

Demystify Self-Attention in Vision Transformers from a Semantic Perspective: Analysis and Application

Leijie Wu¹, Song Guo¹, Yaohong Ding¹, Junxiao Wang¹, Wenchao Xu¹,
Richard Yida Xu², and Jie Zhang¹

¹Department of Computing, The Hong Kong Polytechnic University

²Department of Mathematics, The Hong Kong Baptist University

{lei-jie.wu, yaohong.ding, jieaa.zhang}@connect.polyu.hk

{song.guo, junxiao.wang, wenchao.xu}@polyu.edu.hk, xuyida@hkbu.edu.hk

Abstract

Self-attention mechanisms, especially multi-head self-attention (MSA), have achieved great success in many fields such as computer vision and natural language processing. However, many existing vision transformer (ViT) work simply inherent transformer designs from NLP to adapt vision tasks, while ignoring the fundamental difference between “how MSA works in image and language settings”. Language naturally contains highly semantic structures that are directly interpretable by humans. Its basic unit (word) is discrete without redundant information, which readily supports interpretable studies on MSA mechanisms of language transformer. In contrast, visual data exhibits a fundamentally different structure: Its basic unit (pixel) is a natural low-level representation with significant redundancies in the neighbourhood, which poses obvious challenges to the interpretability of MSA mechanism in ViT. In this paper, we introduce a typical image processing technique, i.e., scale-invariant feature transforms (SIFTs), which maps low-level representations into mid-level spaces, and annotates extensive discrete keypoints with semantically rich information. Next, we construct a weighted patch interrelation analysis based on SIFT keypoints to capture the attention patterns hidden in patches with different semantic concentration. Interestingly, we find this quantitative analysis is not only an effective complement to the interpretability of MSA mechanisms in ViT, but can also be applied to 1) spurious correlation discovery and “prompting” during model inference, 2) and guided model pre-training acceleration. Experimental results on both applications show significant advantages over baselines, demonstrating the efficacy of our method.

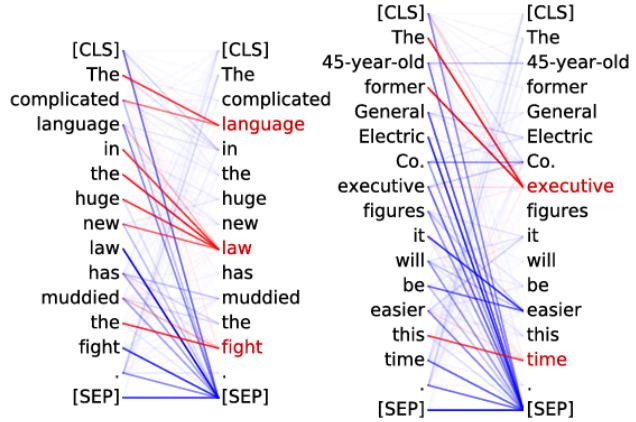


Figure 1. An example showing attention patterns in natural language correspond to human linguistic phenomena, where the line darkness represents the strength of attention weights. The attention pattern in this case is 94.3% consistent with the grammar rule “*The Noun modifiers (e.g., determiners) will attend to their noun*”, which is colored red in the Figure.

1. Introduction

Towards robust representations for various downstream tasks, masked autoencoder frameworks such as BERT are widely used for model pre-training in the field of natural language processing (NLP) [7]. More specifically, they mask parts of the input sequence through Transformer-based encoders to generate latent representations from the rest, and then leverage decoders to predict the masked content. This self-supervised pre-training approach leverages the multi-head self-attention (MSA) mechanism of the Transformer-based network architecture [25]. As a landmark technique in feature modeling, MSA is now ubiquitous also in the field of computer vision. The most widely accepted explanations for the success of MSAs are from the perspective of their weak inductive bias and capture of long-

term dependencies [4, 8, 19, 21]. From a convolutional neural network perspective, MSA is a transformation of all feature map points, with large size and data-specific kernels. Due to the simplicity and effectiveness of feature modeling, MSA is considered to be more expressive than channel attention in convolutional layers [6].

Since the discrete basic units (i.e. words or tokens) abstracted by humans in language are highly semantic and information-dense, encoders can learn bidirectional token-level interrelationships through sophisticated language understanding (e.g., grammar) and generate information containing compact semantics. Recent studies also showed that pre-trained language models have captured substantial linguistic [5, 11] and relational knowledge [14, 22, 24] through pre-training on large-scale text corpora. These methods typically design “fill-in-the-blank” questions based on predefined relationships. For example, a manually created question “Bob Dylan was born in _” for the language model is to answer the “birthplace” relation of “Bob Dylan”. Furthermore, the highly semantic information of language provides favorable conditions for many works to perform interpretable analysis on the MSA mechanism of Transformer, which is directly understandable by humans, i.e., the workflow of attention heads fully conforms with the syntactic and language concepts of coreference defined by human language grammars [5, 15]. Fig. 1 shows an example to illustrate that attention patterns in NLP correspond to linguistic phenomena (the grammar rule of “*The Noun modifiers will attend to their noun*”).

Inheriting the *mask-and-reconstruct* principle from NLP, He et al. first extend the masked autoencoding approach to the pre-training of Vision Transformer (ViT) model, which has gained great success on both model pre-training and inference [10]. Many subsequent studies focus on altering designs directly on the basic structure from NLP to achieve good performance [27, 29, 32], while ignoring the fundamental difference in how the MSA mechanism works between image and language settings. Image data, in contrast, is a natural signal whose basic units (i.e., pixels) contain only low-level representations and often exhibit severe spatial redundancy due to the image continuity in the neighbourhood, which poses a significant challenge to the interpretable analysis of MSA mechanisms in ViT. As yet, only a few studies try to interpret how MSA mechanism works in ViT, but their explanations are all based on empirical observations about attention distributions [18] or model output [1] on low-level space, while interpretable semantic analysis or theoretical support are absent from literature.

In this paper, to quantify the semantic information in low-level image space, we introduce a traditional image processing techniques, called Scale-Invariant Feature Transform (SIFT) [17]. More specifically, SIFT first maps the low-level redundant image space into mid-level fea-

ture space with highly semantic information. In the new space, it extract rich semantic information from image pixels to automatically annotates various discrete feature keypoints. These keypoints are theoretically proved invariant in different situations (e.g., rotation, scaling, brightness, and orientation), which also naturally correspond to the principles of image understanding in human logic [20]. Based on semantic-rich SIFT keypoints, we establish a weighted patch interrelation measurement to capture hidden attention patterns of all heads. The dynamic mathematical statistics from massive results help us to capture the existence of attention bias (i.e., patches with high SIFT keypoint concentration will pay more attention to each other, while relatively ignore other low concentration patches). We further conduct comprehensive analysis yo derive a summary of three different stages, which answers “Yes” to the question: “*Is there a semantically-similar interrelation between patches with different SIFT keypoint concentration?*”.

More interestingly, we find our quantitative analysis is not only an effective complement to the interpretability of existing MSA mechanisms, but can also be used for a variety of applications, including 1) discovering spurious correlations and prompting input during model inference to improve performance, 2) guiding model pre-training to speed up convergence. The main contributions of this paper are summarized as follows:

- We introduce SIFT to map the low-level redundant image space into the mid-level semantic feature space. It can automatically annotate invariant feature keypoints, and provide theoretical guarantee for measuring the semantic information level of different patches.
- We design a weighted quantitative analysis based on SIFT keypoints, which interprets that the attention patterns hidden in MSA mainly exploits the semantically-similar interrelation between patches with different SIFT keypoint concentration during inference.
- We further derive a range of well-suited applications from the interpretable analysis, including 1) discovering spurious correlations and prompting input during model inference to improve performance, and 2) guiding model pre-training to accelerate convergence.

2. Key Elements Hidden in Image Patch

In this section, through the pre-experiments on image recovery, we reveal the fact that patches have significant differences on semantic information level with hidden interrelation. Then, we use SIFT to quantify the hidden semantic information level of different patches.

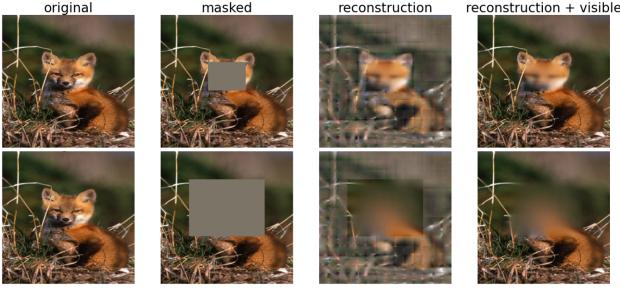


Figure 2. Image restoration results after adding different masks. We use the same pretrained ViT-B/16 model for image restoration, with original images from ILSVRC2012 (ImageNet 1K).

2.1. Patch Interrelation in Image Recovery

As shown in the second “*masked*” column of Fig. 2, we add masks with different sizes to the original image and restore the masked image using the same pretrained ViT-B/16 model, where original images are from ILSVRC2012 (ImageNet 1K). When we only mask the fox face regions in the top row, model can extract enough interrelation information from remaining patches to recover the masked face regions. However, if we extend the mask to the entire head area in the bottom row, model is unable to reconstruct the masked area from the remaining patches.

The above comparison results indicate the significant differences between patches and the absence of a few key patches will lead to model reconstruction capability degradation. According to previous studies [8, 10], the key elements hidden in patches are different semantic information they contain, i.e., key features. In addition, there is also a high degree of interrelation inference between patches containing key features, which together can provide sufficient semantic information to help the model understand parts, objects, and scenes in masked regions. For example, in the first row of Fig. 2, the patch with fox ears indicates that the masked area is likely to be a fox face. When patches with key features are masked (see bottom row of Fig. 2), the model cannot extract enough semantic information from the remaining patch interrelations to understand and recover the masked regions.

2.2. SIFT Keypoint

Before analyzing and quantifying the patch interrelation, we first need to identify the concentration of semantic information (i.e. key features) contained in different patches. As some mid-level representations with rich semantic information, key features should be invariant in different situations, such as rotation, scale, brightness, and orientation. Therefore, we introduce the classical scale-invariant feature transform (SIFT) to automatically annotate broad feature keypoints hidden in images [17].

More specifically, SIFT first uses difference of Gaussian

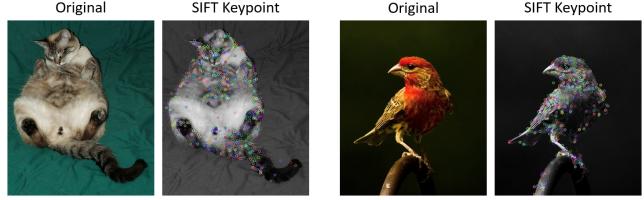


Figure 3. Illustration of SIFT keypoint annotations in an image. SIFT converts the original color image to a grayscale image, avoiding the effect of color on keypoint recognition.

(DoG) to generate a set of results obtained from the same image at different scales. Then, SIFT will calculate the directed gradient of each point and compare it with its neighbors to detect those local extreme points. Only those local extreme points that are invariant in different scale spaces are selected as keypoints. A demonstration of SIFT keypoint annotation is shown in Fig. 3, and we can observe that most SIFT keypoints are concentrated on the target ‘cat’, which is highly consistent with our definition of key features in human semantic understanding.

With the help of SIFT keypoints, we can quantify the concentration of semantic information (i.e. key features) contained in different patches by counting the number of SIFT keypoints in each patch. This quantitative metric will be used in the next section to reveal the patch interrelations hidden in the above image restoration pre-experiments.

3. Patch Interrelation Analysis

In this section, we first introduce the background of ViT and its multi-head self-attention (MSA) mechanism, and then describe our SIFT keypoint-based measure that can reveal the patch interrelations during model inference.

3.1. Preliminaries

For an original input image of resolution (H, W) , given that the resolution of each patch is (P, P) , ViT will split the image into a set of non-overlapping patches $[p_1, \dots, p_N]$, where the patch number is $N = HW/P^2$. By linear projection, these patches will be transformed into patch embeddings $[h_1, \dots, h_N]$ with fixed vector size. The patch embeddings are then fed into a series of stacked transformers, each containing a multi-head self-attention (MSA) mechanism with L parallel SA heads. More specifically, in an SA head, each patch embedding h_i will be mapped to the query vector q_i , key vector k_i and value vector v_i vector by a separate linear projection. The head will compute the attention weights α between all patch pairs by the softmax normalized dot product between the query and the key vector. The output o of the head is a weighted sum of all value vectors.

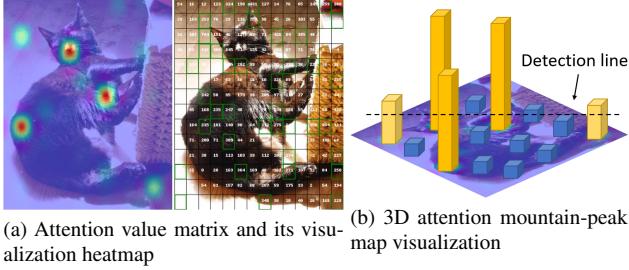


Figure 4. The attention matrix α_i of the patch i computed on an SA head and its visualization. (zoom in to see detailed values within the matrix)

$$\alpha_{i,j} = \frac{\exp(q_i^T k_j)}{\sum_{l=1}^n \exp(q_i^T, k_l)}, \quad o_i = \sum_{j=1}^n \alpha_{ij} v_j. \quad (1)$$

When generating the next representation of the current patch i , the attention weight α_{ij} can be regarded as the ‘importance’ of every other patch j to it, which is implicitly consistent with our patch interrelation. Therefore, our measurement method is based on the attention weight distribution of each MSA transformer block.

We use the **ViT-Large** model (ViT-L/16) pre-trained on ImageNet-21K as our backbone, which has 24 stacked transformer layers, each containing 16 attention heads.

3.2. Measurement of Patch Interrelation

Traditional attention mechanisms follow human logic on image processing to improve model performance by focusing attention on some key regions (such as objects in images) [12, 28]. They usually use spatial or channel attention to individually highlight some key features, while ignoring the semantic understanding hidden in patch interrelations is also important during model inference. Unlike traditional attention mechanisms, the computation process of the SA mechanism naturally involves comparisons between all possible patch pair combinations.

For example, when generating the next representation for patch i in equation (1), each SA head will compute a 16×16 attention matrix $\alpha_i = [\alpha_{ij}]$. The value of α_{ij} in each patch $j \in \{N\}$ represents the importance of other patches j to the patch i . In other words, α_{ij} also means the strength of patch interrelation between patches i and j . The attention heatmap visualized from this matrix is shown in Fig. 4a. The heatmap can also be transformed into a mountain-peak map by the 3D visualization in Fig. 4b, where the mountain height on each patch is α_{ij} .

To quantify patch interrelationships between patch i and another patch j , as shown in Fig. 4b, we design a detection line and define its height according to:

$$\bar{H} = \gamma * \frac{\sum_{j=1}^N \alpha_{ij}}{N}, \quad (2)$$

Where γ is an adjustment parameter that controls the height of the detection line. Only patches j with value α_{ij} higher than \bar{H} ($\alpha_{ij} \geq \bar{H}$) will be filtered out as attended patches (see yellow bars in Fig. 4b), the set of attended patches is denoted by \mathcal{S}_i .

Weighted Patch Interrelation Analysis. As we know from Sec. 2, patches are highly discriminative due to the set of semantic information they contain, i.e., the number of SIFT keypoints in the patch. The set of attended patches \mathcal{S}_i can thus be separated into two non-overlapping subsets: \mathcal{S}_i^{Non} includes attended patches that do not contain SIFT keypoints and \mathcal{S}_i^{Key} includes attended patches that contain SIFT keypoints that satisfy $\mathcal{S}_i = \mathcal{S}_i^{Non} \cup \mathcal{S}_i^{Key}$ and $\mathcal{S}_i^{Non} \cap \mathcal{S}_i^{Key} = \emptyset$. Intuitively, calculating the percentage of \mathcal{S}_i^{Key} in the entire attended patch set \mathcal{S}_i can reflect the patch interrelation distribution (i.e., starting with patch i , whether the model is biased towards those patches contain SIFT keypoints among all attended patches).

However, it is not fair to treat each patch equally, as the SIFT keypoints they contain may vary widely, and patches containing more SIFT keypoints should have higher weights in our patch interrelation analysis. Denote the number of SIFT keypoints in each patch $j \in \{N\}$ by t_j . The weighted patch correlation analysis generated for patch i on an SA head translates to:

$$\theta_i = \frac{|\mathcal{S}_i^{Key}|}{\underbrace{|\mathcal{S}_i^{Non}| + |\mathcal{S}_i^{Key}|}_{\text{Unweighted}}} \Rightarrow \theta_i = \underbrace{\frac{\sum_{j \in \mathcal{S}_i^{Key}} t_j}{|\mathcal{S}_i^{Non}| + \sum_{j \in \mathcal{S}_i^{Key}} t_j}}_{\text{Weighted}} \quad (3)$$

It is worth noting that each SA head generates a total N attention matrix α_i for different starting patch $i \in \{N\}$, and the starting patch i can vary significantly due to the contained SIFT keypoints. They are divided into two non-overlapping sets \mathcal{S}_i^{Non} and \mathcal{S}_i^{Key} , indicating whether the starting patch i contains SIFT keypoints. By the different identities (‘Non-keypoint’ patch or ‘Keypoint’ patch) that patches belong to, there are four different combinations between the starting patch i and the target patch j as: • KK: ‘Keypoint’ → ‘Keypoint’, • KN: ‘Keypoint’ → ‘Non-keypoint’, • NK: ‘Non-keypoint’ → ‘Keypoint’, and • NN: ‘Non-keypoint’ → ‘Non-keypoint’.

Take the • KK: ‘Keypoint’ → ‘Keypoint’ as an example, the target of patch interrelation analysis under this combination is to find ‘Starting from a keypoint patch, whether its attention distribution will present some special interrelation patterns towards other keypoint patches?’ The global patch

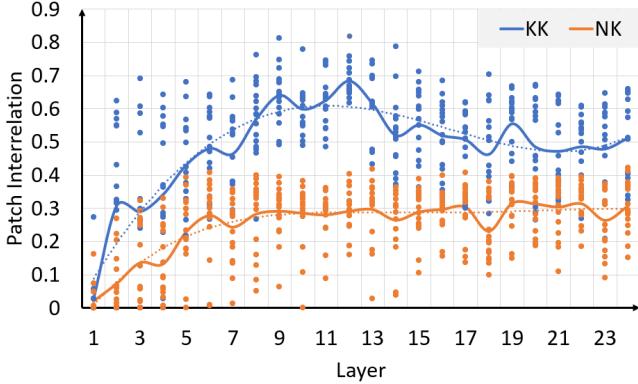


Figure 5. The global patch interrelation analysis on all 24 transformer layers, where each point corresponds to the patch interrelation analysis θ_i on a particular SA head. The solid line is the average of all 16 SA heads, and the dotted line is its trend line.

interrelation analysis under different combinations are:

$$\begin{aligned} \theta_{KK} &= \frac{\sum_{i \in \mathcal{S}^{Key}} \theta_i}{|\mathcal{S}^{Key}|}; & \theta_{NK} &= \frac{\sum_{i \in \mathcal{S}^{Non}} \theta_i}{|\mathcal{S}^{Non}|}; \\ \theta_{KN} &= \frac{\sum_{i \in \mathcal{S}^{Key}} (1 - \theta_i)}{|\mathcal{S}^{Key}|}; & \theta_{NN} &= \frac{\sum_{i \in \mathcal{S}^{Non}} (1 - \theta_i)}{|\mathcal{S}^{Non}|}. \end{aligned} \quad (4)$$

4. Hidden Patterns in Patch Interrelation

The global patch interrelation analysis on the entire image provides a comprehensive and novel understanding about how the ViT model utilizes patch interrelation in different stages during the model forward inference process.

4.1. Attention Patterns among Patches

We first perform the global patch interrelation analysis on all 16 SA heads throughout all 24 transformer layers, where the results are demonstrated in Fig.5. The observation is interesting: starting from keypoint patches $\in \mathcal{S}^{Key}$, their attention distributions are discriminative to different target patches, which are highly biased towards other keypoint patches $\in \mathcal{S}^{Key}$. We can see from the ‘KK’ case (blue line) in Fig.5, the interrelation between keypoint patches has a rapid increasing during the model forward inference process, and presents an obvious gap with the ‘NK’ case (orange line). This discriminative bias of ViT model prove our conjectures from pre-experiment results on image recovery, which indicates that: 1) There are high semantic interrelation patterns among keypoints patches. 2) The MSA mechanism of ViT model mainly utilize the keypoint patch interrelation during its forward inference process. The analysis results on ‘KN’ and ‘NN’ cases are omitted here but attached later in Fig.7a for different stages summary, since they have a clear mathematical $1 - \theta$ complementary re-

lationship with their respective counterparts (for example, ‘KN’ = $1 - \text{‘KK’}$), as shown in Eq.(4).

Moreover, except for the obvious gap between two cases, we can also see a dynamic change of the ‘KK’ case (blue line) in different stages of model forward inference process. This phenomenon indicates that the ViT model has a different semantic understanding of patch interrelation in different stages, which will be elaborated later in Sec.4.3.

4.2. Focused or Broad? — Attention Focus Index

Except for distribution characteristics reflected by above attention patterns, we also measure the attention focus level of ViT model, i.e., whether SA heads will focus on a few patches or attend broadly on many different patches. We use the average information entropy of each head’s attention distribution as our metric, which is called attention focus index. For a 16×16 attention matrix $\alpha_i = [\alpha_{ij}]$ of patch i generated by one SA head, we first transfer all attention value to the $[0, 1]$ interval by Min-Max normalization, and then calculate its information entropy δ_i as:

$$\delta_i = - \sum_{j=1}^{16 \times 16} \log p \left(\frac{\alpha_{ij} - \min(\alpha_i)}{\max(\alpha_i) - \min(\alpha_i)} \right). \quad (5)$$

Finally, by averaging all information entropy δ_i of total 256 different starting patch i on the same SA head, we can obtain the attention focus index of that specific SA head. We show the dynamic change of attention focus index on all SA heads during the model forward inference process in Fig.6.

Intuitively, a larger attention focus index indicates that the model only attend to fewer patches, vice versa. Therefore, we can find the attention focus index has an obvious change that the model rapidly focuses to only a few patches in former layers, but back to broad attention again in later layers. However, the implied focus patterns are highly diverse in different stages and we will provide a full analysis later in Sec.4.3 to explain them.

4.3. Retrieval, Capture, and Coach Stages in Vision Transformer

Combining all analysis results in previous sections, we can obviously see three different stages during the model forward inference process, which are • **Stage 1:** Global information *Retrieval*, • **Stage 2:** Key information *Capture* and • **Stage 3:** Semantic Interrelation *Coach*, as shown in Fig.7a with different background colors. Next, with the visualized attention heatmaps in Fig.7b, we will provide a detailed analysis to explain the differences of three stages and how the patch interrelation is used in diverse ways.

First, in • **Stage 1:** Global information *Retrieval*, the ‘KK’ and ‘KN’ patch interrelation are both very low (only have a small gap), which means that no matter what identity the starting patch is, the ViT model treats all target patches

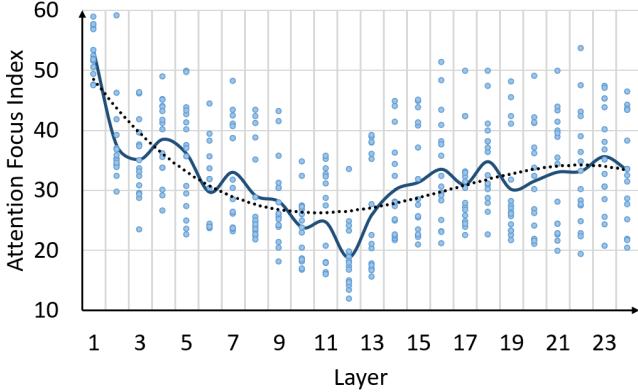
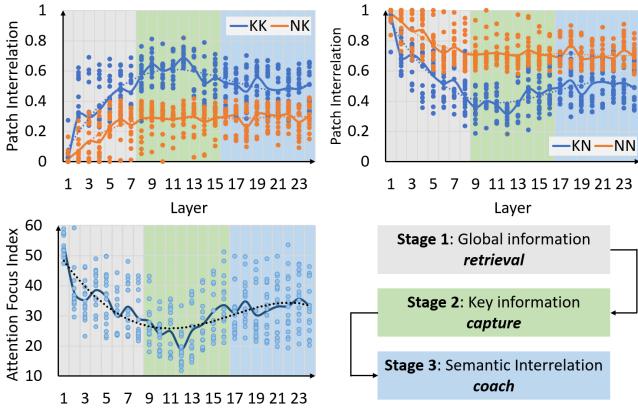
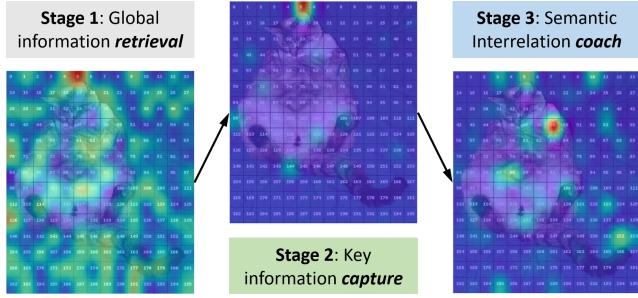


Figure 6. Attention focus index of all 16 SA heads (blue points) in each layer. The bold line shows their average in each layer and the dash line is its trend-line.



(a) Three different stages identified by our quantitative analysis in model forward inference process (indicated by different background colors).



(b) Visualized attention distribution heatmaps in three different stages.

Figure 7. Three different stages (i.e., Retrieval, Capture, and Coach Stages) in model forward inference process and their visualized heatmap samples of attention distributions.

equally. Besides, the high attention focus index at this stage also indicates that the attention of ViT model is broadly distributed over all patches, as shown in the heatmap sample of Fig. 7b. Therefore, the function of ViT model at this stage is mainly to retrieve all patches to integrate global information of the image, and thereby make preparations for the next stage.

Next, in • **Stage 2:** Key information *Capture*, the ‘KK’ patch interrelation is rapidly increasing and opening a huge gap with the ‘KN’, which means that the ViT model is biased in this stage. With the help of integrated global information in last stage, it concentrates on capturing the strong interrelation between those keypoint patches now. Meanwhile, the rapidly dropping attention focus index also suggests that the attention of ViT model is highly focused, as shown in the second sample of Fig. 7b. Thus, the ViT model in this stage is to capture the key patch interrelation and refine the critical information that can help to truly understand the image in later stage.

Finally, in • **Stage 3:** Semantic Interrelation *Coach*, we can observe a slight decrease in the ‘KK’ patch interrelation, along with a slight increase in attention focus index. These observations means that the ViT model reallocates a small portion of its attention back to other Non-keypoint patches, and thus results in the rebound of attention focus index. However, different from the global information retrieval in first stage, this attention reallocation is deliberately operated by the ViT model, whose target is to establish the semantic interrelation coachings among patches with different identities. Although those Non-keypoint patches contain only little information, but they aren’t completely useless for image understanding, since the sharp contrast between them and keypoint patches can further help the model to distinguish the true objective in the image. The third sample of Fig. 7b also further reflects our conclusion at this stage.

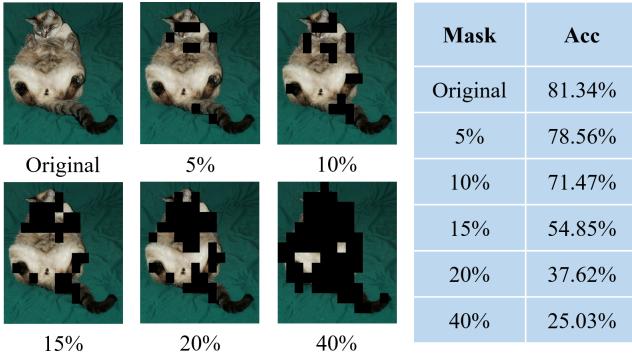
5. Applications in Practice

The previous patch interrelation analysis not only uncovers the hidden working mechanism of MSA in ViT model, but also provide further guidance to both model training and inference process. In this section, we list some of interesting applications in practice that can utilize the advantages of patch interrelation.

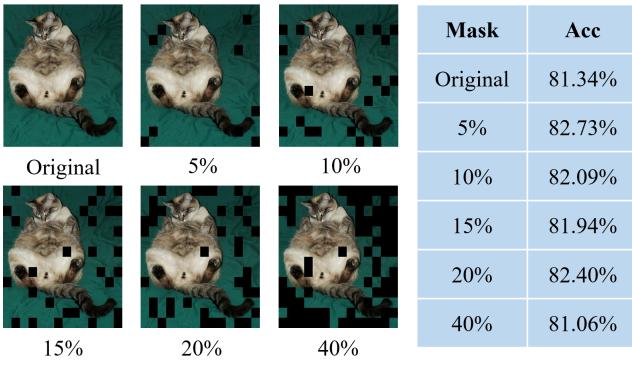
5.1. Spurious Correlation Discovery

Some previous studies [9, 23] have shown that traditional image classification tasks suffer from model performance degradation due to the spurious features hidden in images. For example, blue sky background is likely to co-occur in many bird images. However, the absence of blue sky background from a bird image should not result in misclassification. Different from their single-point spurious feature discovery, our patch interrelation can analyze the mutual attention distribution of different patch combinations, and then identify the true dominant feature correlation. For example, a keypoint patch, that assign most of its attention to those Non-keypoint patch, will be regarded as spurious correlation, even if this patch contains many keypoints.

To verify the effectiveness of our spurious correlation discovery, we design a special patch mask mechanism dur-



(a) Visualization of **Top** patch masking under different mask ratios and the change of model accuracy



(b) Visualization of **Bottom** patch masking under different mask ratios and the change of model accuracy.

Figure 8. Spurious Correlation Discovery. The **Top** case shows the importance of keypoint patch interrelation for image understanding. The **Bottom** case shows we can find the useless or even negative patch interrelation and serve as a *Prompt* to further improve model inference performance. The accuracy is tested on ILSVRC2012 (ImageNet 1K) with pretrained ViT-B/16 model.

ing the model inference process to understand what is the impact of spurious Correlation. According to Sec.3.2, we know that θ_i shows the attention level to other keypoint patches starting from patch i on a specific SA head. Since we have total $m = 16$ SA heads in each layer of our ViT model, we can get the average by $\theta_i = \frac{1}{16} \sum_{m=1}^{16} \theta_i$, which indicates the average attention level of patch i to other keypoint patches. In other words, each patch i will have a score θ_i that reflects its importance level of patch interrelation. Next, we use θ_i to implement our patch mask mechanism from two different aspects: **Top** and **Bottom**. In the Top patch mask case, we will mask a subset of patches based on their score θ_i from highest to lowest, where r is the mask ratio (vice versa for the bottom case). We set different mask ratio from $r = 5\% \rightarrow 40\%$, and visualize all mask results as shown in Fig.8.

Intuitively, in the **Top** case, all masked patches are the most important keypoint patches that contain the critical semantic information for understanding the image. Therefore, we can clearly observe a rapid image classification accuracy

drop as the increasing of mask ratio in the table of Fig.8a. Besides, when the mask ratio r is relatively small (such as 5% and 10%), the ViT model can still use the patch interrelation from the remaining to understand the missing content laterally, which results in a slow performance degradation. Once the mask ratio r is too large that the remaining patches cannot provide enough semantic information for understanding, the model accuracy will quickly break down, and the crash point is between 10% and 20%.

Mask as Prompt. In contrast to the straightforward results in previous Top case, we have some more interesting finding in the bottom case, which can further improve the model inference performance. Since patches in bottom case are masked according to the lowest ordering of θ_i value, which means that they can provide very little semantic information and patch interrelation to understand the image. What's worse, there is usually some redundant background in the image that has no relevance to the real objective, they may even induce negative information to hinder the image understanding. Removing these negative information from the original image will potentially drive the model to focus more on the true objective and learn the real patch interrelation hidden in the image. This concept is borrowed from *Prompt Learning* [13, 33], which only introduce small number of extra trainable parameters (called *Prompt*) in the original inputs to adapt different downstream tasks, thereby improving the model inference performance. Follow the similar insights, the patch mask mechanism is served as the *Prompt* in our case.

From the experiment results in the table of Fig.8b, We can see that the model classification accuracy has a slight increasing when the mask ratio is relatively small (see 5% and 10% cases), which means these masked patches will provide negative information and interrelation to hinder image understanding before. This phenomenon is also consistent with our concept of "Mask as Prompt". Besides, even if the mask ratio r is later increased to 40%, there is no obvious accuracy drop compared to the original image, which empirically proves the effectiveness of our patch interrelation analysis to find the hidden spurious correlation in the image.

5.2. Guided Mask for Model Pre-training

Except the previous applications for model inference, Our patch interrelation analysis can also be utilized in the model pre-training process to speed up the training efficiency. The existing model pre-training of MAE encoder is based on a random mask mechanism [10], which only randomly samples a patch subset to pre-train ViT model. When facing the agnostic patch interrelation, random mask is a straightforward approach to effectively ensure the model pre-training performance. As long as the random sampling time (training time) is large enough to cover differ-

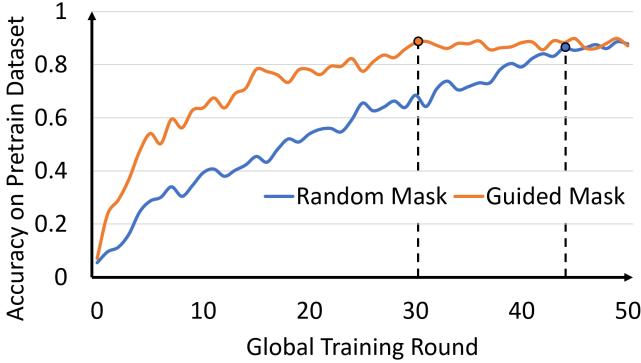


Figure 9. The ViT-Base/16 model pre-training on ImageNette with different mask mechanisms, where our guided mask can accelerate pre-training process to reach convergence with less rounds.

ent patches combinations, random mask can guarantee ViT model to understand the hidden semantic information in images (i.e., various patch interrelation in our paper).

However, according to the studies in Curriculum Learning (CL) [2], the model training process is similar with human learning process, it is advocated to let the model training start from easy data samples first and gradually progress to complex data samples and knowledge. Through a series of well-designed learning tasks, the CL strategy can accelerate the model training process to reach the same model performance with less training iterations. Therefore, based on the insights from CL, we propose to gradually increase the dataset difficulty during ViT model pre-training process, thus to guide and accelerate the model learning. According to previous θ_i derived from our patch interrelation analysis, each patch i will have a score indicates its importance of semantic understanding to other keypoint patches, and we can use it to design our datasets with different difficulty.

Assume the mask ratio r is a constant, such as $r = 50\%$. For random mask, it just needs to randomly mask 50% patches and feed the remaining patches into model for pre-training, where the percentage of Keypoint patches and Non-keypoint patches in masked patches are also random. If the percentage of Keypoint patches in all masked patches is denoted by β , we can now adjust the dataset difficulty by setting different β value. First, we calculate the masked Keypoint/Non-keypoint patch number by: "total patch number $\times r \times \beta / (1 - \beta)$ ", respectively. Then, we shuffle the two lists of all Keypoint/Non-keypoint patches and sample the corresponding number of patches from the top of Keypoint patch list and the bottom of Non-keypoint patch list, where these sampled patches are masked. Obviously, a larger β value means more Keypoint patches are masked, which indicates higher dataset difficulty. When $\beta = 0\%$, all masked patches are Non-keypoint patches (lowest dataset difficulty), vice versa for $\beta = 100\%$.

In our experiment, we design 5 datasets with increas-

ing difficulty from $\beta = 10\%$ to $\beta = 50\%$, and gradually increase the dataset difficulty every 10 rounds. The experiment configurations are listed as below: model (ViT-Base/16), Pre-train dataset (ImageNette¹), GPU (one RTX 3090). We use the classification accuracy on the pretrain dataset to evaluate the model pre-training performance. It is clear from Fig.9 that our guided mask can reach convergence point at around 30-th training round, while the random mask needs to take about 45 training rounds.

6. Related Work

6.1. Transformer-based models for vision tasks

Different from the traditional CNN-based models that use convolution kernel to extract features from the original image for later tasks, the Transformer-based model architecture mainly utilize the self-attention mechanism to capture the long-term dependencies between different parts of the image. Vision Transformer (ViT) model is the first work that applies MSA mechanism on image, by splitting it into a series of non-overlapping patches as the input of Transformer model [8]. The flexibility and effectiveness of ViT make it becomes the ubiquitous landmark technique in computer vision field. Extensive subsequent studies based on ViT are proposed to design more efficient MSA mechanisms, including implementing deeper ViT model with diverse regenerated attention map [31], constructing hierarchical ViT structure to apply the shift window scheme and limiting the focus of self-attention mechanism [16], and multi-scale ViT structure with attention mechanism of Vision Long former [30]. However, despite the promising performance, all these studies have ignored a fundamental problem, i.e., *how the MSA mechanism works in vision?*

6.2. Interpretability of Vision Transformer

When ViT first demonstrate its powerful performance to outperform previous CNN-based baselines [8], its unique model structure has attracted extensive focus from researchers to understand its interpretability from different aspects, including: observing the attention map of Transformer outputs [3, 15], computing the relevancy of different attention heads in Transformer networks [5, 26]. However, their explanations all come from the empirical observations based on Transformer network outputs or layer similarity, without any interpretable analysis backed by a solid theoretical foundation. Different from them, we try to understand the interpretability of ViT from a semantic perspective similar to NLP, and prove that "*the working principle of MSA mechanism in ViT is consistent with human semantic understanding on images*".

¹<https://www.tensorflow.org/datasets/catalog/imagenette>

7. Discussion & Conclusion

In this paper, we find that the low semantic information concentration of pixels in images is a critical challenge that hinders the interpretability of ViT model. To fill this missing gap, we introduce the concept of SIFT to map the low-level redundant image space to a mid-level semantic feature space, which can automatically annotate the invariant feature keypoints in the image, and provide a theoretical cross-correlation for measuring the semantic information level. Then, we design a weighted quantitative analysis to explore the hidden attention patterns in the MSA mechanism, whose interpretability is consistent with the semantic understanding based on the SIFT keypoints. Finally, we further derive a series of well-suited practical applications that benefit a lot from our interpretable analysis, including: 1) discovering spurious correlations and prompting input during model inference to improve performance, and 2) guiding model pre-training to accelerate convergence.

References

- [1] Shir Amir, Yossi Gandelsman, Shai Bagon, and Tali Dekel. Deep vit features as dense visual descriptors. *arXiv preprint arXiv:2112.05814*, 2(3):4, 2021. [2](#)
- [2] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48, 2009. [8](#)
- [3] Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 782–791, 2021. [8](#)
- [4] Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Twins: Revisiting the design of spatial attention in vision transformers. *Advances in Neural Information Processing Systems*, 34:9355–9366, 2021. [2](#)
- [5] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. What does bert look at? an analysis of bert’s attention. *arXiv preprint arXiv:1906.04341*, 2019. [2, 8](#)
- [6] Jean-Baptiste Cordonnier, Andreas Loukas, and Martin Jaggi. On the relationship between self-attention and convolutional layers. In *Proceedings of International Conference on Learning Representations*, 2020. [2](#)
- [7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. [1](#)
- [8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. [2, 3, 8](#)
- [9] Sabri Eyuboglu, Maya Varma, Khaled Saab, Jean-Benoit Delbrouck, Christopher Lee-Messer, Jared Dunnmon, James Zou, and Christopher Ré. Domino: Discovering systematic errors with cross-modal embeddings. *arXiv preprint arXiv:2203.14960*, 2022. [6](#)
- [10] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. [2, 3, 7](#)
- [11] John Hewitt and Christopher D Manning. A structural probe for finding syntax in word representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4129–4138, 2019. [2](#)
- [12] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. [4](#)
- [13] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. *arXiv preprint arXiv:2203.12119*, 2022. [7](#)
- [14] Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438, 2020. [2](#)
- [15] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. What does bert with vision look at? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5265–5275, 2020. [2, 8](#)
- [16] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. [8](#)
- [17] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. [2, 3](#)
- [18] Jie Ma, Yalong Bai, Bineng Zhong, Wei Zhang, Ting Yao, and Tao Mei. Visualizing and understanding patch interactions in vision transformer. *arXiv preprint arXiv:2203.05922*, 2022. [2](#)
- [19] Xiaofeng Mao, Gege Qi, Yuefeng Chen, Xiaodan Li, Ranjie Duan, Shaokai Ye, Yuan He, and Hui Xue. Towards robust vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12042–12051, 2022. [2](#)
- [20] Jean-Michel Morel and Guoshen Yu. Is sift scale invariant? *Inverse Problems & Imaging*, 5(1):115, 2011. [2](#)
- [21] Muhammad Muzammal Naseer, Kanchana Ranasinghe, Salman H Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Intriguing properties of vision transformers. *Advances in Neural Information Processing Systems*, 34:23296–23308, 2021. [2](#)
- [22] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Proceedings of*

- the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, 2019. 2
- [23] Sahil Singla, Besmira Nushi, Shital Shah, Ece Kamar, and Eric Horvitz. Understanding failures of deep networks via robust feature extraction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12853–12862, 2021. 6
- [24] Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. olmpics-on what language model pre-training captures. *Transactions of the Association for Computational Linguistics*, 8:743–758, 2020. 2
- [25] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 1
- [26] Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. *arXiv preprint arXiv:1905.09418*, 2019. 8
- [27] Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14668–14678, 2022. 2
- [28] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018. 4
- [29] Li Yuan, Qibin Hou, Zihang Jiang, Jiashi Feng, and Shuicheng Yan. Volo: Vision outlooker for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 2
- [30] Pengchuan Zhang, Xiyang Dai, Jianwei Yang, Bin Xiao, Lu Yuan, Lei Zhang, and Jianfeng Gao. Multi-scale vision longformer: A new vision transformer for high-resolution image encoding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2998–3008, 2021. 8
- [31] Daquan Zhou, Bingyi Kang, Xiaojie Jin, Linjie Yang, Xiaochen Lian, Zihang Jiang, Qibin Hou, and Jiashi Feng. Deepvit: Towards deeper vision transformer. *arXiv preprint arXiv:2103.11886*, 2021. 8
- [32] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021. 2
- [33] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022. 7