TEVI: Text-Conditioned Editing of Visual Representations via Sparse Autoencoders for Improved Vision-Language Alignment

Sweta Mahajan[*,1], **Sukrut Rao**[*,1], **Jiahao Xie**[1], **Alexander Koller**[1,2], **Bernt Schiele**[1]
[1]Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany
[2]Department of Language Science and Technology, Saarland University, Saarbrücken, Germany
{sweta.mahajan,sukrut.rao,schiele}@mpi-inf.mpg.de
[*]Equal contribution

Abstract

Vision-language models such as CLIP are highly useful for diverse tasks due to their shared image-text embedding space. Despite this, the image and text embeddings are often poorly aligned, affecting downstream performance. Recent work has hypothesized that this can be attributed to an information imbalance: images contain more information than their captions describe. In this work, we propose TEVI, a framework that uses captions as a signal for what to retain from image embeddings. Specifically, we use sparse autoencoders to disentangle image embeddings and train a masking module to selectively reconstruct the embedding based on a given caption. In a controlled setup with synthetic captions, we show that TEVI is effective at preserving caption-described attributes while discarding others. We find that this extends to CLIP models trained on natural images, where TEVI learns to mask meaningfully and allows retrieval based on conditioning. Finally, we use TEVI to achieve improved retrieval performance across coarse-grained and fine-grained benchmarks. Code available at https://github.com/neuroexplicit-saar/TEVI.

1 Introduction

Vision-language models such as the CLIP family (Radford et al., 2021; Ilharco et al., 2021; Zhai et al., 2023; Tschannen et al., 2025) are trained using a contrastive loss to align images and text to a shared embedding space[1]. Such models have been widely successful for a variety of multimodal applications, such as zero-/few-shot classification and cross-modal retrieval. Their embeddings serve as a vital bridge between vision and language for generative models such as text-to-image diffusion models (Razzhigaev et al., 2023) and large multimodal models (LMMs) (Liu et al., 2023). However, it has been shown that the learnt embedding space suffers from a modality gap (Liang et al., 2022), where image and text embeddings lie in different regions of the embedding space, often leading to poor downstream performance (Liang et al., 2022; Eslami and de Melo, 2025; Schrodi et al., 2025).

While various possible causes have been discussed (Eslami and de Melo, 2025; Schrodi et al., 2025), a recent work (Schrodi et al., 2025) hypothesizes that this is caused by an *information imbalance*—the fact that images contain more information than their corresponding captions, which forces models to push their embeddings apart when minimizing training loss—and shows evidence for this hypothesis via systematically controlled experiments on the synthetic MAD dataset. However, despite finding that only a few embedding dimensions characterize the gap, removing these dimensions also significantly degrades performance. Since CLIP dimensions do not directly correspond to concepts, this may not map to selectively removing information as the hypothesis suggests.

In this work, we explore whether captions could instead determine what information should be preserved, by using text as a signal to modify CLIP image embeddings to preserve only what the caption describes. Specifically, we use sparse autoencoders (SAEs) (Bricken et al., 2023; Cunningham et al., 2024) to decompose image embeddings into constituent concepts, and train a conditioning module that selects SAE latents to be used for reconstruction based on the conditioning. Through a controlled setup, we show that our proposed approach, text-conditioned editing of visual representations for image-text alignment (**TEVI**) (Fig. 1, left), can learn to preserve attributes presented in the text while discarding information about other attributes and improve cross-modal alignment and retrieval ordering (Fig. 5).

[1]Throughout this work, we use "vision-language model" to refer to CLIP-like models with aligned image-text encoders, and not autoregressive multimodal models that generate text.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 44358–44382
October 24-29, 2026 ©2026 Association for Computational Linguistics

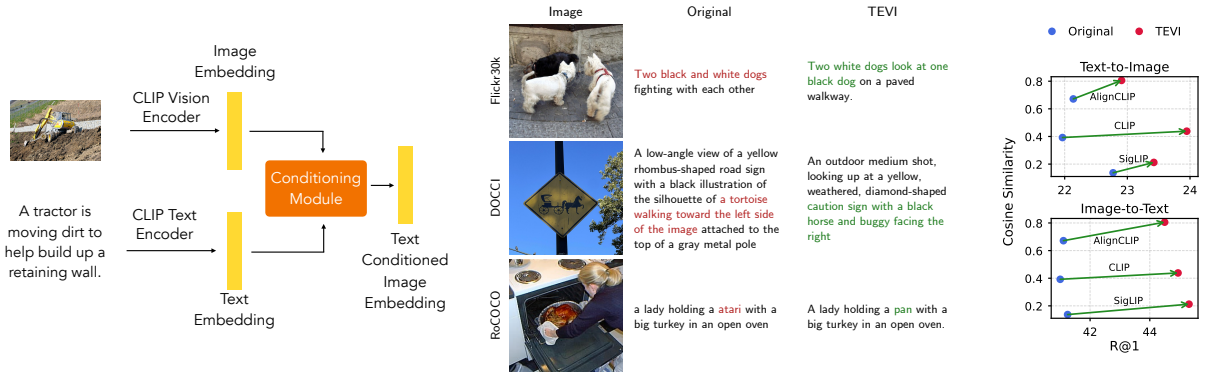


Fig. 1: **TEVI: using captions to edit image embeddings.** *Left:* An overview of our approach. We use text captions as a signal to modify image embeddings from CLIP (Radford et al., 2021). For details, see Fig. 2. *Middle:* Qualitative examples of top texts retrieved for an image by CLIP and TEVI. *Right:* TEVI helps improve cross-modal alignment as well as downstream retrieval performance. We report mean performance across datasets, for full results see Tabs. 2 and B1 and Fig. B2.

We then apply TEVI to CLIP (Radford et al., 2021; Ilharco et al., 2021), SigLIP (Zhai et al., 2023), SharedCLIP, and AlignCLIP (Eslami and de Melo, 2025) models trained on CC12M (Changpinyo et al., 2021). We find that TEVI learns to selectively and meaningfully mask latents based on the conditioning, which helps retrieval with known attributes (Fig. 6). Further, TEVI improves (Fig. 1, middle and right) image-to-text and text-to-image retrieval performance across coarse-grained (MS COCO (Lin et al., 2014), Flickr30k (Plummer et al., 2015)) and fine-grained (DOCCI (Onoe et al., 2024), IIW (Garg et al., 2024)) benchmarks, by serving as an effective reranking method of retrieved candidates from the base model. Using the RoCOCO (Park et al., 2024) benchmark, which augments captions with linguistically perturbed alternatives, we also show that TEVI may help improve robustness against such perturbations. On the whole, we find that having a concept basis makes the information imbalance controllable.

In summary, our **contributions** are as follows:

- CLIP-Guided SAEs (CG-SAEs), a controlled setup with latents that encode predefined text concepts, to study their utility for preserving targeted concepts.
- TEVI, a framework that uses text as a signal to selectively retain caption-described content in image embeddings, via a learnt mask over sparse autoencoder latents.

We use CG-SAEs to show a proof-of-concept of our proposed approach through controlled experiments on MAD (Schrodi et al., 2025). We then use TEVI with MAD to show that it leads to improved alignment and retrieval, and then show that this extends to CLIP models trained on natural images.

2 Related Work

Vision-Language Models (VLMs) (Radford et al., 2021; Zhai et al., 2023; Ilharco et al., 2021; Yu et al., 2022; Jia et al., 2021; Tschannen et al., 2025) learn a joint aligned embedding space between images and texts. They typically consist of separate unimodal image and text encoders that each provide an embedding, and are trained using contrastive losses so that embeddings of similar image-text pairs are placed close to each other and dissimilar pairs are placed apart. Such models are useful for a diverse set of multimodal tasks such as cross-modal retrieval and zero-shot classification. Their embeddings are also used as a bridge between vision and language for text-to-image diffusion models (Razzhigaev et al., 2023) and large multimodal models (LMMs) (Liu et al., 2023). In this work, we focus on CLIP-family models.

Modality Gap (Schrodi et al., 2025; Liang et al., 2022; Eslami and de Melo, 2025; Mistretta et al., 2025; Shi et al., 2023) is a phenomenon observed in trained VLMs, where, despite the contrastive training objective, image and text embeddings lie in different regions of the embedding space. While initially attributed to the 'cone effect' at initialization (Liang et al., 2022), recent work (Schrodi et al., 2025) showed that a likely cause is the information imbalance between the two modalities, i.e., images contain more information than is described in their corresponding caption, which forces the model to push apart their embeddings to reduce the contrastive loss. Existing post hoc approaches to reduce the modality gap (Liang et al., 2022; Schrodi

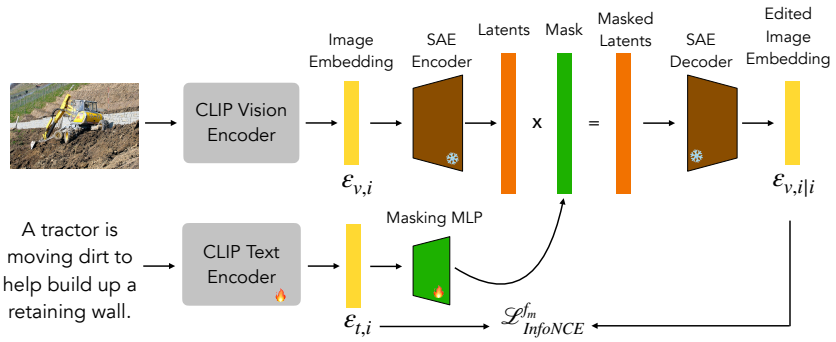


Fig. 2: **Our proposed TEVI framework for obtaining text-conditioned image embeddings.** We train a TopK SAE (Gao et al., 2025) over CLIP image embeddings, and then use an MLP trained using the InfoNCE loss to learn a mask over the SAE latents to obtain conditioned image embeddings. For details, see Secs. 4 and 5.

et al., 2025) have been shown to come at the cost of degraded performance. Recently, AlignCLIP (Eslami and de Melo, 2025) proposed to use an intra-modal separation loss during CLIP training to improve alignment and downstream performance. In contrast, we explicitly control the information imbalance post hoc by using text as a signal to edit image embeddings, and show that this is complementary to methods such as AlignCLIP.

Cross-Modal Conditioning methods such as SmartCLIP (Xie et al., 2025b), FLAIR (Xiao et al., 2025), and FILIP (Yao et al., 2022) produce image embeddings conditioned on text for improved fine-grained retrieval, at the cost of a small increase in retrieval time. Concurently, SteerViT (Ruthardt et al., 2026) conditions a frozen vision encoder on text by inserting cross-attention layers into its blocks, which steers the representation towards the objects that a prompt names. An orthogonal line of work (Xie et al., 2025a; Asokan et al., 2025) performs fine-grained alignment at training without conditioning at inference. Among the former methods, all except SmartCLIP and SteerViT train models from scratch. In contrast, TEVI is post-hoc: it adds a small module on top of pretrained CLIP, with the vision encoder kept frozen and only the text encoder fine-tuned. Similar to SmartCLIP, TEVI learns a mask over vision embeddings conditioned on text, but in contrast, TEVI masks disentangled SAE latents instead of raw representations.

Sparse Autoencoders (SAEs) (Bricken et al., 2023; Cunningham et al., 2024; Gao et al., 2025; Rajamanoharan et al., 2024a,b; Bussmann et al., 2025) are a popular mechanistic interpretability tool to disentangle activations learnt by a deep network into constituent human understandable concepts. While originally used in the context of LLMs (Bricken et al., 2023; Cunningham et al., 2024), SAEs have recently been used to decompose concepts from CLIP vision embeddings (Rao et al., 2024; Zaigrajew et al., 2025) for use in downstream tasks such as constructing concept bottleneck models (Rao et al., 2024). SAEs have also been used as a tool for steering models by performing edits to their latents (Farrell et al., 2024; Cywiński and Deja, 2025; Pach et al., 2025; Joseph et al., 2025). In contrast to these works which apply SAE-based interventions for interpretability or steering, we use them for text-conditioned editing aimed at improving cross-modal alignment.

3 Using SAEs to Edit Representations

In this section, we motivate our text-conditioned image editing approach. In Sec. 3.1, we first provide a brief overview on CLIP (Radford et al., 2021) and sparse autoencoders (SAEs) (Bricken et al., 2023). Then, to test whether SAEs can be used to edit representations, we describe a controlled test on a synthetic setup in Sec. 3.2 and present our findings in Sec. 3.3. We then present our proposed TEVI in Sec. 4, and in Sec. 5, we show how we extend it to real data.

3.1 Background

CLIP (Radford et al., 2021). Let $\mathcal{D} = \{(v_i, t_i)\}_{i=1}^{N}$ be a paired dataset of images v_i and their corresponding texts t_i. A CLIP model $M = (f_v, f_t)$ consists of a vision and text encoder respectively which provide corresponding embeddings $(\varepsilon_{v,i}, \varepsilon_{v,t})$, i.e. $\varepsilon_{v,i} = f_v(v_i) \in \mathbb{R}^d$ and $\varepsilon_{t,i} = f_t(t_i) \in \mathbb{R}^d$. The model is then trained with

a contrastive InfoNCE loss (Oord et al., 2018):

$$\mathcal{L}_{\text{InfoNCE}} = -\frac{1}{2N}\sum_{i=1}^{N}\left[\log\frac{e^{\hat{\varepsilon}_{v,i}\cdot\hat{\varepsilon}_{t,i}/\tau}}{\sum_{j=1}^{N}e^{\hat{\varepsilon}_{v,i}\cdot\hat{\varepsilon}_{t,j}/\tau}} + \log\frac{e^{\hat{\varepsilon}_{v,i}\cdot\hat{\varepsilon}_{t,i}/\tau}}{\sum_{j=1}^{N}e^{\hat{\varepsilon}_{v,j}\cdot\hat{\varepsilon}_{t,i}/\tau}}\right] \quad (1)$$

where τ is a learnable temperature hyperparameter and $\hat{x} = \frac{x}{\|x\|_2}$. Eq. (1) pulls embeddings of corresponding (positive) image-text pairs $(\varepsilon_{v,i}, \varepsilon_{t,i})$ close to each other and pushes embeddings of other (negative) image-text pairs $(\varepsilon_{v,i}, \varepsilon_{t,j}), i \neq j$ away from each other to learn a shared semantic embedding space.

Sparse Autoencoders (Bricken et al., 2023) consist of a linear encoder $W_E \in \mathbb{R}^{d\times d_1}$ and decoder $W_D \in \mathbb{R}^{d_1\times d}$ where typically $d_1 \gg d$. The encoder maps an input $x \in \mathbb{R}^d$ to latents $z = \text{ReLU}(W_E^T(x - b_{pre}) + b_E)$, where $b_{pre} \in \mathbb{R}^d, b_E \in \mathbb{R}^{d_1}$ are learnt biases and z is a sparse disentangled concept representation of x. The decoder then reconstructs x using z, i.e. $\tilde{x} = W_D^T z + b_{pre}$. The SAE is trained with a combination of a reconstruction loss $\mathcal{L}_{\text{recon}} = \|x - \tilde{x}\|_2$ and sparsity loss $\mathcal{L}_{\text{sparse}}$, i.e.

$$\mathcal{L}_{\text{SAE}} = \mathcal{L}_{\text{recon}} + \lambda_{\text{SAE}}\mathcal{L}_{\text{sparse}}, \quad (2)$$

where λ_{SAE} is a hyperparameter. We use $\mathcal{L}_{\text{sparse}} = \text{TopK}(z)$ (Gao et al., 2025) where $\text{TopK}(\cdot)$ selects the top K activated latents in z for hyperparameter K.

3.2 Controlled Setup for Evaluating Representation Editing

To understand whether subselecting SAE latents before reconstruction can effectively edit embeddings, we construct a controlled setup using synthetic data with known attributes, where we induce each SAE latent to represent a predefined attribute. In contrast to a typical setup where SAEs automatically learn to disentangle concepts, this is designed to allow for targeted editing, and we refer to it as CLIP-guided SAE (CG-SAE). Let $C = \{c_j\}_{j=1}^{S}$ be a set of text concepts, and let (W_E, W_D) be an SAE to be trained on image embeddings of CLIP. Then, for an embedding $\varepsilon_{v,i}$,

$$z_i = \text{TopK}(\text{ReLU}(W_E^T(\varepsilon_{v,i} - b_{pre}) + b_E)) \quad (3)$$

$$\widetilde{\varepsilon_{v,i}} = W_D^T z_i + b_{\text{pre}} \quad (4)$$

where $z_i = [z_{i,r}]_{r=1}^{d_1}$ such that elements in z_i outside the top K elements are zeros.

Recent work (Rao et al., 2024) showed that latents $z_{i,r}$ could be assigned meaningful concept names in $c_q \in C$ post hoc by selecting the text embedding that is closest to their corresponding decoder weight vector $W_{D,r}$, i.e.

$$q = \arg\max_j \cos(W_{D,r}; c_j)\,. \quad (5)$$

Inspired by this, for our conditioning, we propose to attempt the opposite—given C, we fix the rows of the decoder weights W_D to be text embeddings of concepts $c_j \in C$, i.e.

$$W_{D,r} = \hat{\varepsilon}_{t,c_r} \quad (6)$$

where $1 \leq r \leq S$ and $d_1 = S$, and then only train the SAE encoder.

Once trained, given an attribute set $C_A = \{c_q\} \subseteq C$, we select latents *not* present in the text, i.e. $\bar{C} = C \setminus C_A$ and set their activations to zero, i.e.

$$z_{p|q} = [z_p]_{z_{p,c_{q'}}=\vec{0}},\ \forall c_{q'} \in \bar{C} \quad (7)$$

The edited image embedding of an image v_p conditioned on text t_q is then given by $\widetilde{\varepsilon_{v,p|q}}$, where

$$\widetilde{\varepsilon_{v,p|q}} = W_D^T z_{p|q} + b_{\text{pre}}\,. \quad (8)$$

We evaluate the following: **(R1)** if each CG-SAE latent indeed encodes the concept assigned to it, and **(R2)** if masking out specific attributes removes information about that attribute from the reconstructed embedding from the SAE. We present our findings in Sec. 3.3.

3.3 Results

Experimental Setup. We use the synthetic MAD dataset (Schrodi et al., 2025), which consists of MNIST digits with colors and morphological transforms. Every image is characterized by six attributes—digit, color, thickthinning, swelling, scaling, and fracture—each of which can be one among a set of predefined options, which are 26 in total. For example, the 'thickthinning' attribute for an image could be 'thickening', 'thinning' or 'nothickthinning', depending on the transform applied to the image. For each image, we create captions that contain the digit and three randomly selected attributes out of the remaining five to simulate information imbalance (Schrodi et al., 2025).

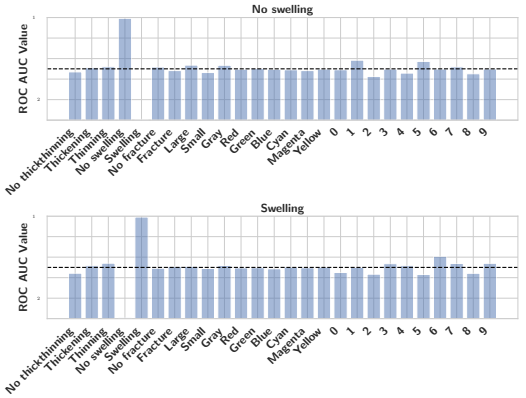


Fig. 3: **Attribute specificity of CG-SAE latents.** We plot the AUC for CG-SAE latents corresponding to the attribute values 'No swelling' and 'Swelling'. We find that the latents are highly attribute specific, which shows that our CG-SAE latents are highly disentangled, despite being assigned to a predefined concept. Full results in Appendix A.2.

Following (Schrodi et al., 2025), we train CLIP models (Radford et al., 2021; Ilharco et al., 2021) consisting of a 6-layer ViT (Dosovitskiy et al., 2021) as the vision encoder and a 6-layer transformer as the text encoder, with a shared embedding dimension of 18, for 200 epochs on using the AdamW (Loshchilov and Hutter, 2019) optimizer. Given the 26 possible attributes per image, the text encoder uses a tokenizer with a vocabulary of size 28, after accounting for the start and end tokens. Then, we train TopK SAE (Gao et al., 2025) on CLIP image embeddings with a latent dimension of 26 (equal to the number of attributes for the dataset). We assign each of the decoder weights as the normalized text embedding of one of the attribute values, so that each SAE latent represents one of the 26 attributes. For the SAE, we ablate across values of K, expansion factors, and learning rates, and select models based on reconstruction mean-squared accuracy and classification performance across the six attributes.

Concept Disentanglement (R1). We evaluate if our CG-SAE learns to disentangle image embeddings into the fixed set of predefined concepts C. In Fig. 3, we show the area under the receiver operating characteristic (ROC) curve (AUC) across images in the dataset for the two latents assigned to the 'swelling' attribute. We find that the latents are able to disentangle the attributes well, with the AUC being close to 1 for the attribute value the latent is assigned to, 0.5 for unrelated attributes, and 0 for attribute values anti-correlated to the assigned attribute (i.e. 'no swelling', for the 'swelling' latent). We also show qualitative examples of top activating images for a selection of latents in Fig. A1, and find that they are highly consistent, e.g. for the 'blue' latent, the top activating images are all digits of the color blue, while also being diverse in all other attributes.

Editing Image Representations (R2). We evaluate if masking latents corresponding to a single attribute discards information about that attribute from the reconstructed representations. To do this, we pick a single attribute 'thickthinning', and set the three latents corresponding to it ('thickening', 'thinning', 'nothickthinning') to zero. This is similar to conditioning on all attributes q except 'thickthinning' (Eq. (7)). We then obtain edited embeddings for all images, and perform classification on all six attributes based on the cosine similarity with the text embeddings of the classes within the attribute, similar to zeroshot classification. For example, for 'Thickthinning', given an edited embedding $\varepsilon_{v,p|q}$, $\text{Pred}_{\text{thickthinning}} = \arg\max \cos(\varepsilon_{v,p|q}; \varepsilon_{t,j})$, where $t_j \in \{\text{thickening}, \text{thinning}, \text{nothickthinning}\}$. We find (Fig. 4, left) that the classification accuracy of the edited embeddings on 'thickthinning' drops to near random chance while that of all other attributes remains close to that with the original embedding. This shows that our conditioning can be an effective way to remove information present in the image embeddings that is not in the text.

4 TEVI: Learning to Mask Latents based on Conditioned Text

In Sec. 3.2, we showed that we could construct CG-SAEs for encoding and editing representations given predefined synthetic concepts. However, this is a restrictive assumption, since CLIP models are typically open vocabulary and a predefined set of concepts is not available. As a result, in this section, we relax this assumption and propose TEVI, an approach to learn a mask over SAE latents z to obtain the text-conditioned image embeddings. An overview of our approach is shown in Fig. 2.

4.1 Optimization Objective

Specifically, we keep the CLIP vision encoder and a trained SAE frozen, and train a small multi-layer perceptron (MLP) network consisting of linear transforms with ReLU activations $f_m : \mathbb{R}^d \to \mathbb{R}^{d_1}$ that maps text embeddings $\varepsilon_{t,i}$ to a mask $m_i \in [0,1]^{d_1}$, i.e. $m_i = \sigma(f_m(\varepsilon_{t,i}))$ where $\sigma(\cdot)$ is the sigmoid function.

Then, the edited image embedding of an image

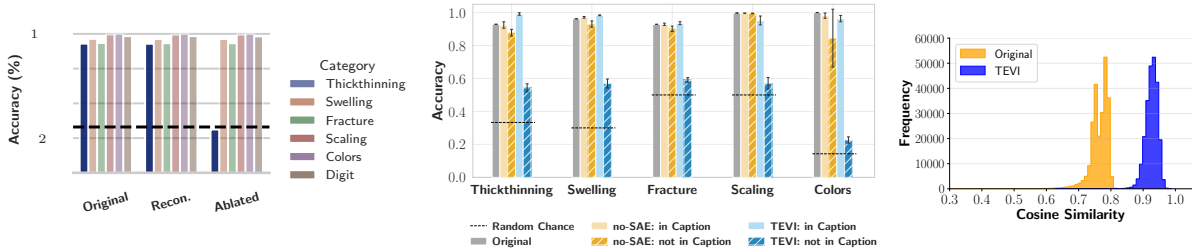


Fig. 4: *Left:* **Accuracy after ablating a single latent.** Accuracy for 'Thickthinning' drops to random chance (dotted line) when editing image embeddings to discard information about that attribute (right group), while the other attributes continue to maintain high accuracy (Sec. 3). *Middle:* **Effectiveness of conditioning.** Attributes present in the conditioning text are preserved in the edited embedding, while attributes that are absent are classified at close to random chance accuracy (Sec. 4), and the no-SAE baseline shows almost no effect. *Right:* **Impact on vision-language alignment.** Pairwise image-text cosine similarities between positive pairs increase on applying TEVI (Sec. 4).

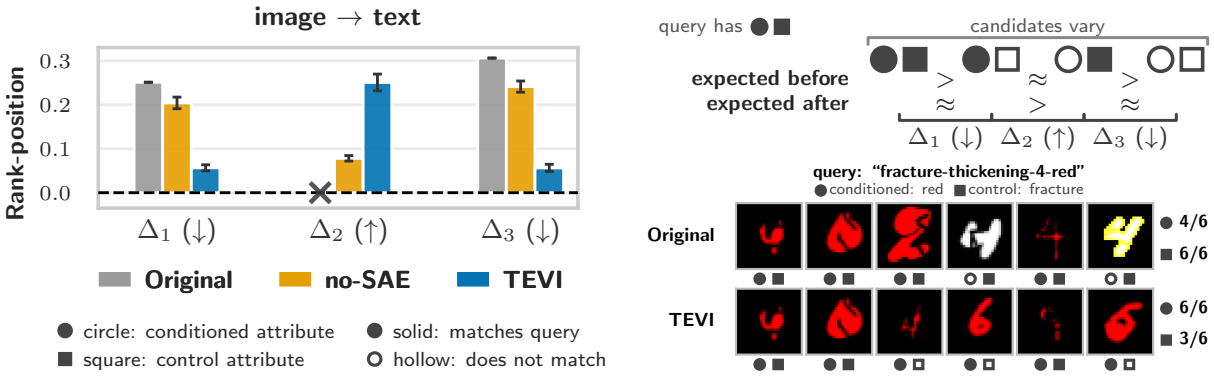


Fig. 5: **Conditioning reorders the retrieved set on MAD.** We condition on one attribute (●) and use another as control (■), see top right. With a query containing both ● and ■ and varying candidates, we expect Δ_1 and Δ_3 to fall towards zero and Δ_2 to grow, as the edited embedding should preserve ● and discard ■. Δ denotes the difference in mean rank position between two groups of candidates, where each rank is normalized between 0 and 1. *Left:* The rank-position differences Δ_1, Δ_2 and Δ_3 before and after conditioning, for I $\rightarrow$ T retrieval. TEVI moves all three in the expected direction, and significantly more than the no-SAE baseline. Full results in Fig. A6. *Right:* a T $\rightarrow$ I example, showing the top-6 retrieved images before and after conditioning on 'red'. The conditioned attribute is kept ($4/6 \rightarrow 6/6$) and the control attribute is dropped ($6/6 \rightarrow 3/6$).

v_p conditioned on text t_q is given by $\varepsilon_{v,p|q}$, where

$$\varepsilon_{v,p|q} = W_D^T(z_p \odot m_q) + b_{\text{pre}} \tag{9}$$

where $\odot$ is the element-wise product.

We train f_m and fine-tune the CLIP text encoder f_t using the InfoNCE loss (Eq. (1)) to pull the edited embeddings towards their conditioning texts and away from other texts, i.e.

$$\mathcal{L}^{f_m}_{\text{InfoNCE}} = -\frac{1}{2N}\sum_{i=1}^{N}\left[\log\frac{e^{\hat{\varepsilon}_{v,i|i}\cdot\hat{\varepsilon}_{t,i}/\tau}}{\sum_{j=1}^{N}e^{\hat{\varepsilon}_{v,i|i}\cdot\hat{\varepsilon}_{t,j}/\tau}} + \log\frac{e^{\hat{\varepsilon}_{v,i|i}\cdot\hat{\varepsilon}_{t,i}/\tau}}{\sum_{j=1}^{N}e^{\hat{\varepsilon}_{v,j|j}\cdot\hat{\varepsilon}_{t,i}/\tau}}\right]. \tag{10}$$

4.2 Results

Experimental Setup. We follow the setup from Sec. 3.3; however, we no longer fix the decoder weights of SAE and keep them learnable. We use 3-layer MLPs for the masking module. As a baseline, we train a mask on CLIP image embeddings.

Effectiveness of Conditioning. We evaluate the effectiveness of conditioning for TEVI on actual captions. First, we perform attribute-wise classification. Since each caption has a random subset of attributes present in the image (Sec. 3.3), we bin the attributes for each image depending on whether the caption contains it, and report the classification accuracy across the two bins in Fig. 4 (middle). We find that (blue) the classification accuracy on attributes *present* in the caption remains similar to the accuracy from the original embeddings, while the accuracy from attributes *absent* from the caption reach near random chance (dotted black line). In contrast, the no-SAE baseline (orange) shows almost no effect. Then, in Fig. 5, we evaluate the impact of conditioning on retrieval. We do this by picking a pair of attributes, selecting a query that contains both attributes, and conditioning on one while keeping the other as control. Before conditioning, candidates containing both attributes should match the strongest, and those that contain just one should match weaker but equally well. However, after conditioning, we expect the conditioned attribute to be matched stronger than the control attribute, whose information should be

		Δ			Classes	
	Exp.	Orig.	no-SAE	TEVI	no-SAE	TEVI
Δ_1	↓	0.118	0.119	**0.095**	32/62	**47/62**
Δ_2	↑	0.120	0.162	**0.195**	45/62	**57/62**
Δ_3	↓	0.203	0.221	**0.188**	18/62	**42/62**

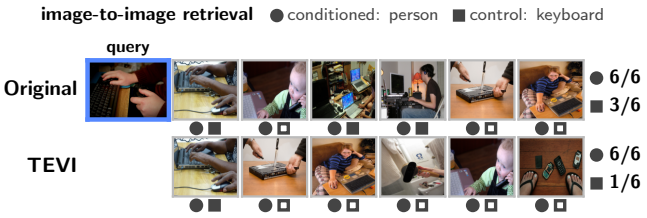


Fig. 6: **Conditioning helps retrieval on MS COCO captions given by attributes.** *Left:* Similar to Fig. 5, we compute rank-position differences for image → text retrieval, averaged over 62 object classes, with equal weight per class and 400 query images. We find that TEVI reorders as per the conditioning direction, and outperforms the no-SAE baseline, especially significantly in discarding information not present in the conditioning (Δ_1, Δ_3). The last two columns count how many of the 62 classes move in the expected direction, where TEVI consistently outperforms the baseline. *Right:* A qualitative example for image → image retrieval. The query image contains a person and a keyboard, and we condition on 'person'. The retrieved set keeps the person (circle, $6/6 \rightarrow 6/6$) and loses the keyboard (square, $3/6 \rightarrow 1/6$).

Table 1: **Ablating mask latents by ranking.** We zero the mask for m/128 latents and report DOCCI I → T R@1, where the baseline performance is 24.20. The latents are ranked in two ways: by their mask value, and by ρ, the correlation between a latent's mask value and the presence of its concept in the conditioning caption. We find that zeroing out top latents leads to the sharpest drops in performance for both rankings, which shows that important latents are ranked higher.

Ranked by	m	top	random	bottom
mask value	32	**8.62**	15.58	22.94
	64	**0.80**	7.24	16.64
grounding ρ	8	**18.0**	22.1	23.6
	16	**12.1**	20.4	22.8
	32	**5.2**	15.2	21.0

discarded from the representation (see Fig. 5, top right). We find that this holds when using TEVI, and significantly outperforms the no-SAE baseline. This shows that TEVI is effective in performing conditioning such that the edited embedding preserves information present in the text while discarding the remaining attributes.

Vision-language Alignment. Following Eslami and de Melo (2025), we also evaluate the pairwise image-text cosine similarities between positive image-text pairs before and after conditioning and the plots are shown in Fig. 4 (right). We find that for TEVI, the pairwise similarities increase significantly after conditioning (blue) as compared to before (orange).

5 Using TEVI with Natural Images

In Sec. 4, we showed the utility of our TEVI on a synthetic setup. In this section, we extend to CLIP models trained on natural images with the goal of improving alignment and retrieval performance.

5.1 Optimization Objective

Despite its effectiveness in the synthetic setup, the objective from Eq. (10)) only uses edited image embeddings $\hat{\varepsilon}_{v,i|i}$ conditioned on *their own* corresponding positive captions, i.e. it does not model conditioning on negative captions. However, image-to-text and text-to-image retrieval involve selecting from a set of candidates, which would require conditioning both positive and negative pairs. So, we modify the objective to incorporate negative conditioning during training:

$$\mathcal{L}^{f_m}_{\text{InfoNCE}} = -\frac{1}{2N}\sum_{i=1}^{N}\left[\log\frac{e^{\hat{\varepsilon}_{v,i|i}\cdot\hat{\varepsilon}_{t,i}/\tau}}{\sum_{j=1}^{N} e^{\hat{\varepsilon}_{v,i|j}\cdot\hat{\varepsilon}_{t,j}/\tau}} + \log\frac{e^{\hat{\varepsilon}_{v,i|i}\cdot\hat{\varepsilon}_{t,i}/\tau}}{\sum_{j=1}^{N} e^{\hat{\varepsilon}_{v,j|i}\cdot\hat{\varepsilon}_{t,i}/\tau}}\right] \quad (11)$$

where e.g. $e^{\hat{\varepsilon}_{v,j|j}}$ in the denominator of the first term from Eq. (10) is changed to $e^{\hat{\varepsilon}_{v,i|j}}$. Conditioning on negative captions is essential, without which the model performs poorly, which could be due to the fact that the model sees both positive and negative captions during inference. We discuss further in Appendix B.6.

5.2 Results

Experimental Setup. Following previous works (Eslami and de Melo, 2025; Goel et al., 2022; Mu et al., 2022; Li et al., 2022), we use the CC12M dataset (Changpinyo et al., 2021) to train a CLIP (Radford et al., 2021) ViT-B/16 (Dosovitskiy et al., 2021), CLIP ViT-L/14, and a SigLIP (Zhai et al., 2023) ViT-B/16 model. To train the SAE, we ablate across values for the expansion factors and learning rates and select the best configuration based on the mean reconstruction accuracy and classification accuracy on the reconstructed features. We then train the masking module by sweeping across learning rates and selecting the configuration that is best for retrieval on the validation split of CC3M (Sharma et al., 2018), to ensure generalization to different datasets. Full details are provided in the supplement in Appendix B.1. As

Table 2: **Fine-grained, long-caption retrieval on DOCCI and IIW.** All models are trained on CC12M dataset. We report R@1 and R@5 for both image-to-text and text-to-image retrieval. For TEVI and No-SAE, we use re-ranking, and report mean ± std over 5 random seeds. We see that TEVI improves retrieval performance across models and datasets.

Model	DOCCI (Onoe et al., 2024)				IIW (Garg et al., 2024)			
	I → T		T → I		I → T		T → I	
	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5
CLIP ViT-B/16	20.38	42.36	7.17	16.96	50.98	77.94	16.88	32.66
No SAE	**24.29** ±0.22	**47.24** ±0.18	**8.40** ±0.08	**18.66** ±0.04	**56.86** ±0.31	**81.83** ±0.50	**18.76** ±0.15	**34.84** ±0.14
TEVI	**24.52** ±0.15	**47.30** ±0.14	**8.30** ±0.04	18.55 ±0.03	55.49 ±0.36	**82.42** ±0.19	**18.77** ±0.08	34.57 ±0.05
CLIP ViT-L/14	23.60	46.68	8.21	18.70	53.92	81.37	18.05	35.59
TEVI	**26.84** ±0.47	**50.04** ±0.20	**9.09** ±0.06	**20.44** ±0.03	**60.78** ±0.60	**84.80** ±0.35	**19.72** ±0.05	**37.63** ±0.08
SigLIP ViT-B/16	20.68	41.84	7.41	17.17	48.20	76.14	16.94	33.08
TEVI	**25.29** ±0.22	**47.37** ±0.09	**8.32** ±0.06	**18.59** ±0.02	**57.84** ±0.95	**80.98** ±0.19	**18.90** ±0.11	**35.68** ±0.10
SharedCLIP ViT-B/16	19.02	41.06	7.16	17.52	51.14	79.08	17.10	33.78
TEVI	**22.64** ±0.27	**45.23** ±0.07	**7.94** ±0.06	**18.42** ±0.03	**55.59** ±0.31	**80.98** ±0.25	**17.91** ±0.18	**34.61** ±0.10
AlignCLIP ViT-B/16	20.32	41.88	7.39	17.43	53.92	82.03	17.46	34.27
TEVI	**23.65** ±0.26	**46.02** ±0.15	**8.13** ±0.08	**18.68** ±0.05	**58.92** ±0.65	**84.31** ±0.12	**19.75** ±0.20	**36.06** ±0.21

in Sec. 4, we compare against a no-SAE baseline with a mask trained on CLIP image embeddings.

Mask Analysis. We first evaluate if the learnt mask is meaningful and selective. As a sanity check, we evaluate the distribution of mask values, and find that it is non-uniform and high only for a few latents, as expected (Tab. B12). We next evaluate whether (i) latents selected by the mask, given by mask values, correlate to (ii) the presence of the concepts they encode in the conditioning caption. In other words, if a latent encodes a particular concept, we want to evaluate if captions containing that concept correspond to high mask values for that latent. As a proxy for the concept encoded by a latent, we follow Rao et al. (2024) and compute the cosine similarity between the text embedding of a caption and the corresponding SAE decoder dictionary vector, where a higher value indicates that the latent encodes the concepts present in the caption. We then compute the Pearson correlation ρ between this similarity and the mask value across captions, with positive correlation indicating that the mask is meaningful. We find that 20% of active latents on DOCCI show $\rho > 0.1$, as compared to 0% for a caption-shuffled null baseline. While this number may appear small, not all latents are likely to be equally important. To evaluate this, we zero out mask values for the top, random, and bottom m latents, and evaluate the impact on downstream retrieval. We sort the latents in two ways, by ρ and by the mask values themselves. If the mask keeps latents important for alignment, then one would expect removing the top latents to lead to a strong drop in retrieval, and the opposite when removing the bottom latents. We find (Tab. 1) that the results consistently match expectations for both rankings: ablating the top latents leads to strong drops in retrieval, followed by random, and followed by the bottom latents. The latents that are well correlated are also the most impactful, e.g. ablating the top 32 by ρ leaves 5.2 R@1, as compared to 21.0 with the bottom 32. Results for text-to-image retrieval are provided in Tab. B13.

Selectivity of Conditioning. Next, we evaluate whether conditioning is selective of the concepts present in the caption. For each DOCCI image whose caption names two objects, we condition the image on each of them in turn, and evaluate whether each edited embedding is closer to its own object. Over 1,061 images and 1,528 object pairs, we find that TEVI selects the named object 79.4% of the time, as compared to 55.6% for the no-SAE baseline, where chance is 50%.

Vision-Language Alignment. To evaluate the impact of TEVI on cross-modal alignment, we plot the cosine similarities between image and corresponding text embeddings, both before and after editing. We find (Fig. B2) that alignment improves across datasets after applying TEVI (e.g. 0.417→0.470 for MSCOCO). Further results on different models are provided in Appendix B.7. Note however that alignment is a diagnostic metric; with the primary goal being to improve downstream performance, which we discuss next.

Retrieval Performance. We evaluate the performance of TEVI for text-to-image and image-to-text retrieval across fine-grained long-caption (DOCCI (Onoe et al., 2024), IIW (Garg et al., 2024)) and coarse-grained short-caption (MS COCO (Lin et al., 2014), Flickr30k (Plummer et al., 2015)) retrieval benchmarks. Similar to other cross-modal conditioning methods (e.g. Xie et al., 2025b; Xiao et al., 2025), TEVI incurs an additional cost for retrieval, since every image is conditioned on every text. To mitigate this, we only use TEVI to

Table 3: **Robust retrieval performance of TEVI on the RoCOCO benchmark (Park et al., 2024).** We find that TEVI improves performance over the baseline across settings.

Model	COCO	Rand-voca			Same-concept			Diff-concept			Danger		
	R@1	R@1 (↑)	drop rate (↓)	RSMS (↓)	R@1 (↑)	drop rate (↓)	RSMS (↓)	R@1 (↑)	drop rate (↓)	RSMS (↓)	R@1 (↑)	drop rate (↓)	RSMS (↓)
CLIP	33.26	21.26	12.00	46.90	21.74	11.52	43.36	22.12	11.14	43.98	22.92	10.34	39.74
+TEVI	**36.00**	**26.14**	**9.86**	**33.98**	**25.74**	**10.26**	**37.12**	**25.60**	**10.40**	**35.70**	**27.12**	**8.88**	**31.00**

rerank the top-10 candidates obtained via the base model. We find (Tabs. 2 and B1) that TEVI consistently improves retrieval performance across nearly all datasets and models. Qualitative examples are provided in Fig. 1 and Appendix B.9. Interestingly, the no-SAE baseline, despite performing worse for latent selectivity and steering, performs comparably well. We believe this could be because of the added challenge of disentangling attributes and the noisiness of captions in real data, and we leave further investigation of this for future work. To further understand this, we evaluate on a more controlled setup, where we fix the conditioning text to a single object and use the exhaustive MS COCO instance annotations as ground truth. We find (Fig. 6) that while both TEVI and no-SAE promote the conditioned object, only TEVI removes the object it was not conditioned on, which suggests that TEVI is more selective.

Applicability to Existing Methods. We additionally compare against SharedCLIP and AlignCLIP (Eslami and de Melo, 2025), which recently proposed using intra-modal separation objectives during CLIP training to improve cross-modal alignment and retrieval performance. Even though these training objectives already provide gains, we find (Tabs. 2 and B1) that our approach can additionally help these methods with consistent and significant improvement on fine-grained datasets and competitive performance on coarse-grained datasets.

Comparison to SmartCLIP. We compare TEVI against SmartCLIP (Xie et al., 2025b), which learns a text-conditioned mask on raw vision embeddings. For a fair comparison, we train on a similar setup with CC12M. We find (Tab. 4, and Appendix B.8) that TEVI consistently outperforms it on fine-grained datasets and is competitive on coarse-grained datasets, highlighting the benefit of disentangling representations. Additionally, similar to Xie et al. (2025b), we obtain comparable zeroshot performance as the baseline; for details refer to Appendix B.5.

Robust Retrieval. We additionally evaluate TEVI for robust retrieval, using the RoCOCO (Park et al., 2024) benchmark. This augments the caption set of MS COCO (Lin et al., 2014) with perturbed captions containing irrelevant concepts that alter their meaning, and should not be retrieved by the model. Specifically, it contains four sets—'Rand-voca', 'Danger', 'Same-concept', and 'Diff-concept'—which replace words in the original caption with alternatives based on the set. Following (Park et al., 2024), we report (Tab. 3) the drop rate (percentage drop in retrieval after augmenting captions), and Recall Score of Manipulated Samples (RSMS) (fraction of data where a perturbed entity was retrieved on top). We find that TEVI provides consistent improvements across these metrics on all data splits. Additional results on other backbones are provided in Appendix B.4.

Table 4: **Retrieval comparison with SmartCLIP.** TEVI outperforms on fine-grained datasets and is competitive on coarse-grained datasets. Full results in Appendix B.8.

Model	DOCCI I→T, T→I				Flickr30k I→T, T→I			
	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5
CLIP	20.4	42.4	7.2	17.0	59.7	83.7	42.5	70.3
+TEVI	**24.2**	**48.7**	**8.6**	**20.0**	**64.2**	**85.7**	44.8	72.1
SmartCLIP	21.8	45.0	7.5	17.8	61.0	85.1	**45.3**	**72.8**

6 Conclusion

In this work, we explored whether captions can guide what information is preserved in CLIP image embeddings, in order to improve cross-modal alignment. We proposed a framework, TEVI, to use text as a signal to edit image embeddings, such that the resultant edited embedding is better aligned with the text. We first used a synthetic setup, where we could explicitly control the information imbalance, to show that text-conditioned editing preserves information described in the caption. We then applied TEVI to models trained on natural images and showed its effectiveness for learning meaningful and sparse masks that are selective of the conditioning attributes. Finally, we showed that TEVI helps improve retrieval, and complements existing approaches such as AlignCLIP. On the whole, we showed that decomposing embeddings into concepts makes the information imbalance controllable.

Limitations

While our proposed TEVI framework provides a controllable setup that shows strong benefits for a variety of CLIP models trained on CC12M, extending to large-scale models trained on billions of data points remains a challenge, likely owing to the complexity of training sufficiently large and diverse SAEs. Nevertheless, our work intends to provide a clear proof of concept. Scaling to models trained on large-scale data would be a fruitful direction for future research, and this might involve using state-of-the-art SAEs and extracting concepts more granularly, such as at the image patch level. In addition, our work relies on SAEs for disentanglement, and is limited by its accuracy. Gains in some setups, e.g. retrieval on benchmark data, remain comparable to learning a mask directly without disentanglement from SAEs, which could be a result of this. As with all cross-modal conditioning-based methods, TEVI also incurs an additional computational cost as compared to standard retrieval. Exploring how to extend TEVI to practical setups remains an interesting direction for future research.

Acknowledgements

Funded in part by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - GRK 2853/1 "Neuroexplicit Models of Language, Vision, and Action" - project number 471607914. We thank Mayank Jobanputra for the helpful discussions and feedback.

References

Mothilal Asokan, Kebin Wu, and Fatima Albreiki. 2025. FineLIP: Extending CLIP's Reach via Fine-Grained Alignment with Longer Text Inputs. In *CVPR*, pages 14495–14504.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, and 6 others. 2023. Towards Monosemanticity: Decomposing Language Models With Dictionary Learning. *Transformer Circuits Thread*.

Bart Bussmann, Noa Nabeshima, Adam Karvonen, and Neel Nanda. 2025. Learning Multi-Level Features with Matryoshka Sparse Autoencoders. In *ICML*.

Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. 2021. Conceptual 12M: Pushing Web-Scale Image-Text Pre-training to Recognize Long-Tail Visual Concepts. In *CVPR*, pages 3558–3568.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. 2024. Sparse Autoencoders Find Highly Interpretable Features in Language Models. In *ICLR*.

Bartosz Cywiński and Kamil Deja. 2025. SAeUron: Interpretable Concept Unlearning in Diffusion Models with Sparse Autoencoders. In *ICML*.

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. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*.

Sedigheh Eslami and Gerard de Melo. 2025. Mitigate the Gap: Improving Cross-Modal Alignment in CLIP . In *ICLR*.

Eoin Farrell, Yeu-Tong Lau, and Arthur Conmy. 2024. Applying Sparse Autoencoders to Unlearn Knowledge in Language Models. In *Neurips Safe Generative AI Workshop 2024*.

Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. 2025. Scaling and Evaluating Sparse Autoencoders. In *ICLR*.

Roopal Garg, Andrea Burns, Burcu Karagol-Ayan, Yonatan Bitton, Ceslee Montgomery, Yasumasa Onoe, Andrew Bunner, Ranjay Krishna, Jason Michael Baldridge, and Radu Soricut. 2024. ImageInWords: Unlocking Hyper-Detailed Image Descriptions. In *EMNLP*, pages 93–127.

Shashank Goel, Hritik Bansal, Sumit Bhatia, Ryan Rossi, Vishwa Vinay, and Aditya Grover. 2022. CyCLIP: Cyclic Contrastive Language-Image Pretraining. In *NeurIPS*, volume 35, pages 6704–6719.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. 2021a. The Many Faces of Robustness: A Critical Analysis of Out-of-Distribution Generalization. In *ICCV*.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Xiaodong Song. 2021b. Natural Adversarial Examples. In *CVPR*.

Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. 2021. OpenCLIP.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling Up Visual

and Vision-Language Representation Learning with Noisy Text Supervision. In *ICML*, pages 4904–4916.

Sonia Joseph, Praneet Suresh, Ethan Goldfarb, Lorenz Hufe, Yossi Gandelsman, Robert Graham, Danilo Bzdok, Wojciech Samek, and Blake Aaron Richards. 2025. Steering CLIP's Vision Transformer with Sparse Autoencoders. *arXiv preprint arXiv:2504.08729*.

Alex Krizhevsky. 2009. Learning Multiple Layers of Features from Tiny Images. *Technical Report, Computer Science Department, University of Toronto*.

Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. 2022. Supervision Exists Everywhere: A Data Efficient Contrastive Language-Image Pre-training Paradigm. In *ICLR*.

Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. 2022. Mind the Gap: Understanding the Modality Gap in Multi-Modal Contrastive Representation Learning. In *NeurIPS*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft COCO: Common Objects in Context. In *ECCV*, pages 740–755.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual Instruction Tuning. In *NeurIPS*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *ICLR*.

Marco Mistretta, Alberto Baldrati, Lorenzo Agnolucci, Marco Bertini, and Andrew D Bagdanov. 2025. Cross the Gap: Exposing the Intra-modal Misalignment in CLIP via Modality Inversion. In *ICLR*.

Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. 2022. SLIP: Self-supervision meets Language-Image pre-training. In *ECCV*, pages 529–544.

Yasumasa Onoe, Sunayana Rane, Zachary Berger, Yonatan Bitton, Jaemin Cho, Roopal Garg, Alexander Ku, Zarana Parekh, Jordi Pont-Tuset, Garrett Tanzer, Su Wang, and Jason Baldridge. 2024. DOCCI: Descriptions of Connected and Contrasting Images. In *ECCV*, pages 291–309.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation Learning with Contrastive Predictive Coding. *arXiv preprint arXiv:1807.03748*.

Mateusz Pach, Shyamgopal Karthik, Quentin Bouniot, Serge Belongie, and Zeynep Akata. 2025. Sparse Autoencoders Learn Monosemantic Features in Vision-language Models. In *NeurIPS*, volume 38, pages 95706–95742.

Seulki Park, Daeho Um, Hajung Yoon, Sanghyuk Chun, and Sangdoo Yun. 2024. RoCOCO: Robustness Benchmark of MS-COCO to Stress-Test Image-Text Matching Models. In *ECCV Synthetic Data for Computer Vision Workshop*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, and 2 others. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *NeurIPS*.

Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models. In *ICCV*, pages 2641–2649.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models from Natural Language Supervision. In *ICML*, pages 8748–8763.

Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Tom Lieberum, Vikrant Varma, János Kramár, Rohin Shah, and Neel Nanda. 2024a. Improving Sparse Decomposition of Language Model Activations with Gated Sparse Autoencoders . In *NeurIPS*.

Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. 2024b. Jumping Ahead: Improving Reconstruction Fidelity with JumpReLU Sparse Autoencoders. *arXiv preprint arXiv:2407.14435*.

Sukrut Rao, Sweta Mahajan, Moritz Böhle, and Bernt Schiele. 2024. Discover-then-Name: Task-Agnostic Concept Bottlenecks via Automated Concept Discovery. In *ECCV*.

Anton Razzhigaev, Arseniy Shakhmatov, Anastasia Maltseva, Vladimir Arkhipkin, Igor Pavlov, Ilya Ryabov, Angelina Kuts, Alexander Panchenko, Andrey Kuznetsov, and Denis Dimitrov. 2023. Kandinsky: An Improved Text-to-Image Synthesis with Image Prior and Latent Diffusion. In *EMNLP (Demos)*.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. 2019. Do ImageNet Classifiers Generalize to ImageNet? In *ICML*, pages 5389–5400. PMLR.

Jona Ruthardt, Manu Gaur, Deva Ramanan, Makarand Tapaswi, and Yuki M. Asano. 2026. Steerable Visual Representations. In *ECCV*.

Simon Schrodi, David T Hoffmann, Max Argus, Volker Fischer, and Thomas Brox. 2025. Two Effects, One Trigger: On the Modality Gap, Object Bias, and Information Imbalance in Contrastive Vision-Language Models. In *ICLR*.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset for Automatic Image Captioning. In *ACL*, pages 2556–2565.

Peiyang Shi, Michael C Welle, Mårten Björkman, and Danica Kragic. 2023. Towards Understanding the Modality Gap in CLIP. In *ICLR 2023 Workshop on Multimodal Representation Learning: Perks and Pitfalls*.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. 2025. SigLIP 2: Multilingual Vision-Language Encoders with Improved Semantic Understanding, Localization, and Dense Features. *arXiv preprint arXiv:2502.14786*.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. 2019. Learning Robust Global Representations by Penalizing Local Predictive Power. In *NeurIPS*, volume 32.

Rui Xiao, Sanghwan Kim, Mariana-Iuliana Georgescu, Zeynep Akata, and Stephan Alaniz. 2025. FLAIR: VLM with Fine-grained Language-informed Image Representations. In *CVPR*.

Chunyu Xie, Bin Wang, Fanjing Kong, Jincheng Li, Dawei Liang, Gengshen Zhang, Dawei Leng, and Yuhui Yin. 2025a. FG-CLIP: Fine-grained Visual and Textual Alignment. In *ICML*.

Shaoan Xie, Lingjing Kong, Yujia Zheng, Yu Yao, Zeyu Tang, Eric P Xing, Guangyi Chen, and Kun Zhang. 2025b. SmartCLIP: Modular Vision-language Alignment with Identification Guarantees. In *CVPR*, pages 29780–29790.

Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. 2022. FILIP: Fine-grained Interactive Language-image Pre-training. In *ICLR*.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. 2022. CoCa: Contrastive Captioners are Image-Text Foundation Models. *TMLR*.

Vladimir Zaigrajew, Hubert Baniecki, and Przemyslaw Biecek. 2025. Interpreting CLIP with Hierarchical Sparse Autoencoders. In *ICML*.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid Loss for Language Image Pre-Training. In *ICCV*.

A Controlled Synthetic Setup

More implementation details about the method on the MAD dataset (Schrodi et al., 2025) is as follows. This is in addition to the details provided in Sec. 3.3.

A.1 Implementation Details

Dataset. The MAD dataset (Schrodi et al., 2025) is a synthetic dataset that consists of images of digits with colors and morphological transforms. Specifically, it consists of the following attributes: 'Digit': {0,1,2,3,4,5,6,7,8,9}, 'Thickthinning': {thickening, thinning, no thickthinning}, 'Scaling': {large, small}, 'Fracture': {fracture, no fracture}, 'Swelling': {swelling, no swelling}, and 'Color': {gray, red, green, blue, cyan, magenta, yellow}, which gives a total of 6 attribute categories and an aggregate of 26 attribute values. The training dataset consists of 1.44 million images, and the test data consists of 240,000 images. For training CLIP models, captions are generated by using the digit and a random sample of three out of the remaining five attribute categories in the image, placed in a random order with a '-' separator.

CLIP models. Following (Schrodi et al., 2025), we train the CLIP model using the AdamW (Loshchilov and Hutter, 2019) optimizer for 200 epochs with a batch size of 128 and a weight decay of 0.1. We sweep over learning rates of $\{10^{-5}, 5 \times 10^{-4}, 5 \times 10^{-5}\}$ and pick the learning rate 5×10^{-4} and the final checkpoint with the lowest loss. We use the cosine annealing for the learning rate. For finetuning the text encoder along with the learnt mask, we sweep over learning rates $\{10^{-2}, 10^{-3}, 10^{-4}, 5 \times 10^{-4}, 10^{-5}, 10^{-6}\}$ and pick the learning rate 10^{-6}. Our code is based on the implementation from OpenCLIP (Ilharco et al., 2021) using PyTorch (Paszke et al., 2019).

SAE models. We use the TopK SAE implementation of (Gao et al., 2025). This particular type of SAE chooses the top few ($= K$) SAE latents to reconstruct the original CLIP embeddings and uses an auxiliary loss that approximates the reconstruction error using the top few (=auxK) dead latents. For our setup, we use K as 12, auxK as 18, after sweeping over these hyperparameters and we train the SAE for 200 epochs. We choose the SAE configuration using with the lowest reconstruction error. We sweep over learning rates $\{10^{-1}, 10^{-2}\}$ and expansion factor $1, 2, 4$ and pick 10^{-2}, 1 respectively for both the SAEs used for CG-SAE and TEVI.

Learnt masks. We use a 3-layer MLP with a hidden dimension of 256 and ReLU activations between linear layers. This MLP predicts values to mask the SAE latents. We sweep over learning rates 0.1, 0.01, and 0.001, and choose 0.01 as the optimal learning rate. We train for 25 epochs with 5 warmup epochs using the AdamW optimizer (Loshchilov and Hutter, 2019).

A.2 Additional Results for Concept Disentanglement with CG-SAE

In this section, we provide full results for concept disentanglement from our CG-SAEs, as discussed in Sec. 3.3. In Figs. A4 and A5, we show AUC ROC plots for other attribute categories, to supplement the results provided for the 'Swelling' category in Fig. 3. In Figs. A2 and A3, we show top activating images for each latent, expanding on Fig. A1. Both quantitatively and qualitatively, we find that our CG-SAE effectively disentangles concepts and each latent activates highly only on the attribute value assigned to it.

A.3 Additional Results for Effectiveness of Conditioning

In this section, we provide additional results on our method's effectiveness of conditioning (Tabs. A1 and A2). Specifically, for robustness, we report results averaged across five runs, and find that the trends observed in Fig. 4 continue to hold.

Further, in Fig. A6, we report the retrieval ordering experiment of Fig. 5 for all three retrieval directions, i.e. image $\rightarrow$ text, text $\rightarrow$ image and image $\rightarrow$ image. We find that the rank-position differences move in the expected direction in each of the three settings.

Finally, in Fig. A7, we show qualitative examples for text $\rightarrow$ image retrieval, which expand on the single example shown in Fig. 5 (right). For each example, we show the top-8 retrieved images before and after conditioning, where the conditioned attribute is marked with ● and the control attribute with ■. In most examples, we find that conditioning may keep the conditioned attribute in the retrieved set while discarding the control attribute.

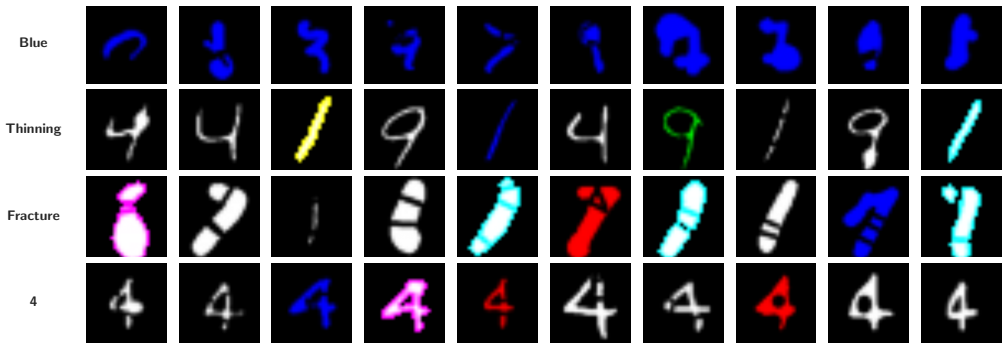


Fig. A1: **Qualitative examples of top activating images for the setup when the CG-SAE is trained with fixed semantics of latents.** Each row corresponds to an SAE latent and is labelled with the predefined concept that is assigned to that latent (Sec. 3.2). The columns show examples of images that maximally activate these latents. We find that the SAE learns to disentangle the CLIP image features into concepts as specified by the fixed weight of the latent.

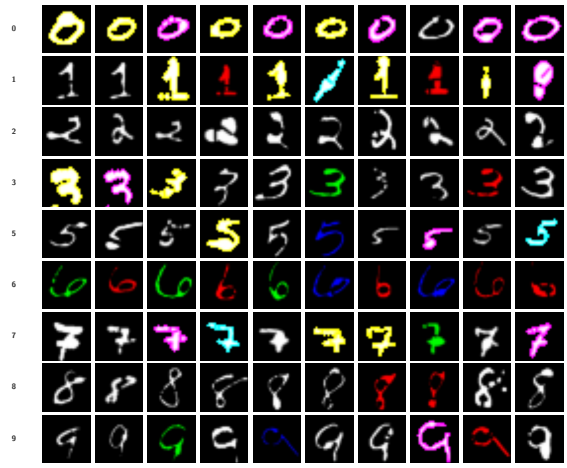


Fig. A2: **Qualitative examples of top activating images for all the digits of the CG-SAE latents.** Each row corresponds to a CG-SAE latent and is labelled with the predefined concept that is assigned to that latent (Sec. 3.2). The columns show examples of images that maximally activate these latents. We find that the CG-SAE learns to disentangle the CLIP image features into concepts as specified by the fixed weight of the latent.

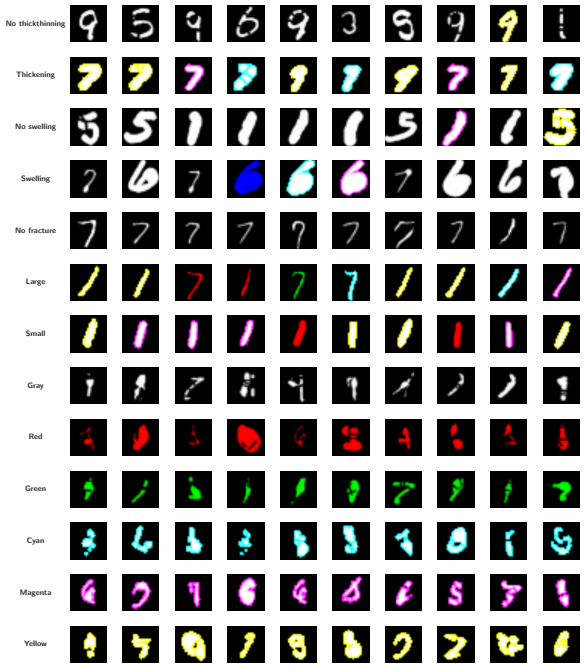


Fig. A3: **Qualitative examples of top activating images of the remaining CG-SAE latents.** Each row corresponds to a CG-SAE latent and is labelled with the predefined concept that is assigned to that latent (Sec. 3.2). The columns show examples of images that maximally activate these latents. We find that the CG-SAE learns to disentangle the CLIP image features into concepts as specified by the fixed weight of the latent.

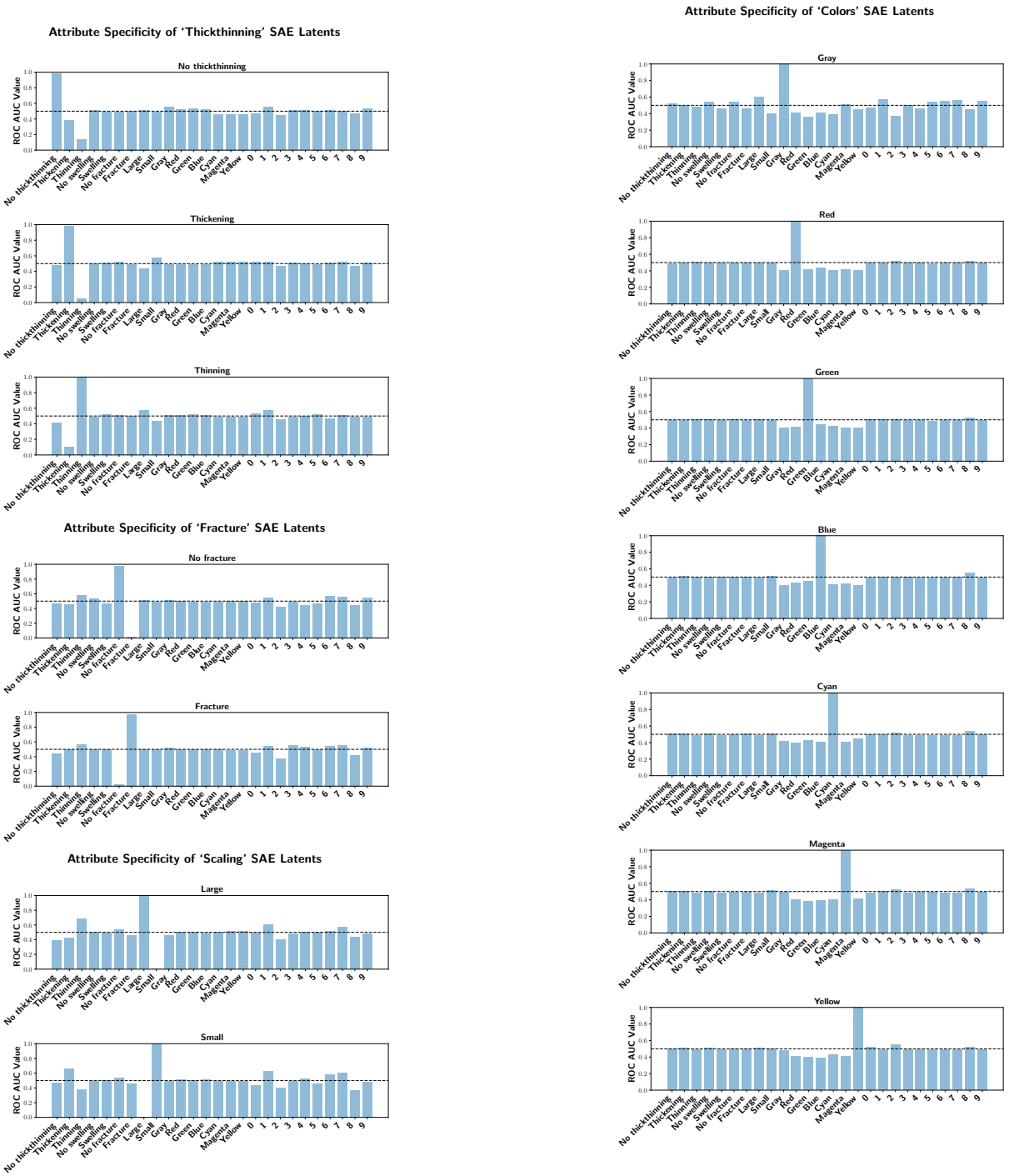


Fig. A4: **Attribute specificity of CG-SAE latents for attribute categories 'Thickthinning', 'Fracture', 'Scaling', and 'Color'.** We plot the area under the receiver operating characteristic (ROC) curve (AUC) for CG-SAE latents corresponding to the attribute values of each attribute category. We find that the latents are highly attribute-specific, with the AUC being close to 1 for the attribute the latent is assigned to, and 0.5 for unrelated attributes. For results for other attribute categories, see Fig. 3 and Fig. A5.

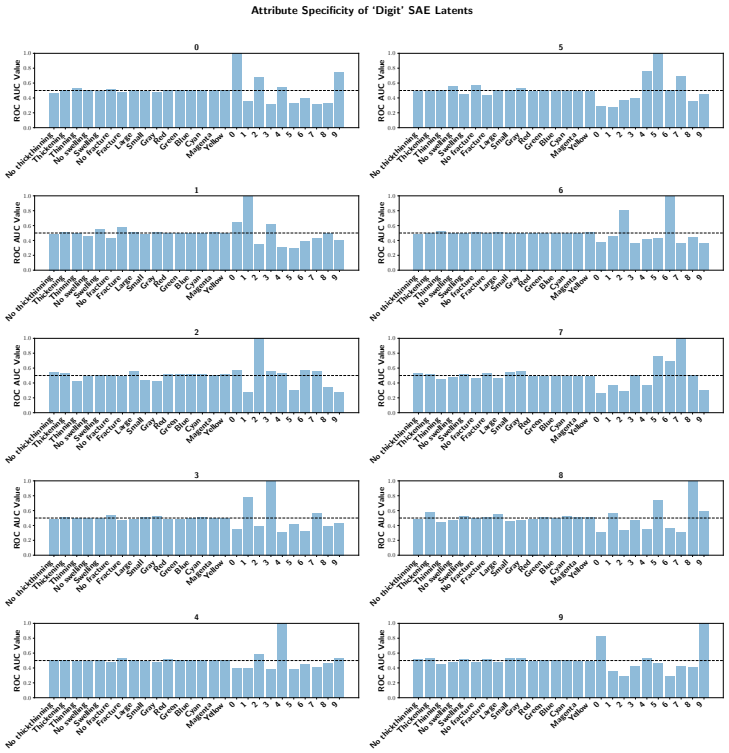


Fig. A5: **Attribute specificity of CG-SAE latents for attribute category 'Digit'.** We plot the area under the receiver operating characteristic (ROC) curve (AUC) for CG-SAE latents corresponding to each digit attribute. We find that the latents are fairly attribute-specific, with the AUC being close to 1 for the attribute the latent is assigned to, and close to 0.5 for unrelated attributes. For results for other attribute categories, see Fig. 3 and Fig. A4.

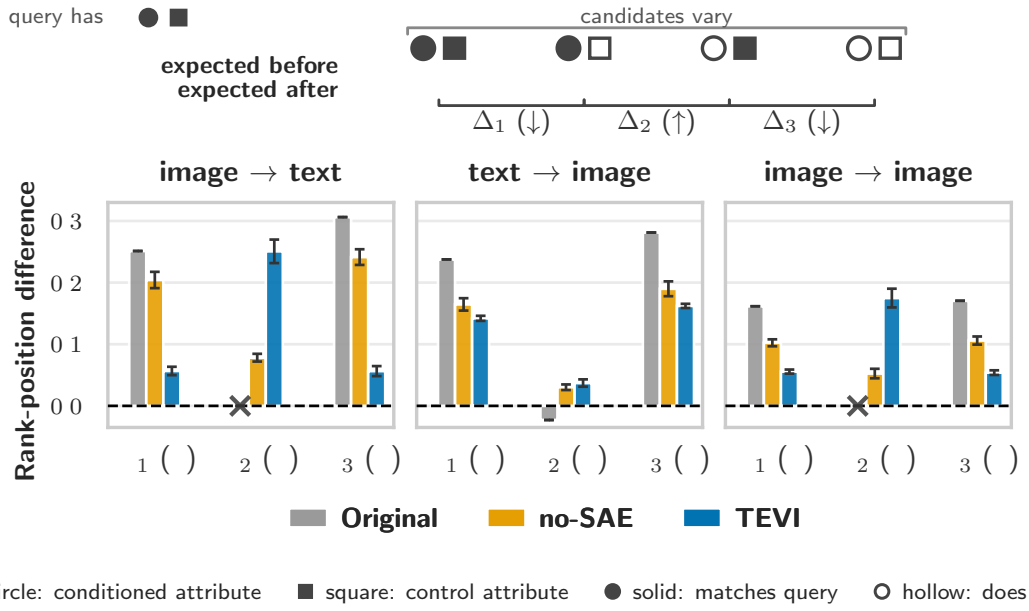


Fig. A6: **Conditioning reorders the retrieved set on MAD.** We condition on one attribute (●) and use another as control (■), see top right. With a query containing both ● and ■ and varying candidates, we expect Δ_1 and Δ_3 to fall towards zero and Δ_2 to grow, as the edited embedding should preserve ● and discard ■. Δ denotes the difference in mean rank position between two groups of candidates, where each rank is normalized between 0 and 1. The pattern holds in all three directions, for image $\rightarrow$ text, text $\rightarrow$ image and image $\rightarrow$ image. All bars are the mean over five training seeds and the error bars are tshe standard deviation over those seeds. See Fig. 5.



Fig. A7: **MAD text → image examples across the range of outcomes.** Each block shows the top-8 retrieved images before and after conditioning, for a different query caption and attribute pair. The circle marks the conditioned attribute and the square the control, filled when the retrieved image matches the query on that attribute. Blocks are a mix, so that weaker cases appear alongside strong ones. Fig. 5 shows one of these examples at full size.

Table A1: **Effectiveness of conditioning with CG-SAE for the MAD dataset.** For each attribute category (rows), we report the accuracy with the original embeddings (col. 1) and conditioned embeddings, where the attribute of that category is present in the caption (col. 2) and is absent from the caption (col. 3), averaged over five runs. We see that when the attribute is present in the caption, the accuracy is at par with the original embeddings. However, when the attribute is absent from the caption, the accuracy reaches close to random chance (col. 4), which shows that conditioning is effective in only preserving information about attributes present in the caption.

	Original (↑)	Attribute in Caption (↑)	Attribute not in Caption	Random Chance
Thickthinning	92.9	97.8 ± 1.3	35.4 ± 2.1	33.3
Swelling	96.2	96.6 ± 6.8	50.2 ± 0.1	30.0
Fracture	92.8	97.5 ± 3.0	50.5 ± 0.7	50.0
Scaling	99.7	99.9 ± 0.1	60.7 ± 3.4	50.0
Color	100.0	100.0 ± 0.0	34.2 ± 1.7	14.3

Table A2: **Effectiveness of conditioning with TEVI for the MAD dataset.** For each attribute category (rows), we report the accuracy with the original embeddings (col. 1) and conditioned embeddings, where the attribute of that category is present in the caption (col. 2) and is absent from the caption (col. 3), averaged over five runs. We see that when the attribute is present in the caption, the accuracy is at par with the original embeddings. However, when the attribute is absent from the caption, the accuracy reaches close to random chance (col. 4), which shows that conditioning is effective in only preserving information about attributes present in the caption.

	Original (↑)	Attribute in Caption (↑)	Attribute not in Caption	Random Chance
Thickthinning	92.9	99.1 ± 0.7	54.3 ± 2.4	33.3
Swelling	96.2	98.3 ± 0.2	57.0 ± 2.7	30.0
Fracture	92.8	93.5 ± 0.9	59.1 ± 1.5	50.0
Scaling	99.7	95.2 ± 2.7	57.1 ± 3.5	50.0
Color	100.0	96.3 ± 1.9	22.6 ± 1.9	14.3

B Evaluation on CLIP Models Trained with Natural Images

B.1 Implementation Details

In this section, we provide implementation details.

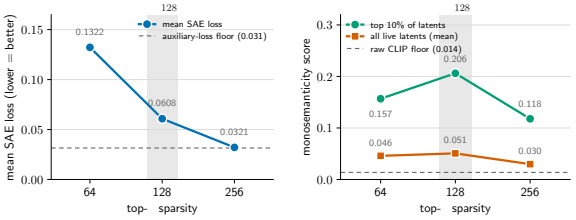
CLIP models. We train CLIP models using the AdamW (Loshchilov and Hutter, 2019) optimizer for 30 epochs with a batch size of 2048 and a weight decay of 0.1. We sweep over learning rates of $\{10^{-3}, 5 \times 10^{-4}\}$ and pick the learning rate 10^{-3} and the final checkpoint with the lowest loss. We use cosine annealing for the learning rate. For fine-tuning the text encoder along with training the masking module, we sweep over learning rates $\{10^{-4}, 10^{-5}, 10^{-6}\}$. Our code is based on the implementation from OpenCLIP (Ilharco et al., 2021; Eslami and de Melo, 2025) using PyTorch (Paszke et al., 2019).

SAE models. We use the TopK SAE implementation of (Gao et al., 2025). This particular type of SAE chooses the top few ($= K$) SAE latents to reconstruct the original CLIP embeddings and uses an auxiliary loss that approximates the reconstruction error using the top few (=auxK) dead latents. For our setup, we vary the sparsity (K) value across values $\{64, 128, 256\}$ and choose $K = 128$, which yields the most monosemantic dictionary; we ablate this choice in Fig. B1. We use auxK as 256, after sweeping over these hyperparameters and we train the SAE for 50 epochs. We sweep over learning rates $\{10^{-3}, 5{\times}10^{-3}, 10^{-4}, 5{\times}10^{-4}\}$, and use an expansion factor of 32 for ViT-B/16 backbones and 16 for ViT-L/14 backbones. For the learning rate and expansion factor, we discard configurations with a high reconstruction error or a large number of dead latents, and among the remaining ones we select the configuration with the best monosemanticity score. We calculate the monosemanticity score following (Pach et al., 2025).

Learnt masks. We use a 3-layer MLP and ReLU activations between linear layers. This MLP predicts values to mask the SAE latents. We sweep over learning rates $\{10^{-4}, 10^{-5}, 10^{-6}\}$. We train for 3 epochs using the AdamW optimizer (Loshchilov and Hutter, 2019).

Baselines. For the results on SharedCLIP and AlignCLIP in Tabs. 2 and B1, we take the checkpoints given by the authors (Eslami and de Melo, 2025) and apply TEVI on them. Note that these checkpoints were trained with a batch size of 512, and in contrast we train our own CLIP models with a batch size of 2048. As a result, the performance of AlignCLIP/SharedCLIP versus our CLIP models cannot be directly compared.

Compute usage. We used H100 and A100 GPUs for training the CLIP models on CC12M, training the SAEs, and the masking module. The maximum number of GPU hours needed for training the CLIP model is upper bounded by 18 hours, and for train-



(a) Training objective (b) Monosemanticity

Fig. B1: **Choosing the SAE sparsity K for CLIP ViT-B/16 on CC12M.** Left: SAE loss vs K plot. Right: Monosemanticity score vs K.

ing the masking module is upper bounded by 15 hours.

Robust retrieval. Following (Park et al., 2024), in Tab. 3, we report the drop rate and the RSMS metric in addition to the retrieval metrics. Drop rate is calculated as $\frac{(R@1 - Rp@1)}{R@1}$, where Rp is the retrieval score under the perturbed setting. RSMS calculates the percentage of newly added semantically perturbed captions that are retrieved in the first spot by the model. We note that, due to that use of SAE to decompose CLIP image features into concepts, our method is unable to work with altered images as given by the benchmark, where the original image is superimposed with a random patch from a different image. As a result, we only report image-to-text retrieval performance in Tab. 3.

B.2 Full Retrieval Results

We report the full retrieval results here. Tab. B2 reports fine-grained, long-caption retrieval on DOCCI (Onoe et al., 2024) and IIW (Garg et al., 2024), and Tab. B3 reports coarse-grained, short-caption retrieval on MSCOCO (Lin et al., 2014) and Flickr30k (Plummer et al., 2015). For the fine-grained retrieval results using re-ranking, refer to Tab. 2 of the main paper. The corresponding coarse-grained results are reported in Tab. B1.

B.3 Baseline Comparisons

TEVI involves training a masking module for a small number of extra epochs while also fine-tuning the text encoder. For a fairer comparison accounting for this extra training budget, we report in Tabs. B4 and B5 the performance of CLIP models where the text encoder is fine-tuned for the same number of extra epochs. Similarly we also add the ablation of training the masking module on random SAE. We find that TEVI shows improved retrieval performance across datasets.

B.4 Additional Results on RoCOCO

In addition to the robust retrieval results provided in Tab. 3 on the CLIP ViT-B/16 backbone, here we provide additional results comparing TEVI against SigLIP and AlignCLIP.

B.5 Zeroshot Classification Results

Similar to SmartCLIP (Tab. 3 in Xie et al., 2025b), we find (Tab. B7) that the zeroshot classification performance is comparable to the baseline, possibly since zeroshot does not use long captions with multiple attributes, unlike retrieval. Since our method is post hoc, one can achieve the best of both worlds by using the original CLIP embeddings for zeroshot classification.

B.6 Ablation without Negative Conditioning

As noted in Sec. 5.1, we use Eq. (11) when applying TEVI on CLIP models trained with CC12M, to additionally provide conditioning signal on negative captions. In Tabs. B8 and B9, we empirically validate this by performing an ablation study against models trained without negative conditioning, i.e. using Eq. (10), and find that the performance degrades significantly.

B.7 Cross-modal Alignment

Similar to Fig. B2, Figs. B3 and B4 show cross-modal alignment for SharedCLIP and AlignCLIP across four downstream datasets before and after applying TEVI, and find that it similarly improves alignment.

B.8 Comparison against SmartCLIP

In Tabs. B10 and B11, we compare our approach against SmartCLIP (Xie et al., 2025b), applied on the same CLIP models trained on CC12M. We see that TEVI performs comparably on MS COCO and Flickr30k and outperforms on the fine-grained DOCCI and IIW datasets, possibly due to explicit disentanglement from SAEs.

Additional mask analysis. We provide the full results for the mask analysis of Sec. 5.2. Table B12 reports the mask sparsity, Tab. B13 reports the latent ablations in both retrieval directions, and Tab. B14 repeats the conditioning test of Fig. 6 for image-to-image retrieval, where the gallery consists of the other images instead of the captions. We find that TEVI still removes the control object, but neither arm promotes the conditioned object, which makes this the weaker of the two directions. We find that out of K=128 dimensions,

Table B1: **Coarse-grained, short-caption retrieval on MSCOCO and Flickr30k.** All the models are trained on CC12M dataset. We report R@1 and R@5 for both image-to-text and text-to-image retrieval. For TEVI and No-SAE, we use re-ranking, and report mean ± std over 5 random seeds. We see that TEVI improves retrieval performance across models and datasets.

Model	COCO (Lin et al., 2014)				Flickr30k (Plummer et al., 2015)			
	I → T		T → I		I → T		T → I	
	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5
CLIP ViT-B/16	32.98	59.02	21.38	45.30	59.66	83.73	42.47	70.34
No SAE	**36.26** ±0.37	**62.09** ±0.18	**23.59** ±0.12	**47.31** ±0.09	63.06 ±0.40	**85.25** ±0.27	**45.00** ±0.14	**72.02** ±0.15
TEVI	**36.25** ±0.23	**61.98** ±0.18	**23.48** ±0.04	**47.43** ±0.05	**64.22** ±0.37	**85.17** ±0.11	**44.79** ±0.13	**71.74** ±0.17
CLIP ViT-L/14	36.52	62.50	23.85	48.42	63.71	86.49	46.51	72.84
TEVI	**38.41** ±0.24	**64.27** ±0.16	**24.98** ±0.10	**49.75** ±0.15	**65.92** ±0.43	**87.46** ±0.34	**47.92** ±0.10	**74.54** ±0.08
SigLIP ViT-B/16	33.88	60.30	22.06	46.21	62.23	85.40	**44.69**	70.61
TEVI	**37.22** ±0.34	**62.48** ±0.14	**22.53** ±0.06	**47.67** ±0.02	**64.48** ±0.40	**87.08** ±0.21	43.96 ±0.23	**71.75** ±0.08
SharedCLIP ViT-B/16	32.62	58.88	21.54	44.81	**60.75**	84.22	43.31	69.53
TEVI	**34.80** ±0.13	**60.79** ±0.14	**22.15** ±0.11	**46.56** ±0.13	**60.87** ±0.61	**85.50** ±0.42	**44.17** ±0.16	**71.22** ±0.25
AlignCLIP ViT-B/16	32.70	58.92	**21.79**	44.55	57.50	82.35	41.91	70.02
TEVI	**35.53** ±0.13	**61.56** ±0.20	21.63 ±0.16	**45.70** ±0.13	**61.56** ±0.55	**83.85** ±0.32	**42.85** ±0.44	**71.21** ±0.32

Table B2: **Full retrieval results: Fine-grained, long-caption retrieval on DOCCI and IIW.** All the models are trained on CC12M dataset. We report R@1 and R@5 for both image-to-text and text-to-image retrieval. For TEVI we report mean ± std over 5 random seeds. We see that TEVI improves retrieval performance across models and datasets.

Model	DOCCI (Onoe et al., 2024)				IIW (Garg et al., 2024)			
	I → T		T → I		I → T		T → I	
	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5
CLIP ViT-B/16	20.38	42.36	7.17	16.96	50.98	77.94	16.88	32.66
No SAE	**24.03**±0.24	47.95 ± 0.29	**8.60** ± 0.07	**20.18**±0.09	**56.80**±0.82	79.74 ± 0.50	**19.15**±0.13	**36.62**±0.38
TEVI	**24.05**±0.21	**48.48**±0.14	**8.47** ± 0.07	19.96 ± 0.06	**55.75**±0.35	**81.05**±0.59	**19.14**±0.14	**36.36**±0.10
CLIP ViT-L/14	23.60	46.68	8.21	18.70	53.92	81.37	18.05	35.59
TEVI	**26.63**±0.35	**51.55**±0.28	**9.30** ± 0.07	**21.69**±0.04	**58.89**±0.59	**84.67**±0.78	**19.69**±0.08	**37.86**±0.05
SigLIP ViT-B/16	20.68	41.84	7.41	17.17	48.20	76.14	16.94	33.08
TEVI	**24.59**±0.14	**48.50**±0.20	**8.51** ± 0.06	**20.09**±0.05	**57.68**±0.74	**81.18**±0.45	**18.96**±0.13	**36.81**±0.14
SharedCLIP	19.02	41.06	7.16	17.52	51.14	79.08	17.10	33.78
TEVI	**22.28**±0.32	**45.51**±0.29	**7.90** ± 0.04	**18.65**±0.07	**57.22**±0.61	**82.16**±0.61	**17.97**±0.13	**34.77**±0.21
AlignCLIP ViT-B/16	20.32	41.88	7.39	17.43	53.92	82.03	17.46	34.27
TEVI	**22.94**±0.25	**47.18**±0.45	**8.28** ± 0.12	**19.61**±0.15	**58.59**±0.55	**84.48**±0.95	**19.50**±0.29	**36.37**±0.32

Table B3: **Full retrieval results: Coarse-grained, short-caption retrieval on MSCOCO and Flickr30k.** All the models are trained on CC12M dataset. We report R@1 and R@5 for both image-to-text and text-to-image retrieval. For TEVI and No-SAE we report mean ± std over 5 random seeds. We see that TEVI improves retrieval performance across models and datasets.

Model	COCO (Lin et al., 2014)				Flickr30k (Plummer et al., 2015)			
	I → T		T → I		I → T		T → I	
	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5
CLIP ViT-B/16	32.98	59.02	21.38	45.30	59.66	83.73	42.47	70.34
No SAE	**35.79**±0.34	**62.29**±0.25	**23.50**±0.14	**47.37**±0.27	**63.57**±0.59	**85.33**±0.20	**44.92**±0.17	**72.67**±0.25
TEVI	**35.52**±0.23	**62.05**±0.18	23.27 ± 0.09	**47.17**±0.09	**64.54**±0.64	**85.33**±0.26	**44.77**±0.11	71.98 ± 0.12
CLIP ViT-L/14	36.52	62.50	23.85	48.42	63.71	86.49	46.51	72.84
TEVI	**38.08**±0.23	**64.24**±0.10	**24.85**±0.12	**49.70**±0.26	**65.27**±0.45	**87.75**±0.28	**47.79**±0.10	**74.54**±0.14
SigLIP ViT-B/16	33.88	60.30	22.06	46.21	62.23	85.40	**44.69**	70.61
TEVI	**35.93**±0.25	**61.41**±0.27	**22.32**±0.06	**46.66**±0.10	**63.10**±0.25	**85.86**±0.28	43.75 ± 0.22	**71.54**±0.11
SharedCLIP	32.62	58.88	21.54	44.81	**60.75**	**84.22**	43.31	69.53
TEVI	**34.06**±0.15	**60.10**±0.16	**21.92**±0.14	**45.78**±0.11	59.71 ± 0.89	**83.77**±0.51	**43.98**±0.20	**70.81**±0.30
AlignCLIP ViT-B/16	32.70	58.92	**21.79**	**44.55**	57.50	82.35	**41.91**	**70.02**
TEVI	**34.34**±0.24	**60.22**±0.16	21.18 ± 0.19	44.12 ± 0.32	**60.53**±0.50	**83.59**±0.09	**42.34**±0.50	**70.18**±0.34

Table B4: **Coarse-grained retrieval performance on MSCOCO (Lin et al., 2014) and Flickr30k (Plummer et al., 2015) as compared to fine-tuning CLIP.** All the models are trained on CC12M dataset. We report R@1, R@5 and R@10 for both image-to-text and text-to-image retrieval. We see that TEVI improves retrieval performance across models and datasets in comparison to baseline finetuning and random SAE variation as well.

Model	MS COCO (Lin et al., 2014)						Flickr30k (Plummer et al., 2015)					
	I → T			T → I			I → T			T → I		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP ViT-B/16	32.98	59.02	70.04	21.38	45.30	57.03	59.66	83.73	90.14	42.46	70.33	79.31
TEVI	**35.66**	**61.82**	**72.94**	**23.12**	**47.03**	**58.67**	**64.20**	85.70	**91.12**	**44.75**	**72.10**	**81.22**
Random SAE	31.36	56.24	68.04	18.12	40.09	51.87	56.71	81.36	88.36	37.65	65.09	75.80
Fine-tuned	35.60	60.78	71.62	22.50	46.40	58.13	63.21	**86.29**	90.93	44.12	71.50	80.47

the mask preserves a small number with high value. Interestingly, fine-grained datasets (DOCCI, IIW) have more latents preserved as compared to coarse-grained ones (COCO, Flickr), which could be attributed to them having more attributes per caption. We also report the coefficient of variation of mask values, computed per image. A uniform mask would simply scale the embedding, and so the high coefficient of variation (∼0.4) shows that the mask is selective, actively changing the representation's direction based on the text.

Table B5: **Fine-grained retrieval performance on DOCCI (Onoe et al., 2024) and IIW (Garg et al., 2024) as compared to fine-tuning CLIP.** All the models are trained on CC12M dataset. We report R@1, R@5 and R@10 for both image-to-text and text-to-image retrieval. We see that TEVI improves retrieval performance across models and datasets in comparison to baseline finetuning and random SAE variation as well.

Model	DOCCI (Onoe et al., 2024)						IIW (Garg et al., 2024)					
	I → T			T → I			I → T			T → I		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP ViT-B/16	20.38	42.36	53.52	7.16	16.96	22.86	50.98	77.94	**88.89**	16.88	32.66	41.46
TEVI	**24.20**	**48.68**	**60.06**	**8.55**	**19.98**	**26.80**	**55.72**	**81.21**	88.72	**19.37**	**36.44**	**45.47**
Random SAE	21.30	43.54	54.52	6.82	16.97	23.33	54.74	77.94	86.11	17.01	34.11	43.02
Fine-tuned	23.54	46.48	56.92	7.79	18.57	25.00	53.10	80.56	88.23	18.34	35.99	45.20

Table B6: **Robust retrieval performance of TEVI on the RoCOCO benchmark (Park et al., 2024).** We find that TEVI improves performance over the baseline across settings.

Model	COCO	Rand-voca			Same-concept			Diff-concept			Danger		
	R@1	R@1 (↑)	drop rate (↓)	RSMS (↓)	R@1 (↑)	drop rate (↓)	RSMS (↓)	R@1 (↑)	drop rate (↓)	RSMS (↓)	R@1 (↑)	drop rate (↓)	RSMS (↓)
SigLIP	33.66	21.98	11.68	44.16	22.38	11.28	42.12	22.66	11.00	41.84	24.98	8.68	31.96
TEVI	**36.46**	**26.32**	**10.14**	**33.58**	**26.14**	**10.32**	**35.90**	**27.06**	**9.40**	**32.74**	**29.14**	**7.32**	**24.28**
AlignCLIP	32.86	21.46	**11.40**	**43.66**	**22.72**	**10.14**	**40.96**	21.94	**10.92**	41.20	23.44	9.42	37.52
TEVI	**34.70**	**22.32**	12.38	43.70	22.58	12.12	44.16	**22.74**	11.96	**44.48**	**25.48**	**9.22**	**35.20**

Table B7: **Zeroshot performance across benchmarks.** We evaluate on ImageNet-V2 (Recht et al., 2019), ImageNet-Sketch (Wang et al., 2019), ImageNet-A (Hendrycks et al., 2021b), ImageNet-O (Hendrycks et al., 2021b), ImageNet-R (Hendrycks et al., 2021a), and CIFAR-100 (Krizhevsky, 2009).

Model	IMN V2	Sketch	-A	-O	-R	CIF100
CLIP ViT-B/16	**29.24**	**21.93**	**7.64**	**26.20**	42.69	28.07
+TEVI	28.60	**21.93**	7.42	25.90	**42.76**	**30.66**

Table B8: **Coarse-grained retrieval performance of CLIP ViT-B/16 when we do not use the negative conditioning on MSCOCO (Lin et al., 2014) and Flickr30k (Plummer et al., 2015).** All the models are trained on CC12M dataset. We report R@1, R@5 and R@10 for image-to-text and text-to-image retrieval.

Model	MS COCO (Lin et al., 2014)						Flickr30k (Plummer et al., 2015)					
	I → T			T → I			I → T			T → I		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP ViT-B/16	**32.98**	**59.02**	**70.04**	**21.38**	**45.30**	**57.03**	**59.66**	**83.73**	**90.14**	**42.46**	**70.33**	**79.31**
No neg.	10.28	26.24	36.48	9.65	23.77	32.84	26.04	53.65	63.81	22.47	46.07	56.82

Table B9: **Fine-grained retrieval performance of CLIP ViT-B/16 when we do not use the negative conditioning on DOCCI (Onoe et al., 2024) and IIW (Garg et al., 2024).** All the models are trained on CC12M dataset. We report R@1, R@5 and R@10 for image-to-text and text-to-image retrieval.

Model	DOCCI (Onoe et al., 2024)						IIW (Garg et al., 2024)					
	I → T			T → I			I → T			T → I		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP ViT-B/16	**20.38**	**42.36**	**53.52**	**7.16**	**16.96**	**22.86**	**50.98**	**77.94**	**88.89**	**16.88**	**32.66**	**41.46**
No neg.	5.54	14.50	20.88	3.07	8.29	11.90	17.65	38.56	52.12	7.80	19.44	26.82

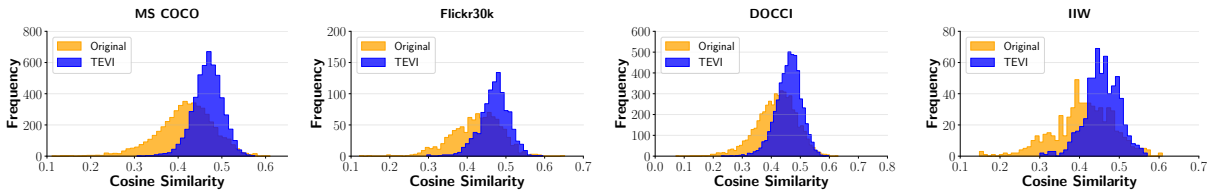


Fig. B2: **Cross-modal alignment** across datasets using CLIP ViT-B/16. We find that the alignment between image-text pairs increases after applying TEVI.

B.9 Additional Qualitative Examples

We provide additional randomly sampled qualitative examples in Fig. B6. For each of the four datasets (DOCCI, IIW, Flickr30k, MS COCO), we provide examples of instances both where TEVI retrieves correctly and the baseline does not, and vice-versa. Note that some of the retrieved captions (e.g. by the baseline model for the first MS COCO example) appear correct but do not match the image as per the dataset, which shows the challenging nature of the benchmarks. We show additional examples in Fig. B5, similar to Fig. 6.

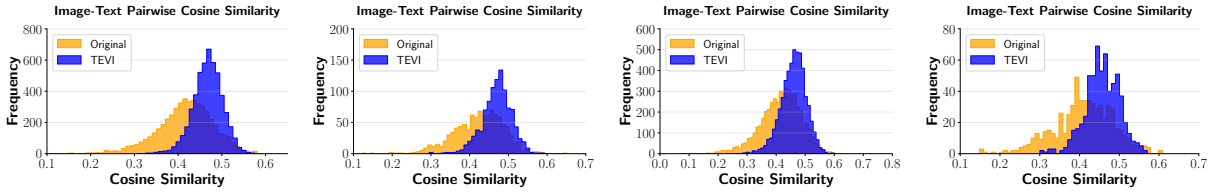


Fig. B3: **Cross-modal alignment with SharedCLIP** on (1) MSCOCO (Lin et al., 2014), (2) Flickr30k (Plummer et al., 2015), (3) DOCCI (Onoe et al., 2024), (4) IIW (Garg et al., 2024). The alignment is measured by the cosine similarity between the positive image-text pairs, and the y-axis denotes the number of data points for each alignment score. We see the distribution after applying TEVI (blue) shifts to the right as compared to the baseline (orange), showing improved alignment.

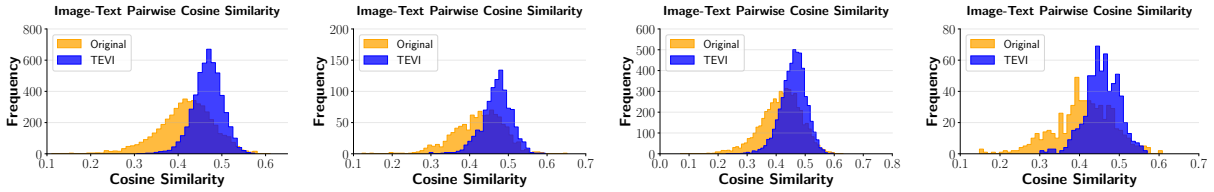


Fig. B4: **Cross-modal alignment with AlignCLIP** on (1) MSCOCO (Lin et al., 2014), (2) Flickr30k (Plummer et al., 2015), (3) DOCCI (Onoe et al., 2024), (4) IIW (Garg et al., 2024). The alignment is measured by the cosine similarity between the positive image-text pairs, and the y-axis denotes the number of data points for each alignment score. We see the distribution after applying TEVI (blue) shifts to the right as compared to the baseline (orange), showing improved alignment.

Table B10: **Fine-grained retrieval performance on DOCCI (Onoe et al., 2024) and IIW (Garg et al., 2024) as compared to SmartCLIP.** All the models are trained on CC12M dataset. We report R@1, R@5 and R@10 for both image-to-text and text-to-image retrieval.

Model	DOCCI (Onoe et al., 2024)						IIW (Garg et al., 2024)					
	I → T			T → I			I → T			T → I		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP ViT-B/16	20.38	42.36	53.52	7.16	16.96	22.86	50.98	77.94	88.89	16.88	32.66	41.46
SmartCLIP	21.76	45.04	55.64	7.47	17.83	23.80	54.74	**82.35**	**89.05**	17.62	33.90	42.15
TEVI	**24.20**	**48.68**	**60.06**	**8.55**	**19.98**	**26.80**	**55.72**	81.21	88.72	**19.37**	**36.44**	**45.47**

Table B11: **Coarse-grained retrieval performance on MSCOCO (Lin et al., 2014) and Flickr30k (Plummer et al., 2015) as compared to SmartCLIP.** All the models are trained on CC12M dataset. We report R@1, R@5 and R@10 for both image-to-text and text-to-image retrieval.

Model	MS COCO (Lin et al., 2014)						Flickr30k (Plummer et al., 2015)					
	I → T			T → I			I → T			T → I		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP ViT-B/16	32.98	59.02	70.04	21.38	45.30	57.03	59.66	83.73	90.14	42.46	70.33	79.31
SmartCLIP	35.54	**61.98**	72.90	**23.89**	**48.24**	**59.93**	61.05	85.11	**91.32**	**45.29**	**72.78**	**81.24**
TEVI	**35.66**	61.82	**72.94**	23.12	47.03	58.67	**64.20**	**85.70**	91.12	44.75	72.10	81.22

Table B12: **Mask sparsity.** Active latents retained and coefficient of variation of the learned mask.

Split	active latents kept	coeff. of variation
MS COCO	8.15 $\pm$4.24	0.456 $\pm$0.062
Flickr30k	8.74 $\pm$4.98	0.429 $\pm$0.060
DOCCI	11.26 $\pm$6.88	0.423 $\pm$0.070
IIW	12.01 $\pm$7.31	0.417 $\pm$0.066

Table B13: **Ablating mask latents by ranking.**

Direction	m	top	random	bottom
I → T (24.20)	32	**8.62**	15.58	22.94
	64	**0.80**	7.24	16.64
	128	0.02	0.02	0.02
T → I (8.55)	32	**4.76**	6.00	8.30
	64	**2.11**	3.75	7.22
	128	0.01	0.01	0.01

Table B14: **Conditioning helps image-to-image retrieval on MS COCO.**

		Δ			Classes	
	Exp.	Orig.	no-SAE	TEVI	no-SAE	TEVI
Δ_1	↓	0.101	0.102	**0.086**	30/61	**52/61**
Δ_2	↑	0.067	0.065	**0.081**	29/61	**45/61**
Δ_3	↓	0.212	0.206	**0.202**	**51/61**	50/61

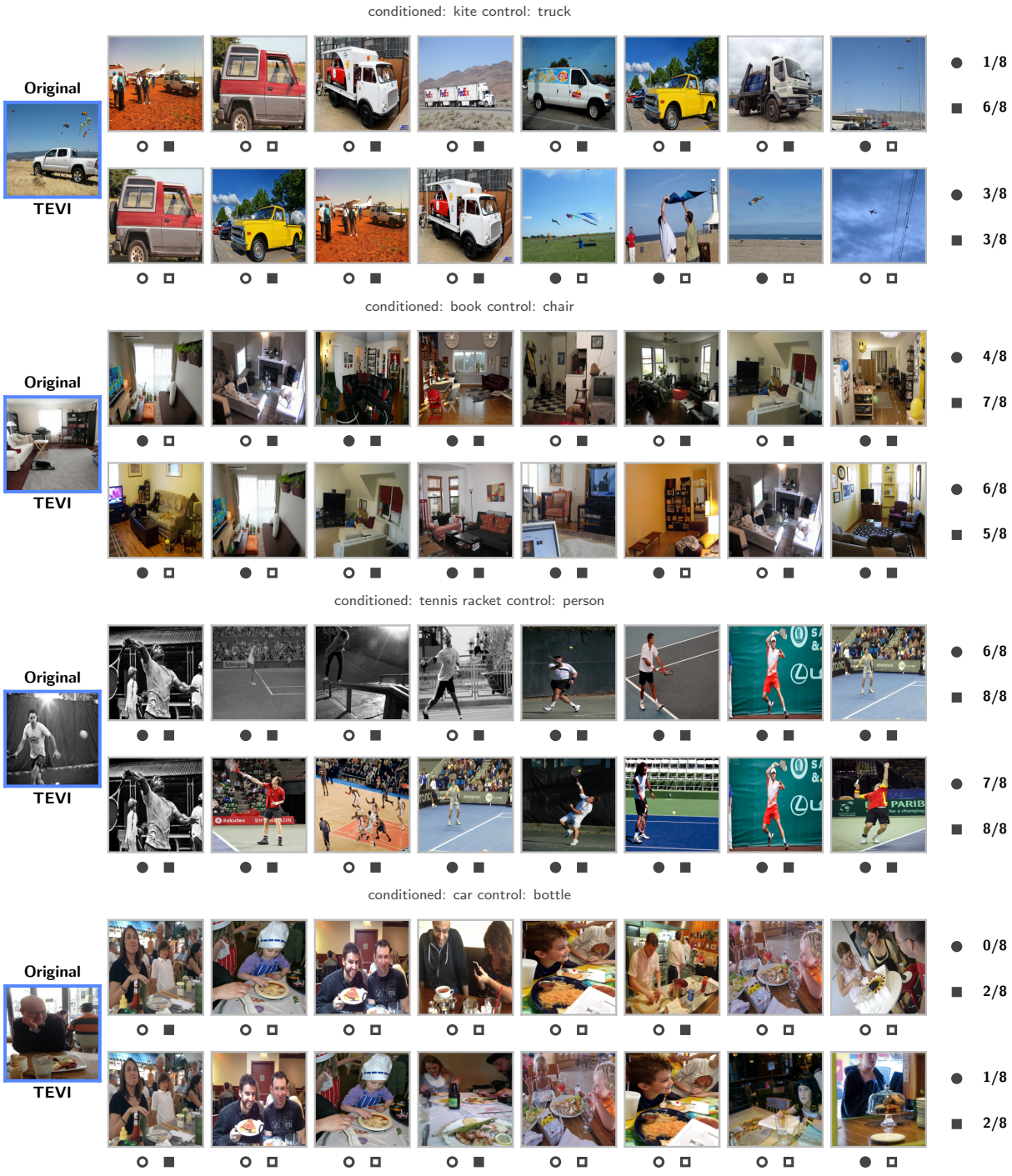


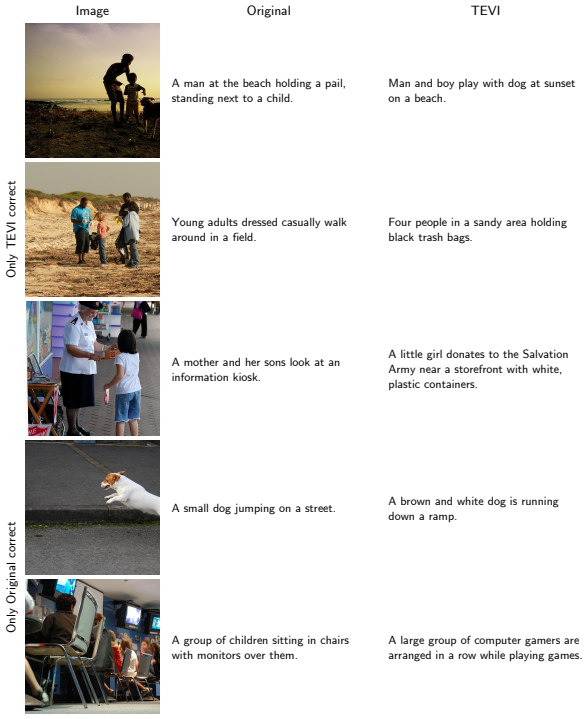
Fig. B5: **Conditioning helps retrieval on MS COCO captions given by attributes.** We show additional examples here. Fig. 6 shows one example at full size.



(a) DOCCI



(b) IIW



(c) Flickr30k



(d) MS COCO

Fig. B6: **Additional qualitative examples.** For each dataset, we show examples of both instances where TEVI retrieves correctly when the baseline does not, and vice-versa. All examples are sampled randomly.

C Broader Impact

The use of vision-language models for multimodal tasks is widespread, from tasks like image retrieval to visual question answering to image generation. This makes it increasingly important that such models work reliably. Lack of proper semantic alignment can lead to biased outputs (Liang et al., 2022) or poor robustness to distribution shifts (Eslami and de Melo, 2025). Improved visual-textual alignment can enhance downstream performance of such models and help alleviate the above mentioned drawbacks. Our work is fundamental research on addressing vision-language alignment and deals with relatively small-scale models, and therefore has low direct risk. However, like many multimodal frameworks, our methods could be co-opted for malicious purposes if deployed without safeguards.

D Licenses of Artifacts Used

We adhere to the licenses provided by the artifacts we use in our work. OpenCLIP and RoCOCO are provided under the MIT license; SharedCLIP and AlignCLIP are released under the CC-BY-NC-ND-4.0 license; DOCCI, MS COCO, IIW are under the CC-BY-4.0 license; and SmartCLIP is released under the Apache 2.0 license. Flickr30k permits use for non-commercial research purposes. Our usage of all artifacts are in accordance with their allowed and intended use.