Can't See the Forest for the Trees: Diagnosing the Spatial Blindness in CNN-based Models via Grad-CAM

Yixuan Zheng
School of EECS, Peking University
2400011003@stu.pku.edu.cn

Abstract

Convolutional Neutral Network(CNN) has long been considered a powerful tool in Computer Vision, and is applied in many downstream tasks such as object classification, image caption and visual question answering(VQA). However, there ability to capture global spatial logic remains poorly understood. To open the black box, we conduct a multi-stage investigation into CNN's spatial perception ability via some visual explanation techniques. We first implemented Class Activation Mapping(CAM) [10] and Gradient-weighted Class Activation Mapping (Grad-CAM) [9] as our main tools to analyze models' behavior. Also, we introduced a deletion metric to quantitatively evaluate the heatmap generated by CAM and Grad-CAM. Through a series of diagnostic tests, we found that CNN-based architectures are heavily biased towards local texture cues over global spatial structure. Finally, inspired by these findings, we explore a novel adversarial attack strategy guided by Grad-CAM heatmaps. Our experiments show that perturbing pixels deemed least important by the model is surprisingly more effective than attacking the most salient regions, further suggesting the model's reliance on non-intuitive, imperceptible features rather than overall structure. These findings collectively highlight fundamental limitations in the spatial and structural reasoning of current CNN-based models and suggest critical directions for future research.

1 Introduction

In biological vision systems, spatial perception is a fundamental and critical capability. From a predator locating its prey to human's recognition, an accurate understanding of the spatial relationships between and in objects—such as their position, orientation, and one's own topological structure—is a cornerstone of survival and interaction. However, when we examine some CNN-based models, we find this foundational ability to be strikingly absent. (See Fig. 10)

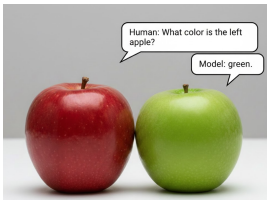


Figure 1: Spatial blindness of model (QA with MDETR-efficientnetB5)

This project aims to systematically diagnose, visualize, and investigate the underlying mechanisms of this "Spatial Blindness". Our research is structured into three stages:.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

In **stage 1**, We begin by implementing two visual explanation techniques, CAM and Grad-CAM. To move beyond qualitative result, we implement a quantitative Deletion Metric.

In **stage 2**, we designed three experiments to uncover the spatial blindness of CNN-based models. They include patch shuffling, image-text corresponding, and visual question answering. In patch shuffling test, we divide the image into 4×4 grids and randomly shuffle them, then measure the change in the logit score for the ground-truth class. We found a surprising result that most images get higher logit after shuffling. In image-text corresponding test, we use a pretrained CLIP-ResNet50 [8] which gives an alignment logit to a description about the input image. When the model assigned a higher logit to a spatially incorrect description, we investigated the failure by applying Grad-CAM. We use a pretrained MDETR-EfficientNetB5 [4] for the VQA task, and found the failure pattern is similar.

In **stage 3**, we explored another interesting field—adversarial attack, a technique used to deceive models by feeding them specially crafted, often subtly altered, input data to cause incorrect predictions or unintended actions. In our human's prior knowledge, an object's most salient features, such as a dog's nose or ears, are the most critical for its classification. A logical hypothesis, therefore, is that an attack would be more effective when focused on these regions. Our experiments with ResNet-50, however, reveal the opposite to be true.

2 Related work

2.1 Visual explanation methods

CAM leverages the Global Average Pooling (GAP) layer to generate coarse, class-specific heatmaps by taking a weighted average of the final convolutional feature maps. However, its requirement for a specific network architecture limited its general applicability. This limitation was overcome by Grad-CAM. Grad-CAM generalizes the CAM approach by using the gradients of the target class score with respect to the feature maps of any convolutional layer. This makes it applicable to a wide variety of models and tasks. Our work adopts Grad-CAM as the primary diagnostic tool to generate the visual attention maps. To adjust Grad-CAM in some CNN-based models, we made some small changes in basic Grad-CAM implementation.

2.2 Faithfulness metric for explanation

In the work of Petsiuk et al., [7] they implemented an deletion metric which measures the Area Under the Curve (AUC) of the model's output probability for the target class as pixels are progressively removed in order of their saliency. However, probability can be influenced by other class, a direct consequence of the Softmax function. We propose a new method evaluates the decline in the **target class's normalized logit score**. Since logits provide a more direct and isolated measure of the evidence for a single class, we argue that our method provides a more precise measure of a pixel's influence.

Besides, we observe that in the later stages of the deletion process, the model output becomes weakly correlated with the heatmap ordering. Therefore, we focus our analysis on the early phase of deletion and report the metric over the first 50% of removed pixels.

2.3 CNN model bias

Despite of the great success in CNN, Geirhos et al. found that CNN trained on large-scale datasets like ImageNet exhibit a strong bias towards recognizing local textures rather than holistic shapes. [1] Our patch shuffling experiment is directly inspired by this research and serves as another evidence that CNN is not recognizing shape but salient feature. It demonstrates that CNN fails to perceive the global spatial structure.

2.4 Adversarial attack

The Fast Gradient Sign Method (FGSM) [2] constitutes the foundational approach for first-order adversarial attacks, perturbing an input x in the direction that maximally increases the loss. Subsequent methods, such as the Iterative FGSM (I-FGSM) [5] and Projected Gradient Descent (PGD) [6], extend

this idea by applying FGSM multiple times while constraining the perturbation within a ϵ-ball. Our attack method builds upon these methods by introducing a spatial mask that constrains the perturbation in specific regions. More details will be discussed in Section 4

3 Data

CIFAR-10 This dataset consists of 60000 32×32 colour images in 10 classes and is used for the initial implementation and quantitative evaluation of CAM and Grad-CAM in Stage 1.

Oxford-IIIT Pet Dataset This is a 37 category pet dataset with roughly 200 images for each class. We use the test split for patch shuffling and adversarial attack experiments.

Custom Images We created some images featuring objects with controlled spatial relations by artificial intelligence. They are used to test CNN-based models about their spatial perceiving ability.

All images are resized, cropped, and normalized following the preprocessing method of the target model.

4 Methods

4.1 Our deletion metric

Unlike standard approaches that track probability, our metric measures the Area Under the Curve (AUC). The y-axis is the normalized logit score for the target class and the x-axis is the percentage of pixels that are blurred by gaussian kernel. The logit score is normalized between its value on the original image $L_{initial}$ and its value on the fully blurred image L_{final} as:

$$\tilde{L} = \frac{L - L_{final}}{L_{initial} - L_{final}}$$

When the removed pixels are important for the target class, the curve is expected to drop rapidly and lead to a small AUC. The advantage of our metric is that it exclude other classes' influence as we have discussed in Section 2.2.

4.2 Our modifications on basic Grad-CAM

Grad-CAM is initially used in CNN. To apply it to models that are not purely CNN but incorporate CNN backbones, we make minor adaptations to the original Grad-CAM implementation. In CLIP-ResNet50, the model outputs a similarity logit for each image–text pair. We backpropagate the logit associated with the target text prompt through the CNN backbone to compute heatmap. For MDETR-EfficientNetB5-GQA, we select the score corresponding to the predicted answer and backpropagate it to the backbone feature maps.

4.3 Our adversarial attack method

Unlike standard first-order adversarial attacks such as FGSM, I-FGSM, and PGD, which are typically formulated as untargeted attacks that maximize the cross-entropy loss of the original (ground-truth) class, our method considers a targeted attack setting. Specifically, given a target class y, we try to decrease the cross-entropy loss with respect to the target y, thereby fooling the model to classify the image as class y.

Furthermore, we introduce a spatial mask $M \in \{0,1\}^{H \times W}$ to restrict perturbations within a subset of pixels. The mask can be defined according to different criteria, such as the top 10% of pixels ranked by heatmap values. To stabilize the optimization and avoid overly aggressive updates in later iterations, we adopt a cosine annealing schedule for the attack step size. Specifically, the step size is initialized to α_0 and gradually decayed to a minimum value of α_{min} following a cosine schedule over the attack iterations. Details can be found in Appendix B

5 Experiments

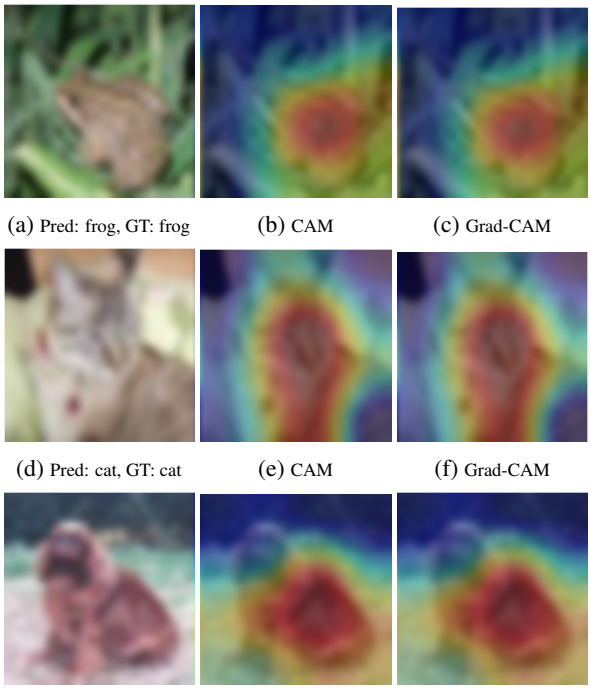
5.1 Basic CAM and Grad-CAM implementation

We choose ResNet implemented in Homework 6 as our model architecture. (See Fig. 7) and CIFAR-10 as the dataset. Our ResNet achieved an accuracy of 86.94% on the test set. Representative results are shown in Fig. 2

From the visualizations, both CAM and Grad-CAM highlight image regions that are consistent with human visual attention, indicating that the model bases its classification decisions on semantically meaningful object parts.

In Fig. (g), where the model incorrectly predicts a dog as a cat, the attention is primarily concentrated on the texture and fur patterns of the animal's body, while relatively little emphasis is placed on the head region. This bias toward body-level texture cues, rather than more discriminative facial features, likely contributes to the misclassification.

Since the final classification layer of our network is a single linear layer, the gradients with respect to the class score correspond directly to the learned class-specific weights. Under this setting, CAM and Grad-CAM are expected to produce highly similar activation maps, which is consistent with our empirical observations.



(a) Pred: frog, GT: frog (b) CAM (c) Grad-CAM

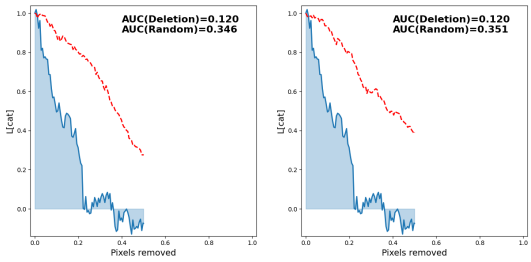
(d) Pred: cat, GT: cat (e) CAM (f) Grad-CAM

(g) Pred: cat, GT: dog (h) CAM (i) Grad-CAM

Figure 2: Representative CAM and Grad-CAM visualizations for correct and incorrect predictions.

5.2 Faithfulness

In this experiment, we implement the deletion metric mentioned in Section 4.1. Specifically, we first fix the total number of deletion steps T. In each step, $1/T$ pixels is blurred, following the descending order of heatmap values, until 50% of the pixels are processed. We record the logit after each step and plot a curve to calculate AUC. To assess the reliability of this metric, we introduce a random baseline for comparison. In the random setting, the same fraction of pixels is blurred at each step, but the pixel locations are randomly selected. This process is repeated five times and the results are averaged to obtain a stable curve.



(a) CAM (b) Grad-CAM

Figure 3: Deletion metric result for Fig. 2 (d)

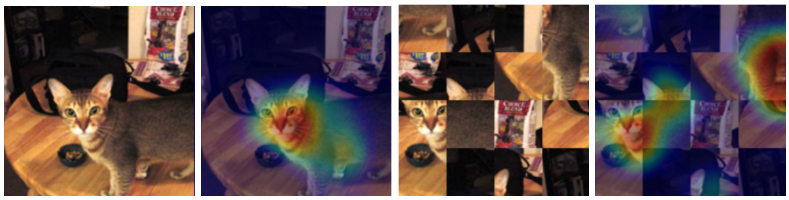
The blurring operation is implemented using a Gaussian kernel with kernal size $(11, 11)$ and standard deviation $\sigma = (15, 15)$. We test CAM, Grad-CAM and Random on the first 100 images, the AUC results are **0.190, 0.191, 0.314** respectively.

As shown in Fig. 3, deletion guided by CAM and Grad-CAM leads to a substantially faster drop in the target logit compared to the random baseline. This indicates that pixels with high attribution scores indeed play a more critical role in the model's prediction, confirming the faithfulness of the heatmaps. Also, because the similarity we have discussed in Section 5.1, we cannot answer which method appears more faithful to the model's behavior.

5.3 Patch shuffle

Intuitively, shuffling local patches of an image destroys its global spatial structure and semantic coherence. We would therefore expect the model to show a noticeable drop in confidence under patch shuffling. We use ResNet-50 and Oxford-IIIT Pet dataset. Quantitatively, out of 3669 test samples, 2828 (77.1%) remain the same prediction after patch shuffling. Among these samples, 95.58% have an increase in logit score, with an average logit change of +2.24.

We analyze the result by Grad-CAM. The visualizations (Fig.4 provide strong evidence that CNN-based models do not concern about the global spatial structure of visual features. Instead, their predictions are largely driven by the presence of locally discriminative patterns, regardless of their spatial arrangement.



(a) Input (b) Grad-CAM (c) Shuffled (d) Grad-CAM

Figure 4: Patch shuffle test. Original logit: 5.5061, Shuffled logit: 8.2222

5.4 CLIP and MDETR experiments

We next extend our analysis to multimodal models with CNN backbones, including CLIP and MDETR, with some adaptations in Grad-CAM mentioned in Section 4.2. CLIP is a vision–language model trained to align images and text in a shared embedding space via contrastive learning. We use CLIP with a ResNet-50 backbone for experiment. We choose the last convolutional layer as the target layer to generate Grad-CAM heatmap.

We conduct a qualitative analysis on CLIP to examine whether it captures spatial alignment between visual regions and textual descriptions. Specifically, we input an image containing two apples: a red apple on the left and a green apple on the right. We then construct four captions: (a) the color of the left apple is **red**. (b) the color of the left apple is **green**. (c) the left apple. (d) the right apple. We visualize the model attention by Grad-CAM. The results are shown in Fig. 5, from left to right.

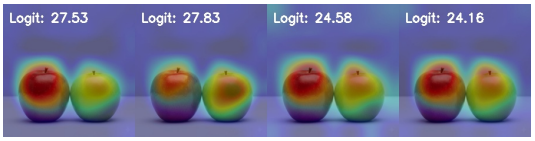


Figure 5: CLIP, from left to right shows caption (a), (b), (c), (d)'s visualization result

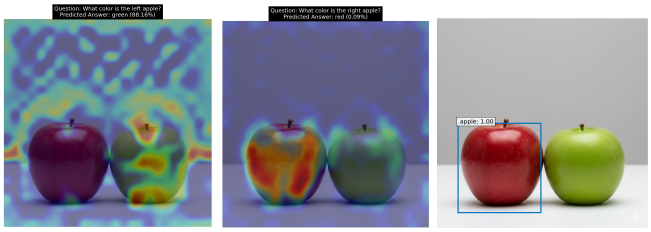
The heatmaps reveal that CLIP fails to localize objects based on spatial language. It simply associates textual descriptions with visual objects.

We use MDETR with an EfficientNet-B5 backbone on the VQA task. It performs multimodal reasoning by aligning textual queries with image regions via cross-attention. We choose the last convolutional layer as the target layer to generate Grad-CAM heatmap.

We use the same photo and query the model with the question "What color is the left apple?". The model gives the answer "green" with a 88.16% confidence and Grad-CAM highlight the right green apple. Interestingly, if we force the model to answer "red" and regenerate the heatmap, Grad-CAM highlight the left red apple. This serve as the evidence that the model can distinguish the red color, but is "blind" with the prominent green feature and fails to reliably obey the spatial constraint specified in the question.

Another interesting finding is that when prompted with the question "What color is the apple on the left?", MDETR is able to correctly localize the left apple by producing an accurate bounding box though gives the same "green" answer. This observation points to a potential direction for improving spatial reasoning in multimodal models: do the object detection and then explicitly constrain the answering process based on bounding box regions. Such combination may allow models to better leverage spatial structure rather than relying solely on global semantic cues.

More about CLIP and MDETR can be found in Appendix D, E.



(a) Answer: green (b) Answer: red (c) Bbox MDETR circled

Figure 6: MDETR visualization result

5.5 Adversarial attack

We implement the masked PGD attack described in Section 4.3. Specifically, we design six different masking strategies to control which pixels are allowed to be perturbed: (1) an all-one mask, corresponding to standard PGD; (2) a mask covering the top 10% pixels with the highest heatmap values of the original image; (3) a mask covering the bottom 10% pixels with the lowest heatmap values of the original image; (4) a mask covering the top 10% heatmap pixels of a target-class reference image, where reference image is selected for each class from ImageNet-1000[1]; (5) a mask covering the bottom 10% heatmap pixels of the target-class reference image; and (6) a uniformly spaced mask selecting 10% of pixels at regular intervals.

We test on the Oxford-IIIT Pet dataset's first 100 images. The target class index is 42(agama). The optimization terminates early if the model assigns a probability greater than 0.7 to the target class y or stops after a maximum of 50 iterations(which is considered as a failure). The result is shown in Table 1.

[1]https://github.com/EliSchwartz/imagenet-sample-images

Table 1: Adversarial Attack

Method	Success Rate	Average Steps
(1) Full-PGD	100%	8.5
(2) Origin Top-10%	52%	33.2
(3) Origin Bottom-10%	75%	28.1
(4) Target Top-10%	47%	36.0
(5) Target Bottom-10%	70%	26.2
(6) Uniform-10%	73%	21.1

Contrary to our initial assumption that perturbing the most salient regions would yield the most effective attacks, we observe that attacks targeting low-activation or uniformly masked regions can be more effective. This highlights a fundamental gap between **model interpretability** and **model vulnerability**.

As argued by Ilyas et al. [3], adversarial examples reveal that modern neural networks rely heavily on non-robust features—which are more **sensitive** to pixels change than robust features. Although such features may not produce strong activations in visualization methods like Grad-CAM, they can exert a disproportionate influence on the model's decisions. As a result, small perturbations applied to these regions are sufficient to significantly alter the model's predictions. Importantly, these non-robust features do not have coherent spatial structure, they are more like a chaotic noise learned from the dataset. This explains why a uniform masking strategy can outperform saliency-guided attacks. Our findings therefore caution against interpreting Grad-CAM heatmaps as a complete representation of the features that govern model decisions. Subtle features that are barely perceptible to humans and weakly activated in the heatmap may dominate the model's decision process.

6 Conclusion

In this project, we conducted a systematic investigation into the spatial reasoning and interpretability of CNN-based models through a series of diagnostic experiments. Using CAM and Grad-CAM as primary analysis tools, we examined both classification models and multimodal systems with CNN backbones, combining quantitative faithfulness metrics with qualitative visualization.

Our results reveal that, CNN-based models are heavily biased toward salient features rather than coherent global or spatial structures. This limitation persists across different tasks and architectures, including multimodal settings such as CLIP and MDETR. Moreover, we observe that there is a gap between Grad-CAM explanations and the features that actually govern model behavior.

References

[1] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *CoRR*, abs/1811.12231, 2018. URL http://arxiv.org/abs/1811.12231.

[2] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015. URL https://arxiv.org/abs/1412.6572.

[3] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features, 2019. URL https://arxiv.org/abs/1905.02175.

[4] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr – modulated detection for end-to-end multi-modal understanding, 2021. URL https://arxiv.org/abs/2104.12763.

[5] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world, 2017. URL https://arxiv.org/abs/1607.02533.

[6] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2019. URL https://arxiv.org/abs/1706.06083.

[7] Vitali Petsiuk, Abir Das, and Kate Saenko. RISE: randomized input sampling for explanation of black-box models. *CoRR*, abs/1806.07421, 2018. URL http://arxiv.org/abs/1806.07421.

[8] 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. Learning transferable visual models from natural language supervision, 2021. URL https://arxiv.org/abs/2103.00020.

[9] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, abs/1610.02391, 2016. URL http://arxiv.org/abs/1610.02391.

[10] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. *CoRR*, abs/1512.04150, 2015. URL http://arxiv.org/abs/1512.04150.

A Appendix: Supplementary material

Code is available in https://github.com/VVKAHPM/2025-Fall-Computer-Vision-Final-Project

B Appendix: Our adversarial attack algorithm

Algorithm 1: Targeted Masked Gradient-based Attack

Input: Input image x, target label y_t, model f, loss function $\mathcal{L}$, binary mask M, step size α, number of steps T, perturbation budget ϵ initial step size α_0, minimum step size $\alpha_{\min}$

Output: Adversarial example $x^{(T)}$

Initialize $x^{(0)} = x$;

for $t = 0$ **to** $T - 1$ **do**

Compute gradient $g^{(t)} = \nabla_x \mathcal{L}(f(x^{(t)}), y_t)$;

Compute step size using cosine schedule:

$$\alpha_t = \alpha_{\min} + \frac{1}{2}(\alpha_0 - \alpha_{\min}) \left(1 + \cos\left(\frac{\pi t}{T}\right)\right)$$

Update adversarial example:

$$x^{(t+1)} = x^{(t)} - \alpha_t \cdot M \odot \text{sign}(g^{(t)})$$

Project perturbation into ϵ-ball:

$$x^{(t+1)} = \text{clip}_{x,\epsilon}(x^{(t+1)})$$

end

return $x^{(T)}$

C Appendix: Our ResNet architecture

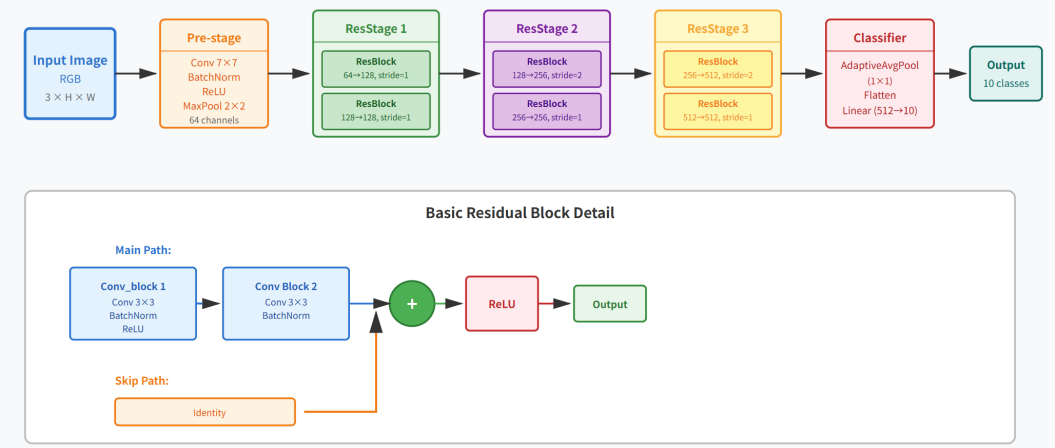


Figure 7: Custom ResNet architecture

D Appendix: CLIP

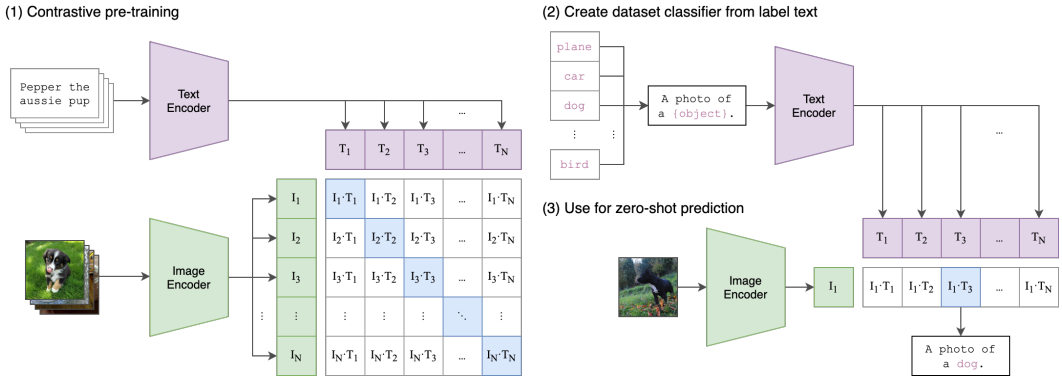


Figure 8: CLIP Pipeline from [8]

Figure 9: left: "the apple is on the left", right: "the banana is on the left"

E Appendix: MDETR

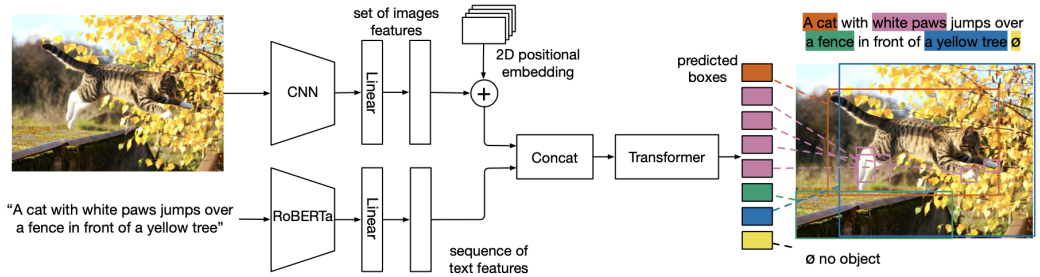
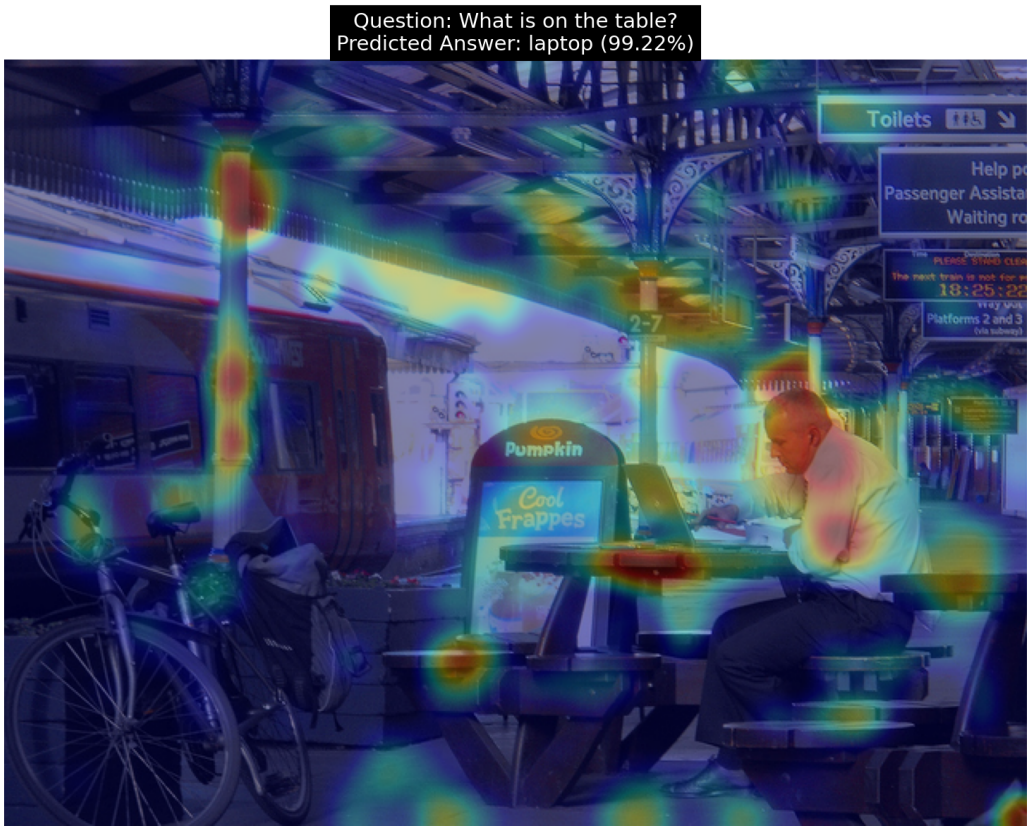


Figure 10: MDETR pipeline from [4]



(a) Grad-CAM

(b) Bounding box

Figure 11: MDETR visualization result with prompt "What is on the table?"