections on image borders are padded with the mean RGB value of the dataset. Detections smaller than 64^2 pixels and those which capture a region bounded by another detection from it's parent in the class hierarchy (*e.g.*, a nose detection on a face) are discarded. In total, this yields a dataset of $I_a = 8.1$ million images. Neither VL*Slice* or ListSort use the image labels.

Model-of-Interest. We use CLIP [29] as a representative vision-and-language alignment model – using the common variant utilizing a vision transformer [12] with sequence length sixteen from HuggingFace Transformers [38].

4.2. Protocol

We perform a between-subjects study (n=22) for VL*Slice*. Each participant is randomly assigned to use either the VL*Slice* or ListSort interface and are instructed to complete two tasks using their assigned interface.

Participants. We recruited 22 participants using departmental mailing lists and word of mouth. Five self-identified as women, fifteen as men, and two as non-binary or other genders. The average age of participants was 27 years old, three were most recently enrolled or completed an undergraduate degree, and 19 a graduate degree. Participation was limited to people who have taken three or more AI/ML courses or have at least two years of professional experience in AI/ML (including graduate studies), and are 18 years old or older. Each session had only one participant and all participants joined remotely via video call. All participants were compensated with a $15 gift card upon completion.

Protocol. Each participant takes approximately one-hour to complete the study. First, they are presented with a pre-study questionnaire to collect demographic information, familiarity with vision-and-language tasks in machine learning, and prior experience using other tools for analyzing their models results and behaviors. After completing the pre-study questionnaire, they are introduced to the tasks and given a guided demo of the selected interface with the query C_b = "*A photo of a car*", C_a = "*A photo of a fast car*". The study facilitator first demonstrates how to construct slices with the interface for the participant, then they switch roles and the participant demos the same task back to the facilitator. Once the participant is comfortable with the interface, they are given fifteen minutes to complete each of their two tasks and asked to "think aloud" while doing so. Finally, the participant completes a post-study questionnaire collecting twelve 7-point Likert-scale ratings and three free-form responses evaluating their experience with the interface.

Tasks. Participants are given two tasks: (1) C_b = "*A photo of a person*", C_a = "*A photo of a CEO*"; and (2) C_b = "*A photo of a house*", C_a = "*A photo of a nice house*" with $k = 3,000$ for both. For each, participants are instructed to discover as many slices as possible while attempting to adhere to the desirable properties given in Sec. 1. They are informed that these slices should contain visually coherent images with consistent response to the augmented caption.

At the end of each task, we save a snapshot of the slices created by the participant. This snapshot contains the names of each slice, what images were added to it, and all images included in the working set for the query. After each session, recordings of the study are reviewed to transcribe comments made by participants and slices captured by them are manually coded into higher-order categories.

	ListSort				VL*Slice*			
	Slices	# Img.	F1	Missed	Slices	# Img.	F1	Missed
Person/CEO	4.3	107	.42	90	**4.6**	**141**	**.59**	**54**
House/Nice	3.6	87	.20	**41**	**5.1**	**211**	**.46**	70

Table 1: Average number of slices identified with average total images cataloged into those slices, F1 slice coherency, and missed images representation metrics, aggregated by participant. **Bold** is better. Participants assigned VL*Slice* capture more images with higher coherency in both tasks.

5. Quantitative Results

Size and Number of Slices. Using the participant snapshots in each task, we evaluate the number of slices identified and the total number of images categorized into them by participants in Tab. 1. We find VL*Slice* outperforms ListSort in all cases. We hypothesize causes for the difference between our task results in the discussion (Sec. 6). To assess statistical significance, we fit linear mixed effect regression models (`lmer`) of the form: $y = w * \text{Interface} + \beta_{\text{Task}}$ where y is the measured outcome, Interface is a binary variable indicating VL*Slice* or ListSort, w estimates the effect strength of the interface, and β_{Task} is a per-task intercept modeled as a random effect. Under this model, VL*Slice* results in 84.58 more images (p=0.017) and 0.5955 more slices (p=0.295). This suggests using VL*Slice* yields statistically significantly **larger** image sets than the baseline.

Coherency. To evaluate visual coherency of slices captured by participants, we have annotators perform an outlier detection task. For each participant-collected slice, we subsample eight images and randomly select zero-to-two of those images to replace with outliers. Outliers are randomly sampled from other slices captured by the participant and are constrained to images with visual similarity to the slice centroid within one positive standard deviation of the mean of similarities for all candidate images. This helps prevent trivial outliers and ensures that the participant had consid-

ered the outlier images during slice formation. For slices with fewer than eight images, no subsampling is performed and slices with fewer than two images are excluded from the analysis. Annotators are then prompted that each slice contains 0-2 outliers and are asked to select them. We compute F1 over all slices for each annotator. Again using a linear mixed effect regression model of the same form, we find that VL*Slice* results in an increase in 0.215 F1 score (p=0.006). This suggests VL*Slice* leads to statistically significantly **greater coherence** than the baseline.

Representativeness. We measure representation by asking annotators to identify images that potentially should have been included in each slice but were missed by participants. For each slice, we measure the similarity between the slice centroid and each image in the working set that was not included in that slice. We then subsample 50 of the 100 most similar images. We display all images captured in the slice, the participant's description of the slice, and the subsampled similar images to annotators and ask that they select all images that should have been included in the slice. Again using a linear mixed effect regression model, we find VL*Slice* reduces the average number of missed images by 2.1 (p=0.35) but this result is not significant at 95% confidence. As VL*Slice* slices are both larger and more coherent than the baseline (and thus capture more images relevant to the slice), we suspect the non-significance of this result may reflect a lack of sensitivity in this study.

User Ratings. We evaluate participants' average response to Likert-scale post-questionnaire ratings in Tab. 2. Participants score VL*Slice* more favorably that ListSort in 10 of 12 questions with both lower scoring cases being relegated to simplicity of learning and using the interface, an expected result considering the short tutorial period during the study and comparably substantial feature set of VL*Slice* against ListSort. Nine of ten questions scoring higher on average for VL*Slice* are measured as statistically significant at 95% confidence under a Mann-Whitney U test.

Question	ListSort	*VLSlice*
Easy to learn how to use	**6.27**	5.55
Easy to use	**6.00**	5.55
Confident when using the tool	5.45	**5.73**
Enjoyed using the tool	5.27	**6.45***
Would like to use again	4.73	**6.45***
Image sets capture intended concept	4.36	**5.36***
Helpful for finding new behavior	5.09	**6.55***
Helpful for confirming behavior	4.82	**6.27***
Easy to build sets of images	5.00	**6.27***
Easy to discover additional images	4.82	**6.36***
My image sets are coherent	5.09	**5.64***
My image sets capture systemic bias	5.00	**6.27***

Table 2: Participants' average 7-point Likert-scale rating for each interface. VL*Slice* outscores ListSort in 10 of 12 questions. * indicates statistical significance at 95% confidence in a one-sided Mann-Whitney U test.

6. Qualitative Results

Below, we highlight several observations from the user study, providing insights about how people use VL*Slice*.

VL*Slice* users discover more abstract slices. Mapping slices discovered by participants to higher-level categories reveals trends in the types of relationships typically identified while using each interface. Participants assigned to ListSort more frequently discover slices capturing visual concepts which are easy to identify from low-level visual features that require little, and often pre-attentive, visual processing. These slices are frequently based in color (*e.g.*, "black and white"), structure (*e.g.*, "truncated subject"), and photographic qualities (*e.g.*, "blurry"), accounting for 17% (0% VL*Slice*) of all concepts identified in the Person/CEO task and 45% (14% VL*Slice*) of the House/Nice House task for ListSort. In contrast, participants leveraging VL*Slice* capture slices relating to historically gendered features, skin tone, and age in 62% (34% ListSort) of Person/CEO task concepts. For the House/Nice House task, VL*Slice* participants identify concepts relating to housing density, cultural cues, and features indicative of wealth in 33.9% (7.5% ListSort) of all concepts. We note that these choices in slices are emergent and that participants were not instructed to search for any specific or socially relevant visual concepts. We show sample slices created by participants using VL*Slice* for the Person/CEO task in Fig. 4. Additional examples for both tasks are provided in the appendix.

VL*Slice* promotes iterative refinement of slices. We find that **all** participants assigned the VL*Slice* interface engage in iteratively improving the quality of cluster recommendations by utilizing a feedback loop. Participants typically first identify a small number of relevant images from the cluster display and add it to a slice to bootstrap recommendations. Pointing out three neighboring clusters, participant P5 stated *"Some of these look a lot like houses in the neighborhoods I grew up in, lower-income in India specifically"* and selected a small subset of images from each cluster continuing to say *"I can see that it picks up on some of the visual cues individually, but struggles to put them all together"* before adding his selection to a slice and expanding the *similar clusters* display. The most similar clusters provided additional relevant samples and, as P5 continued to update the slice, was satisfactory in support for discovering

"masculine glasses" $\Delta C > 0$

"people of color" $\Delta C < 0$

Figure 4: Example slices created by participants for the Person/CEO task with VL*Slice*. In the "masculine glasses" slice (top), the participant identified that people wearing glasses with larger features or facial hair have a positive ΔC, indicating a CEO-like bias. In contrast, the "people of color" (bottom) slice has a negative ΔC, indicating bias against people with darker skintones being CEO-like.

an otherwise difficult to identify set of images. Likewise, participant P25 stated *"I liked how I could further refine my sets of images. This made it easy to quickly build sets with similar attributes,"* with respect to his cluster recommendations. We anticipate this workflow as the source of the increase in images captured as reported in Tab. 1.

Counterfactuals and correlation plots improve coherency and confidence. The above iteration is frequently followed by using counterfactuals to diagnose if the participant's slice has fallen into a systemic positive or negative subset of the visual concept they wish to capture. Participants typically add counterfactual samples until exhausted. For slices with a large support and high variance in ΔC, some participants iteratively switch between similar and counterfactual images until both are exhausted of relevant samples. Participant P1 leverages the correlation plot commenting that *"I'm using the correlation to try and find outliers that I missed when looking through similar and counterfactual photos"* and that *"when I'm looking at this [correlation plot], I'm checking to see if the model really knows the concept I'm trying to capture"* continuing to clarify that they are looking for a steep regression to imply bias and proximal in-concept images most similar to the slices centroid for coherency. We speculate that this workflow helps to improve coherency as reported in Sec. 5.

VL*Slice* accommodates users regardless of their familiarity with bias dimension. VL*Slice* can successfully accommodate participants who are both familiar and unfamiliar with the bias dimension. When a participant is familiar with the dimension, they ask directed questions about behavior with respect to some preconceived notion. For example, participant P6 investigated the intersection of historically gendered features and skin tone in the Person/CEO task stating "*I have a few biases in mind that I'm already aware of and want to search for, so I'm going to [...] to see what the model thinks of them.*" When a participant is unfamiliar with the bias dimension, they seek support from the interface for prompting visual concepts. Several participants find that the initial clustering primes them to investigate different visual concepts they would not have otherwise thought of. P6 stated that *"[filtering and sorting the cluster display] is nice because I'm not really sure what things a house would be biased against, but [VLSlice] primes me to explore some directions."* This flexibility makes VL*Slice* an effective tool for both directed bias search and discovery.

ListSort result in unfounded confidence. Although rated lower than VL*Slice*, participants assigned ListSort are still very confident overall in the coherency of slices they capture (Tab. 2). However, our quantitative results (Sec. 5) show that this is not the case. The common methodology of analysis may be leading participants to unsubstantiated conclusions about their model behavior. Specifically, some participants assigned ListSort identify a visual concept with many neighboring samples in the interface, select all those samples for a slice, then continue to a new slice. Examining slices discovered in this case reveals that they often capture a subset of the visual concept targeted by the participant. For example, many participants label their slice as "*formal wear*" but captures only images of masculine presenting people wearing suits. Conversely, participant P17 was assigned VL*Slice* and arrived at a similar result, but after using the provided exploratory tools (*e.g.*, counterfac-

tuals) determined that he was capturing the concept "men in suits" instead, proceeding to search for feminine formal apparel as a second distinct slice.

7. Limitations

Choosing an appropriate top-k. Users are faced with an implicit sensitivity-specificity trade-off for identifying the domain of subjects with baseline caption C_b through their choice in k for working set filtering. A small k has high specificity for filtering to this domain but poor sensitivity, potentially excluding informative samples. Conversely, a large k has low specificity and high sensitivity, potentially capturing irrelevant samples which will be out-of-distribution for the ΔC metric. We hope for future work to explore interactive processes at the working set boundary to intelligently select an appropriate value of k.

Working-set level model bias. While VL*Slice* allows users to discover biases falling within the working set of samples, it is unable to discover biases along a subject the model is unable to identify or where the bias presented in C_a correlates with a bias against the subjects identified in C_b. For example, given a model which is unable to effectively count, and baseline caption C_b = "*A photo of two people*", the model will fail to identify the subject domain. If given the baseline caption C_b = "*A photo of a meal*", augmented caption C_a = "*A photo of a healthy meal*", and a model trained from image-caption pairs scrapped from the web, then the notion of both what a meal and healthy meal are will likely be confounded with western cultural norms and thus unable to capture working set examples for evaluating the bias term "*healthy.*" We therefore advise that slices identified with VL*Slice* have high precision and are likely to be representative of a biased notion learned by the model, but potentially poor recall in either case described above and should not be used to argue the absence or non-existence of a bias.

Behavior with strongly biased subgroups. In the case of a model with strong bias towards some subgroup, but that is still effective for capturing that subgroup in the working set (*e.g.* feminine presentation in the Person/CEO task), there are two ways we may hypothesize VL*Slice* to change. First, we suspect the Ⓑ **Explore** step cluster presentation to be more likely to bifurcate along subgroup dimensions, forming additional clusters which capture the subgroup with high magnitude ΔC. Second, during the Ⓑ **Refine** step, we suspect these bifurcated clusters to be highly ranked within counterfactual cluster recommendations. For example, participants studying people wearing glasses in the Person/CEO task frequently found the subgroup was bifurcated by gender presentation, then discovered the two cluster components using counterfactual recommendations. If model bias is orthogonal to the biases targeted for evaluation, increased user effort may be needed during the refine step to guide the model away from recommendations capturing the disruptive bias instead of the one targeted by the user. For example, by searching through more counterfactuals and bootstrapping VL*Slice* recommendations for additional steps.

Computational complexity for joint encoders. As presented, VL*Slice* is limited to models which compute independent representations of language and imagery, which are used to compute similarity for clustering and ΔC calculations efficiently. Hypothetically, these same similarities could be computed as the output of joint encoder models (*e.g.* ViLBERT [24]), but at high computational expense.

8. Conclusion

In this work, we proposed VL*Slice*, an interactive system to discover slices from unlabeled collections of images. We conducted a between-subjects user study to evaluate the effectiveness of VL*Slice* against common methodologies for identifying model behavior. The results indicate that VL*Slice* outperforms the baseline for the number of images captured and slice coherency in both tasks. Additionally, participants rate it more favorable than the baseline in 10 of 12 Likert-scale questions describing usability and user confidence in desirable slice properties. We discuss the results of the study and find VL*Slice* to better support user workflows and promote discovering high quality abstract slices.

References

[1] Abeba Birhane, Vinay Uday Prabhu, and Emmanuel Kahembwe. Multimodal datasets: misogyny, pornography, and malignant stereotypes. *arXiv:2110.01963 [cs]*, 2021. 2

[2] Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. Language (technology) is power: A critical survey of "bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, Online, July 2020. Association for Computational Linguistics. 2

[3] Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam Tauman Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *NeurIPS*, 2016. 2

[4] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *FAT*, 2018. 2

[5] Ángel Alexander Cabrera, Marco Tulio Ribeiro, Bongshin Lee, Rob DeLine, Adam Perer, and Steven M Drucker. What did my ai learn? how data scientists make sense of model behavior. *ACM Transactions on Computer-Human Interaction*, 2022. 3

[6] Ángel Alexander Cabrera, Will Epperson, Fred Hohman, Minsuk Kahng, Jamie H. Morgenstern, and Duen Horng Chau. Fairvis: Visual analytics for discovering intersectional bias in machine learning. *2019 IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2019. 3

[7] V. Chen, Sen Wu, Zhenzhen Weng, Alexander J. Ratner, and C. Ré. Slice-based learning: A programming model for residual learning in critical data slices. *NeurIPS*, 2019. 3

[8] Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Alexander Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, Alexander Kolesnikov, Joan Puigcerver, Nan Ding, Keran Rong, Hassan Akbari, Gaurav Mishra, Linting Xue, Ashish Thapliyal, James Bradbury, Weicheng Kuo, Mojtaba Seyedhosseini, Chao Jia, Burcu Karagol Ayan, Carlos Riquelme, Andreas Steiner, Anelia Angelova, Xiaohua Zhai, Neil Houlsby, and Radu Soricut. Pali: A jointly-scaled multilingual language-image model. In *arXiv*, 2022. 1, 2

[9] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European conference on computer vision*, 2020. 1

[10] Yeounoh Chung, Tim Kraska, Neoklis Polyzotis, Ki Hyun Tae, and Steven Euijong Whang. Automated data slicing for model validation: A big data - ai integration approach. *IEEE Transactions on Knowledge and Data Engineering*, 2020. 1, 3

[11] 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. *ACM FAccT*, 2022. 1, 2, 3

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929, 2021. 6

[13] Conor Dougherty. Google photos mistakenly labels black people 'gorillas'. *New York Times*, 2015. 2

[14] Sabri Eyuboglu, Maya Varma, Khaled Saab, Jean-Benoit Delbrouck, Christopher Lee-Messer, Jared Dunnmon, James Zou, and Christopher Ré. Domino: Discovering systematic errors with cross-modal embeddings. *arXiv:2203.14960 [cs]*, 2022. 1, 2, 3

[15] Lisa Anne Hendricks, Kaylee Burns, Kate Saenko, Trevor Darrell, and Anna Rohrbach. Women also snowboard: Overcoming bias in captioning models. In *NeurIPS*, 2016. 1, 2

[16] Yusuke Hirota, Yuta Nakashima, and Noa Garcia. Quantifying societal bias amplification in image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13450–13459, June 2022. 2

[17] Kenneth Holstein, Jennifer Wortman Vaughan, Hal Daumé III, Miro Dudik, and Hanna Wallach. Improving fairness in machine learning systems: What do industry practitioners need? In *Proceedings of the 2019 CHI conference on human factors in computing systems*, pages 1–16, 2019. 1

[18] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning (ICML)*, 2021. 1, 2

[19] Kimmo Karkkainen and Jungseock Joo. Fairface: Face attribute dataset for balanced race, gender, and age for bias measurement and mitigation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1548–1558, 2021. 1

[20] Daniel Keim, Gennady Andrienko, Jean-Daniel Fekete, Carsten Gorg, Jorn Kohlhammer, and Guy Melançon. Visual analytics: Definition, process, and challenges. *Lecture notes in computer science*, 4950:154–176, 2008. 3

[21] Michael P. Kim, Amirata Ghorbani, and James Zou. Multiaccuracy: Black-box post-processing for fairness in classification. *arXiv:1805.12317 [cs, stat]*, 2018. 1, 2, 3

[22] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020. 5

[23] Hanseung Lee, Jaeyeon Kihm, Jaegul Choo, John Stasko, and Haesun Park. ivisclustering: An interactive visual document clustering via topic modeling. In *Computer graphics forum*, volume 31, pages 1155–1164. Wiley Online Library, 2012. 3

[24] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 2019. 9

[25] Jiasen Lu, Vedanuj Goswami, Marcus Rohrbach, Devi Parikh, and Stefan Lee. 12-in-1: Multi-task vision and language representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 1

[26] Ryan Mac. Facebook apologizes after a.i. puts 'primates' label on video of black men. *New York Times*, 2021. 2

[27] Youssef Mohamed, Faizan Farooq Khan, Kilichbek Haydarov, and Mohamed Elhoseiny. It is okay to not be okay: Overcoming emotional bias in affective image captioning by contrastive data collection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21263–21272, June 2022. 2

[28] Safiya Noble. *Algorithms of Oppression: How Search Engines Reinforce Racism*. New York University Press, Cambridge MA, 2018. 1, 2

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*. PMLR, 2021. 1, 2, 3, 6

[30] Candace Ross, Boris Katz, and Andrei Barbu. Measuring social biases in grounded vision and language embeddings. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 998–1008, Online, June 2021. Association for Computational Linguistics. 1, 2, 5

[31] Svetlana Sagadeeva and Matthias Boehm. Sliceline: Fast, linear-algebra-based slice finding for ml model debugging. *ICMD*, 2021. 1, 3

[32] Nimit S. Sohoni, Jared A. Dunnmon, Geoffrey Angus, Albert Gu, and Christopher Ré. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. *arXiv:2011.12945 [cs]*, 2022. 1, 2, 3

[33] Tejas Srinivasan and Yonatan Bisk. Worst of both worlds: Biases compound in pre-trained vision-and-language models. In *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*, pages 77–85, 2022. 1, 2

[34] Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth Belding, Kai-Wei Chang, and William Yang Wang. Mitigating gender bias in natural language processing: Literature review. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1630–1640, Florence, Italy, July 2019. Association for Computational Linguistics. 2

[35] Ki Hyun Tae and Steven Euijong Whang. Slice tuner: A selective data acquisition framework for accurate and fair machine learning models. *SIGMOD Conference*, 2021. 3

[36] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2019. 1

[37] Angelina Wang, Solon Barocas, Kristen Laird, and Hanna Wallach. Measuring representational harms in image captioning. *ACM FAccT*, 2022. 1

[38] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, Oct. 2020. Association for Computational Linguistics. 6

[39] Zhenge Zhao, Panpan Xu, Carlos Scheidegger, and Liu Ren. Human-in-the-loop extraction of interpretable concepts in deep learning models. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):780–790, 2021. 3