Intriguing Equivalence Structures of the Embedding Space of Vision Transformers

Shaeke Salman*[1], Montasir Shams*[1], Chashi Mahiul Islam[1], Mao Nishino[2], Xiuwen Liu[1]
[1]Department of Computer Science, Florida State University, Tallahassee, FL, USA
[2]Department of Mathematics, Florida State University, Tallahassee, FL, USA
salman@cs.fsu.edu, mshams@fsu.edu, cislam@fsu.edu, mnishino@fsu.edu, liux@cs.fsu.edu

***Abstract*—Large foundation models play a central role in the recent surge of artificial intelligence, resulting in models with remarkable abilities when measured on benchmark datasets, standard exams, and applications. Due to their inherent complexity, these models are not well understood systematically; in particular, the structures of the representation space are not well characterized despite their fundamental importance. In this paper, using the vision transformers, we show via analyses and systematic experiments that the representation space largely consists of approximately piecewise linear subspaces where there exist very different inputs sharing the same representations, and at the same time, there are visually indistinguishable inputs having very different representations. Furthermore, we show how the vulnerabilities can affect the downstream applications significantly; using image retrieval, visual question answering, and image captioning applications, our results show that the outputs can be altered arbitrarily. Therefore, understanding the inherent risks of applications built on foundational models must consider this and similar vulnerabilities.**

***Index Terms*—Vision Transformers, Equivalence Structures, Representation Learning, Embedding Space, Multimodal Learning, Adversarial Examples, Robustness, Image Retrieval, Visual Question Answering (VQA)**

I. Introduction

Applications leveraging large pre-trained foundation models [1] have demonstrated remarkable capabilities across diverse tasks, achieving breakthrough performance on benchmarks and excelling in professional evaluations - from standardized tests to specialized certification exams [2]–[5]. While transformers have become essential architectural components that have dramatically improved performance across many applications [6]–[8], current understanding of their inner workings remains limited. Although considerable effort has been made to understand transformers [9], a systematic study of the equivalence structures of their underlying representations has yet to be conducted. To comprehend generalization and overgeneralization within a model, it is essential to identify the equivalence classes of inputs that produce the same representation, as downstream applications will treat these inputs equivalently. Additionally, understanding the properties of embeddings for semantically equivalent inputs is crucial: if such inputs yield significantly different representations, the underlying models may struggle to achieve consistent and reliable generalization across applications.

* Denotes Equal Contributions.

In this paper, using multiple datasets and deployed models, we systematically demonstrate that the embeddings utilized by downstream applications are not semantically meaningful. Through a gradient-descent-based optimization procedure, we show that it is possible to perturb an input to a deployed model in imperceptible ways such that its resulting representation matches that of any chosen target. These perturbed inputs can lead to significant changes in application outcomes, all without making any modifications to the model itself. Furthermore, our experimental results reveal two key findings: (1) visually indistinguishable inputs can have vastly different embeddings, and (2) highly distinct images can produce nearly identical embeddings. These findings expose fundamental limitations in relying on model embeddings for semantically meaningful downstream tasks. In addition, we demonstrate the real-world implications of this vulnerability by analyzing its impact on several widely used downstream applications.

Our main contributions are as follows:

- We clearly demonstrate that the embedding space of vision transformers consists of approximately piecewise linear subspaces where different images can share the same representation and visually indistinguishable images can have very different representations.
- We propose an efficient computational procedure for finding equivalence structures of the embedding space and demonstrate their effectiveness on deployed models.
- We show the significant impacts of representation vulnerabilities on deployed models for representative applications where the outputs from deployed models can be manipulated and, therefore, the results can be altered.

II. Related Work

Our work focuses on the vulnerabilities of representations in vision transformers (ViTs). Although deep learning has advanced rapidly on benchmarks, the nature of model-internal representations remains only partially understood. For example, Zhang et al. reveal that deep networks can memorize training sets yet still generalize [10], highlighting gaps in our understanding of how embeddings capture semantic relationships. This gap inspires a more in-depth exploration of the actual structure of underlying representation spaces and whether they might exhibit problematic equivalences.

Some prior works study equivalence structures or collisions in neural representations, although often in more limited

contexts [11]. Early research on classification tasks revealed that minimal input perturbations can enable targeted misclassification attacks [12], [13]. Shao et al. extend these findings to vision transformers, showing that such vulnerabilities persist in newer architectures as well [14]. However, such adversarial attacks typically target particular application endpoints, such as misleading a classifier to produce an incorrect label. Consequently, the techniques developed in those studies often do not transfer beyond classification. Recently, researchers have studied the robustness of multimodal models via an adversarial attack on visual inputs [15], [16]. In contrast, our work focuses on representation vulnerabilities: we show that inputs can be made nearly indistinguishable in pixel space yet occupy distant points in embedding space, or map to the same embedding despite large visual differences, affecting any downstream application that relies on those embeddings.

In terms of broader limitations and weaknesses of deep learning, many researchers have noted the fragility of high-dimensional feature spaces, e.g., adversarial examples revealing linear regimes in seemingly nonlinear models [12], [17]. Bhojanapalli et al. and Shao et al. investigate the robustness of ViTs against attacks where the attacker has access to the model's internal structure [14], [18]. A recent work [19] explores the interplay between alignment techniques and adversarial attacks in neural networks, highlighting the potential vulnerabilities of aligned models. However, existing attack methods usually target a single task (e.g., image recognition or sentiment classification). Our contributions differ in both scope and method: we propose a systematic gradient-based procedure to locate and exploit representation vulnerabilities across tasks, rather than designing application-specific perturbations. As we demonstrate, once an embedding is forced to align (or misalign) with a desired target, downstream applications (e.g., image retrieval, visual question answering, image captioning) inherit the misalignment, regardless of the model's original performance on benchmarks. By examining these representation vulnerabilities, our findings underscore that representation-level manipulations pose a broad risk, calling into question the robustness of systems that rely on pretrained embeddings without deeper scrutiny of their internal structures.

III. Proposed Framework

Here we describe the framework that enables us to explore the embedding space, analyze their properties, and verify them in large models. We conceptualize the neural network's representation (including transformer architectures) as a mathematical mapping f that takes inputs from an *m-dimensional* space and transforms them into outputs in an *n-dimensional* space, written as a function $f : \mathbb{R}^m \to \mathbb{R}^n$. The core challenge is to develop a computationally efficient and effective method for exploring input embeddings in the representation space. Specifically, the goal is to identify inputs whose embeddings align closely with the representation produced by $f(x_{img})$, where x_{img} is the target input with the desired embedding. To put it simply, consider an image of a lizard in Fig. 2 as an example, any image that shares the same representation, as determined by a model, will also be identified as a lizard.

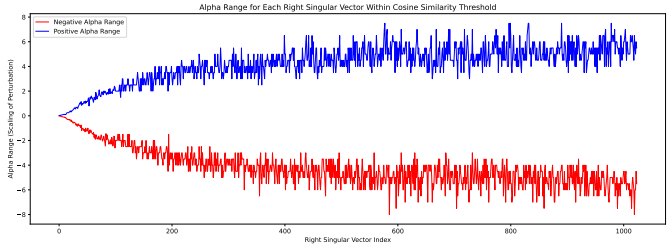

Fig. 1: Range along both the positive and negative directions of right singular vectors where the difference between model output and linear approximation remains small.

A. Embedding Matching Procedure

The core of the proposed method involves an iterative gradient optimization process, similar to the training of neural networks, but with one key difference: the gradient is computed with respect to the input variables rather than the model parameters [20]. Since we need to match two vectors, we define the loss for finding an input matching a given representation as

$$L(x) = L(x_0 + \Delta x) = \frac{1}{2}\|f(x_0 + \Delta x) - f(x_{img})\|^2, \quad (1)$$

where x_0 is an initial input and $f(x_{img})$ specifies the target embedding, and and Δx is a small perturbation to x_0, which acts as a variable of our optimization function. The gradient of the loss (1) with respect to Δx is

$$\nabla_{\Delta x} L = f'(x_0 + \Delta x)^T (f(x_0 + \Delta x) - f(x_{img})). \quad (2)$$

where $f'(x_0 + \Delta x)$ denotes the Jacobian matrix of a function f at $x_0 + \Delta x$. For numerical calculation, we will approximate the Jacobian by the one at x_0, considering that Δx is small. The loss defined by Eq. (1) is related to the similarity of representations as measured by the cosine similarity. Indeed, by expanding the squared norm, we obtain

$$\|f(x_0+\Delta x)\|^2 - 2\langle f(x_0+\Delta x), f(x_{img})\rangle + \|f(x_{img})\|^2 \quad (3)$$

where $\langle x, y\rangle$ denotes the Euclidean inner product. As $f(x_{img})$ is constant, minimizing this loss would be a combination of minimizing $\|f(x_0 + \Delta x)\|$ and maximizing $\langle f(x_0 + \Delta x), f(x_{img})\rangle$, which would attempt maximizing

$$\frac{\langle f(x_0 + \Delta x), f(x_{img})\rangle}{\|f(x_0 + \Delta x)\|} \quad (4)$$

which is a constant multiple of the cosine similarity between $f(x_0 + \Delta x)$ and $f(x_{img})$.

In each step, our optimization algorithm begins by calculating the loss, defined by Eq. 1, which measures the difference between the input image embedding and the target image embedding. Next, the algorithm performs backpropagation

to compute the gradient. Once the gradient is obtained, the pixel values are updated using gradient descent. We call this procedure as embedding matching procedure.

Eq. 2 shows how the gradient of the mean square loss function is related to the Jacobian of the representation function at $x = x_0$. To see how the difference and the gradient relate, we use the singular value decomposition of the Jacobian:

$$f'(x_0 + \Delta x) = U\Sigma V^T \tag{5}$$

where U, V are orthogonal matrices of left and right singular vectors, respectively, and Σ is the diagonal matrix of singular values. Substituting this to the gradient, we obtain

$$\nabla_{\Delta x} L = V\Sigma U^T \Delta f. \tag{6}$$

Here, we denote $\Delta f = f(x_0 + \Delta x) - f(x_{img})$. The above equation projects the difference onto the vector space spanned by the left singular vectors, scales each component by the singular values, and returns it to the original space by the right singular vectors. Therefore, the gradient is large when aligned with the left singular vectors with the largest singular values.

To analyze our algorithm further, we will see how the representation changes after a gradient descent step. Consider the following numerical approximation of the representation after an update:

$$f(x_0 + \Delta x + \delta) = f(x_0 + \Delta x) + f'(x_0 + \Delta x)\delta + o(\delta) \tag{7}$$

Here, $\delta = -\eta \nabla_{\Delta x} L$ is the increment in the update, and $o(\delta)$ is the error term for the approximation. Substituting Eq. (6) to (7), $f(x_0 + \Delta x + \delta)$ is equal to

$$f(x_0 + \Delta x) - \eta U\Sigma U^T \Delta f + o(\delta) \tag{8}$$

Therefore, $U^T f(x_0 + \Delta x + \delta)$ is equal to

$$U^T f(x_0 + \Delta x) - \eta \Sigma U^T \Delta f + U^T o(\delta) \tag{9}$$

Let us call $U^T f(x)$ a projected representation, as this is the representation projected to the space spanned by the left singular vectors. Eq. (9) shows that the gradient descent in the image space approximately corresponds to the gradient descent in the projected representation space. Notice that since U is orthogonal, our loss function can be written as

$$L = \frac{1}{2} \| U^T f(x_0 + \Delta x) - U^T f(x_{img}) \|^2 \tag{10}$$

so that the gradient with respect to the projected representation is simply the difference of projected representations, i.e., $U^T \Delta f$. We empathize that optimizing our loss function in the projected representation space is the same as optimizing a simple function $\frac{1}{2}\|x - a\|^2$ with respect to x. This partially explains why our optimization works well, although we do not capture the full picture since it remains an approximation. Moreover, we note that U, Σ, and V could be different for different values of Δx.

Consistent with this analysis, we successfully minimized the loss in all the examples we have tested. Comprehensive details are provided in the experimental results section.

Our technique requires a good linear approximation of the underlying model. For vision transformers, we show this using a first-order linear approximation through the Jacobian matrix. Figure 1 shows a representative example illustrating the range along both the positive and the negative directions of the right singular vectors (defined in Eq. 5), where the difference between model output and linear approximation remains small (measured by cosine similarity [1]); highlighting that the linear approximation works very well [2].

IV. Experiments and Results

In this section, we begin by providing the specifics of our experimental settings, implementation details, and results. Our proposed framework is systematically applied across various datasets and multiple vision transformer models; in the subsequent subsections, we present both the experimental outcomes and quantitative results. More importantly, we show that our framework exhibits versatility, being agnostic to both the model architecture and dataset characteristics.

A. Experiment Setup

Datasets We perform extensive experiments to assess our proposed framework using examples (randomly selected) from publicly available vision datasets, specifically, ImageNet [21], MS-COCO [22] and Google Open Images [23].

Implementation Details. To demonstrate the feasibility of the proposed method on large models, we have used the pre-trained model publicly available by ImageBind,[3] which in turn uses a CLIP model [4]. More specifically, ImageBind utilizes the pre-trained vision (ViT-H 630M params) and text encoders (302M params) from the OpenCLIP [24], [25]. The input size is $224 \times 224 \times 3$, and the dimension of the embedding is 1024. To clarify, in our experiments, we maintain the pre-trained weights of both visual and text encoders without modification, only optimizing the perturbations applied to input images. This design choice ensures that the learned perturbations specifically target the embedding space without altering the underlying model parameters. Apart from the OpenCLIP, we have also used BLIP-2 model [26] for experiments. All experiments are conducted on a workstation in our lab, equipped with two NVIDIA A5000 GPUs. The code for the proposed framework and experiments can be found in this GitHub repository [5].

Computational Cost. The computational complexity of each gradient calculation iteration scales with the embedding dimension. Specifically, doubling the embedding dimension increases computation time by a factor between 2 and 4,

[1]Cosine similarity threshold we have used is 0.95

[2]There are only a few directions where the range of the linear approximation is relatively small which donot impact on our result significantly

[3]https://github.com/facebookresearch/ImageBind

[4]https://github.com/mlfoundations/open_clip

[5]https://github.com/programminglove08/EquivalenceStruct

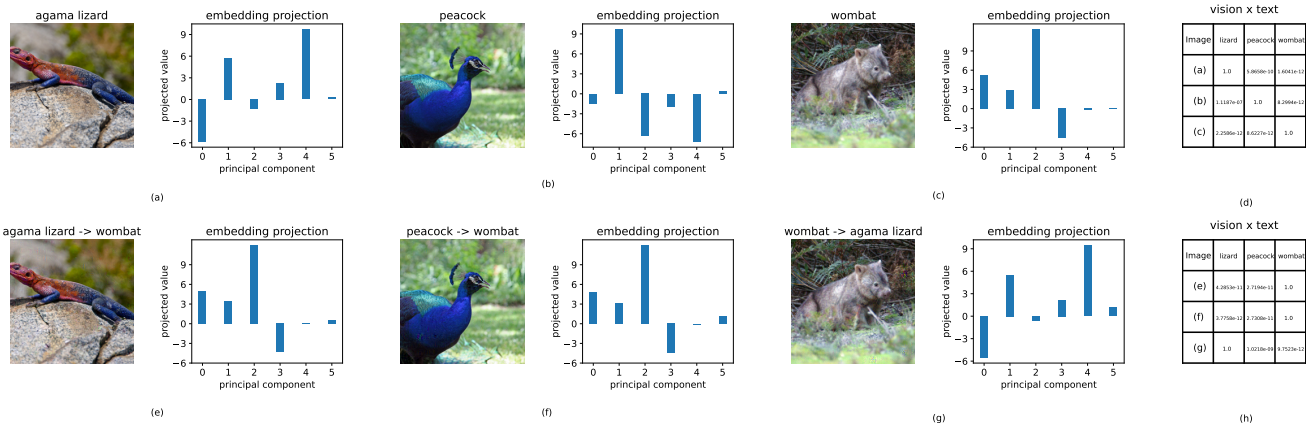

Fig. 2: Typical examples from ImageNet obtained using the proposed framework. Three pairs of visually indistinguishable images (a and e, b and f, c and g) have different representations from each other as shown in their low-dimensional projections. In contrast, very similar representations are seen for the images in (e), (f), and (c), despite their substantial semantic differences; similar goes with images in (a) and (g). Note that the arrow in the title ($original \rightarrow target$) signifies a derived image from the original one by aligning the embedding of the original image with the target image using our method. The matrices (d) and (h) show the classification outcomes from the multimodal ImageBind pre-trained model used directly with no modifications.

depending on how projection matrices are used to match the given dimension. Our empirical analysis supports this relationship by using vision transformers of increasing size, we have observed:- ViT-B/16 (512-dimensional embeddings): 0.6 ms per iteration; ViT-L/14 (768-dimensional embeddings): 1.1 ms per iteration; ViT-H/14 (1024-dimensional embeddings): 2.15 ms per iteration.

Embedding Projections. To obtain the low-dimensional projections of the images shown in Fig. 2 and other similar figures, we employ a dimensionality reduction approach based on Principal Component Analysis (PCA). We first compute the principal components from a representative subset of images from each dataset. The embeddings are then projected onto the six principal components corresponding to the largest eigenvalues. It is worth noting that these projections serve primarily as a visualization tool. The specific details of the principal components are not critical to interpreting the results, as the fundamental relationships between embeddings are preserved: similar embeddings will cluster together in the projected space, while dissimilar embeddings will remain separated.

B. Experimental Results

We have tested the embedding matching procedure using many image pairs. To highlight the key results of our framework, we use the ImageBind model as an example [25]. Fig. 2 shows several images along with their representations and the classification results. The three visually indistinguishable pairs in Fig. 2, (a) and (e), (b) and (f), and (c) and (g), respectively, have very different representations, as shown by their low-dimensional projections. On the other hand, the images in (e), (f), and (c) have very similar representations even though they are semantically very different; the images in (a) and (g) show another set. When we pass these images to the unmodified multimodal ImageBind model, the images with similar embeddings are classified into the same class, regardless of their semantic similarity, as shown in Fig. 2 (d) and (h).

Fig. 3 shows a typical example, where the left one shows the evolution of loss when matching a specified target embedding. We use a small step size to make sure it converges. The right one shows that cosine similarity increases steadily. We also show the average pixel value difference between the new input and the original image at each step; one can see the values remain very small even though they increase as well. The algorithm is not sensitive to the learning rate and works effectively across a broad range of values, spanning from 0.001 to 0.09. For instance, with a learning rate of 0.001, convergence is achieved in around 25,000 iterations, while 0.09 requires around 3,000 iterations. The visual differences in the resulting images are not noticeable.

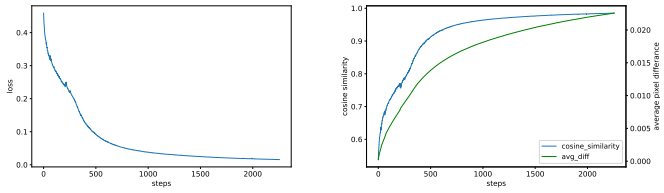

Fig. 3: The evolution of loss while matching a target embedding. (left) the loss w.r.t. steps. (right) the cosine similarity between the embeddings of the new input and the target w.r.t. the steps, along with the average pixel value difference between the new input and the original image.

Qualitative evaluation. Our key result is that the semantic meanings of the embeddings given by transformer models are fundamentally limited as different inputs share similar embeddings while visually indistinguishable inputs have very different embeddings. As the techniques are model and dataset-

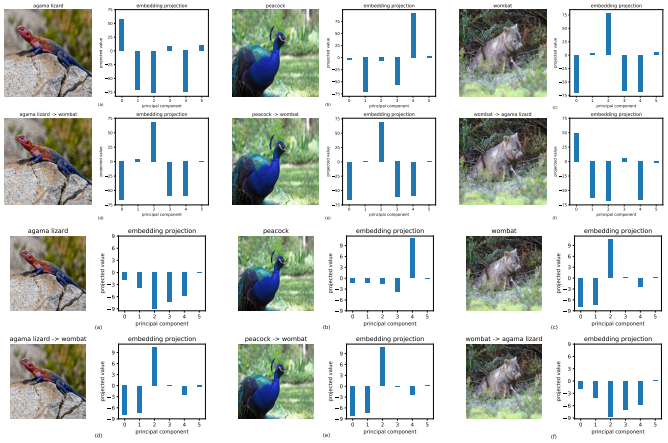

Fig. 4: Examples obtained while the proposed framework is applied on different vision transformer models, such as (top two rows) BEiT, and (the next two rows) Swin Transformer. The results are given in the same format as depicted in Fig. 2. Additional plots for other models are also consistent and added to the github due to the page limit. The example demonstrates that the method is model-agnostic.

agnostic, they should be effective on different transformer models and datasets, including ones for other modalities. We have conducted experiments with various vision transformer models, including MAE-like models from HuggingFace,[6] such as BEiT, DeiT, Swin, ViTMAE, ViTMSN [27]–[31] and two examples are given in Fig. 4.

In general, as shown in the examples, the proposed technique works well with any randomly chosen image from a different target class.

Exploring Representation Space. So far, we have shown the structures of the embedding space of a particular point using concrete examples. Our framework allows us to explore the paths and the space more broadly. Fig. 5 shows how the embeddings change as the input changes from one image to the one that matches a specified target but remains visually indistinguishable. The plot shows the changes roughly linearly.

Compared to existing adversarial attack methods, one distinctive feature of our proposed framework is that we can exploit how different subspaces are connected. Fig. 6 shows one such path example. By applying the match-finding procedure, we are able to construct and connect different subspaces. The results show that the embeddings are inherently limited semantically when analyzed systematically. Locally, the model is sensitive to small changes along the directions in the normal space. In the null space, the embeddings remain constant while the input changes substantially. By connecting local null spaces, we have connected spaces where embeddings are similar, but inputs can be very different. As the embedding space is high dimensional in nature, testing using datasets is inherently limited. The systematic analyses are essential.

[6]https://huggingface.co/docs/transformers/model_doc/beit

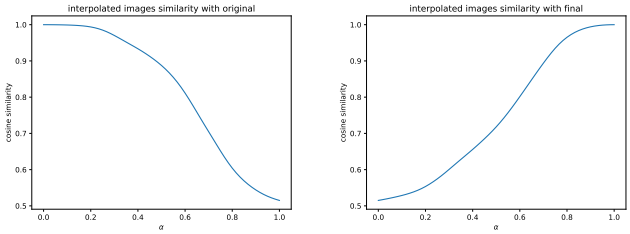

Fig. 5: The change of the embeddings as the input changes linearly for the image pair in Fig. 2(a). (left) cosine similarity between embeddings of the interpolated and the original. (right) same as left but for the final image.

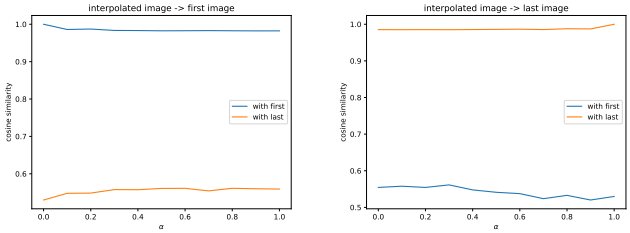

Fig. 6: (Left) The cosine similarity along the path that matches the embedding of the first image. (Right) Same as the left except along the path for that of the last image.

Adversarial Modification Detection. We have observed that the embedding-matched images exhibit much higher sensitivity to Gaussian noise than the original ones. Leveraging this insight, we have designed a detection algorithm. The process is as follows: we add Gaussian noise of a specified standard deviation to a given image and then classify them. If the labels of the two images agree, the image is unmodified; otherwise, the image is modified. The detection algorithm works reliably and consistently for a wide range of standard deviations (Due to anonymity requirements, we refrain from citing our previous work for details. However, as an example, you can refer to Islam et al., who have successfully applied the same approach to the CLIP-based LM-Nav visual navigation system using the real-world RECON robot dataset to detect modifications [32]).

V. Applications and Impacts of Representational Vulnerability

The embedding space vulnerabilities we have identified have significant implications for downstream vision-language tasks. In this section, we demonstrate how these representational vulnerabilities can impact three widely used applications: Image Retrieval, Visual Question Answering (VQA) and Image Captioning. Our analysis reveals how visually imperceptible modifications that align an image's embedding with that of a different target image can dramatically alter the model's behavior across these tasks.

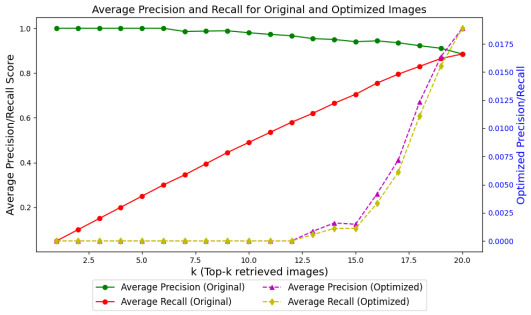

Fig. 7: Comparison of average Precision and Recall for original and optimized Images across top-k retrieved images. The primary y-axis shows the average precision and recall for the original images (green and red), while the secondary y-axis highlights the average precision and recall for the optimized images (magenta and yellow). The figure exhibits a stark difference in performance metrics.

A. Image Retrieval

Representation matching procedures, as well as representation vulnerabilities, have shown a significant effect on the image retrieval task. In our experiments, we consider 200 randomly selected images from 10 different classes of the Imagenette[8] dataset, which is a subset of ImageNet. For each class, we generate nine optimized images, and these optimized images are used as query images. We employ a top-k image retrieval algorithm to retrieve the most relevant images from the dataset. The evaluation process is conducted using several metrics, including Recall@k and Precision@k.

Fig. 7 shows the average Precision@k and Recall@k for all the optimized query images from Imagenette dataset across all classes. It demonstrates that the system's retrieval performance exhibits a steady decline in precision and recall as k increases, reflecting the challenges of maintaining relevance when retrieving more images. This highlights the potential impact of representation vulnerabilities on the retrieval task. On the contrary, the red and the green line represents the average Precision@k and Recall@k for ten original query images. The retrieval performance for original images showcases better precision and recall scores across all k, indicating that optimized images introduce additional challenges for the retrieval system.

By assessing the performance of the image retrieval system using these metrics and comparing optimized and original query images, we gain valuable insights into the effectiveness of the representation matching procedure. These results emphasize the need to improve the robustness and accuracy of image retrieval systems, particularly in the presence of representation vulnerabilities introduced by optimized query images.

B. Visual Question Answering

1) Experimental Setup: We have designed a comprehensive evaluation framework to assess the impact of representation alignment attacks on Visual Question Answering (VQA) performance. We utilize the VQA v2[9] validation dataset as our foundation. To create a balanced evaluation set, we first analyze the distribution of question types and select the top 10 most frequent categories. From each category, we randomly sample 50 question-image pairs, resulting in a curated subset of 500 samples that ensures equal representation across question types. We employ the BLIP-2 model as our base architecture to generate optimized examples.

2) Evaluation Metrics: We evaluate VQA performance using the standard VQA accuracy metric [33], which is based on the matches to multiple human annotations. For a predicted answer, the accuracy is considered to be 100% if it agrees with three or more human annotations; otherwise, it is the number of matches divided by 3.

3) Results and Analysis: Our evaluation compares performance across original and optimized images. Pairs of images (original and optimized) have been considered along with a question and reported their corresponding answer. Fig. 8 clearly demonstrates the impact of representation vulnerabilities on VQA system. Despite the images being visually indistinguishable, the system produces drastically different answers for each pair: a garbage truck is misidentified as a fish in a net, green apples are described as sheep, and a speed limit sign generates two distinct interpretations. The answers, highlighted in red, illustrate how subtle perturbations can cause significant changes in the model's responses while maintaining the visual integrity of the images. Table I presents the detailed results of the analysis.

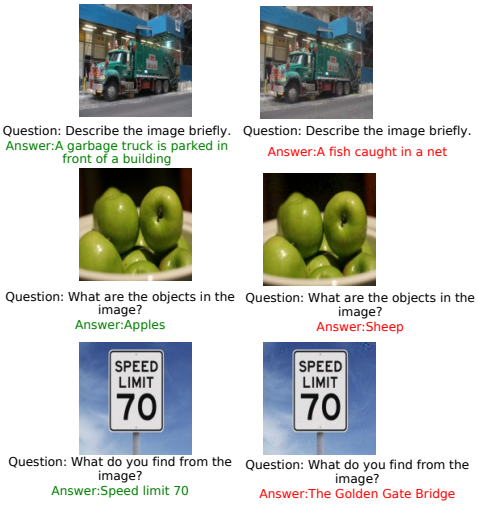

Fig. 8: Demonstration of representation vulnerabilities on Visual Question Answering (VQA) systems. Each row shows pairs of visually identical images (left: original, right: optimized) along with the questions and their corresponding VQA outputs. Original target images are added to the github.

[8]https://github.com/fastai/imagenette

[9]https://visualqa.org/

TABLE I: VQA Performance Comparison Between Original and Optimized Images

Question Type	Original Accuracy	Optimized Accuracy
how many	45.33%	**23.33%**
is the	78.00%	**71.33%**
what	36.00%	**10.67%**
what color is the	42.67%	**24.67%**
what is the	28.00%	**8.67%**
none of the above	36.00%	**28.00%**
is this	70.00%	**58.00%**
is this a	72.00%	**70.67%**
what is	33.00%	**5.33%**
what kind of	37.33%	**16.67%**
Overall Mean	47.93%	**31.73%**

TABLE II: Caption Quality Metrics Comparison

Metric	Original Images	Optimized Images
CLIP Score	25.72	18.27
BERTScore F1	0.92	0.89
ROUGE-L	0.44	0.24
BLEU-1	0.72	0.39
BLEU-4	0.29	0.09

The results reveal several key findings:

- The overall mean accuracy drops significantly by 33.79% (relative) on optimized images, demonstrating the effectiveness of the representation alignment procedure.
- Questions about visual attributes (e.g., "what is the," "what," and "what is") show the largest performance degradation, with accuracy (relative) drops of 69.03%,70.36%, and 83.84%, respectively..
- For counting-related questions ("how many"), the performance is moderately unstable (random) on original images The underlying reason for this behavior is not yet fully understood and remains an area of ongoing investigation.

These findings indicate that representation alignment attacks can significantly impact VQA system performance, with particularly strong effects on questions requiring detailed visual attribute recognition.

C. Image Captioning

Our results, presented in Table II, demonstrate significant differences between original and optimized images:

- **Semantic Understanding:** While BERTScore shows only a modest decrease (4%), suggesting some preservation of semantic content, other metrics show more substantial degradation;
- **Sequential Patterns:** ROUGE-L drops by 45% and BLEU-4 by 70%, indicating severe disruption of phrase-level patterns;
- **Image-Text Alignment:** CLIPScore decreases by 29%, showing significant disruption in the model's ability to align captions with visual content.

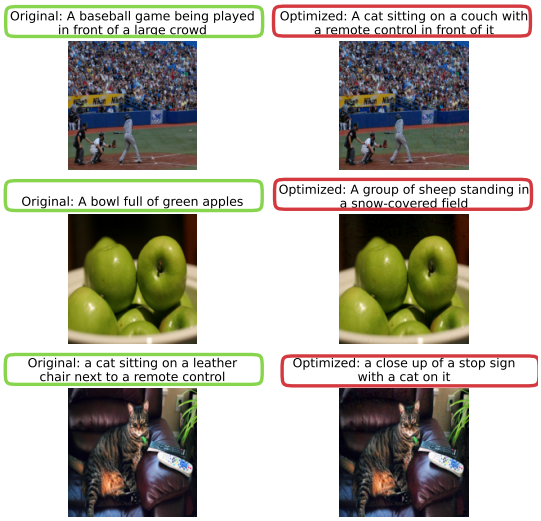

Fig. 9: Comparison of original and optimized image captioning outputs on three test images. Each row shows the input image paired with its caption (left) and the optimized image paired with caption (right), demonstrating significant semantic differences between the two captioning approaches despite being visually indistinguishable. Original target images are added to the github.

These results quantitatively demonstrate that embedding alignment procedure can significantly impact caption generation while maintaining visual similarity. The disproportionate impact on higher-order metrics (BLEU-4) compared to simpler ones (BLEU-1) suggests that this type of approach particularly affects the model's ability to generate coherent, complex descriptions. Fig. 9 clearly demonstrates the impact of representation vulnerabilities on image captioning through multimodal models. These applications highlight the impact of the fundamental representational vulnerability on the performance of deployed models.

While some may categorize our framework as an adversarial attack technique, our primary focus is on analyzing the embedding space and its vulnerabilities. Unlike traditional adversarial attacks [12], which are application-specific and target specific classifiers, our technique is application-agnostic. Traditional adversarial attacks aim to make minimal changes to an input to force the target model into misclassifying it; for instance, slightly altering an image of a "lizard" to make the model misclassify it as a "wombat."

In contrast, our approach extends beyond causing misclassification. It involves modifying an input sample not only to misclassify but also to precisely align its underlying representation in the embedding space with that of a specific target sample (e.g., a specific image of a "wombat"). This requires a deeper understanding and more precise manipulation of the model's embedding space. The challenge lies in ensuring that the transformed representation of the input matches the embedding of the target sample, even when the two samples

are visually and semantically distinct. Therefore, our approach differs fundamentally from traditional adversarial techniques, like Fast Gradient Sign Method (FGSM) or Projected Gradient Descent (PGD), which target classification boundaries while we manipulate the underlying representation space itself.

VI. Potential Mitigation and Future Work

Given that the limitations are intrinsic to the representations, it is crucial to acknowledge that commonly employed robustness training will not effectively address the issue of change of classifications; for example, Mao [34] demonstrate that modifying the last linear layer does not resolve the problem. A potential strategy to mitigate such modifications in practical applications might involve implementing smooth detection mechanisms. Modified images may display distinctive characteristics due to gradient descent optimization. For instance, singular value distributions of the Jacobian matrices of modified images reveal significant discrepancies when compared to those of the original natural images. The differences in singular values could be used to identify modified images, which is being investigated.

VII. Conclusion

In this paper, we have identified a new representation vulnerability of the vision transformers using algorithms and mathematical analyses. It is attempting to conclude that recent pre-trained transformer models can be used to build any effective applications based on their performance on benchmark datasets. While such models give impressive performance, their inherent generalization abilities are limited by the properties of the underlying embedding spaces. In particular, as representations of images with imperceptibly small changes can be very different and at the same time semantically unrelated images can have same representations, the outputs for applications can be altered arbitrarily demonstrated using image retrieval, visual-question-answering, and image captioning. Before this fundamental limitation can be addressed, such models should not be used for critical applications.

References

[1] R. Bommasani, D. A. Hudson, E. Adeli, Others, and et al., "On the opportunities and risks of foundation models," *arXiv e-prints*, 2022.

[2] OpenAI, "GPT-4 Technical Report," *arXiv e-prints*, 2023.

[3] N. Brandes, D. Ofer, Y. Peleg, N. Rappoport, and M. Linial, "ProteinBERT: a universal deep-learning model of protein sequence and function," *Bioinformatics*, vol. 38, pp. 2102–2110, 02 2022.

[4] T. H. Kung, M. Cheatham, A. Medenilla, and O. et al., "Performance of chatgpt on usmle: Potential for ai-assisted medical education using large language models," *PLOS Digital Health*, 2023.

[5] J. H. Choi, K. E. Hickman, A. Monahan, and D. B. Schwarcz, "Chatgpt goes to law school," *Journal of Legal Education (Forthcoming)*, 01 2023.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv e-prints*, 2023.

[7] A. Dosovitskiy, L. Beyer, and A. K. et al., "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv e-prints*, 2021.

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv e-prints*, 2018.

[9] D. Biś, M. Podkorytov, and X. Liu, "Too much in common: Shifting of embeddings in transformer language models and its implications," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2021.

[10] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning (still) requires rethinking generalization," *Commun. ACM*, vol. 64, p. 107–115, Feb. 2021.

[11] U. Ozbulak, M. Gasparyan, S. Rao, W. D. Neve, and A. V. Messem, "Exact feature collisions in neural networks," 2022.

[12] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv e-prints*, 2014.

[13] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv e-prints*, 2015.

[14] R. Shao, Z. Shi, J. Yi, P.-Y. Chen, and C.-J. Hsieh, "On the adversarial robustness of vision transformers," *arXiv e-prints*, 2022.

[15] C. Schlarmann and M. Hein, "On the adversarial robustness of multimodal foundation models," 2023.

[16] W. Zhou, S. Bai, D. P. Mandic, Q. Zhao, and B. Chen, "Revisiting the adversarial robustness of vision language models: a multimodal perspective," 2024.

[17] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," *arXiv e-prints*, 2017.

[18] S. Bhojanapalli, A. Chakrabarti, D. Glasner, D. Li, T. Unterthiner, and A. Veit, "Understanding robustness of transformers for image classification," *arXiv e-prints*, 2021.

[19] N. Carlini, M. Nasr, C. A. Choquette-Choo, M. Jagielski, I. Gao, A. Awadalla, P. W. Koh, D. Ippolito, K. Lee, F. Tramer, and L. Schmidt, "Are aligned neural networks adversarially aligned?," *arXiv e-prints*, 2023.

[20] S. Salman, M. Shams, M. Nishino, and X. Liu, "Unaligning everything: Or aligning any text to any image in multimodal models," in *2025 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–9, 2025.

[21] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

[22] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft coco: Common objects in context," *arXiv e-prints*, 2015.

[23] A. Kuznetsova, H. Rom, N. Alldrin, and J. U. et al., "The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale," *IJCV*, 2020.

[24] G. Ilharco, M. Wortsman, R. Wightman, C. Gordon, N. Carlini, R. Taori, A. Dave, V. Shankar, H. Namkoong, J. Miller, H. Hajishirzi, A. Farhadi, and L. Schmidt, "Openclip," 2021.

[25] R. Girdhar, A. El-Nouby, Z. Liu, M. Singh, K. V. Alwala, A. Joulin, and I. Misra, "Imagebind: One embedding space to bind them all," *arXiv e-prints*, 2023.

[26] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," 2023.

[27] H. Bao, L. Dong, S. Piao, and F. Wei, "Beit: Bert pre-training of image transformers," *arXiv e-prints*, 2022.

[28] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," *arXiv e-prints*, 2021.

[29] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," *arXiv e-prints*, 2021.

[30] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," *arXiv e-prints*, 2021.

[31] M. Assran, M. Caron, I. Misra, P. Bojanowski, F. Bordes, P. Vincent, A. Joulin, M. Rabbat, and N. Ballas, "Masked siamese networks for label-efficient learning," *arXiv e-prints*, 2022.

[32] C. M. Islam, S. Salman, M. Shams, X. Liu, and P. Kumar, "Malicious path manipulations via exploitation of representation vulnerabilities of vision-language navigation systems," 2024.

[33] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh, "Vqa: Visual question answering," in *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015.

[34] C. Mao, S. Geng, J. Yang, X. Wang, and C. Vondrick, "Understanding zero-shot adversarial robustness for large-scale models," 2023.