



**HAL**  
open science

# Masking Strategies for Background Bias Removal in Computer Vision Models

Ananthu Aniraj, Cassio F. Dantas, Dino Ienco, Diego Marcos

► **To cite this version:**

Ananthu Aniraj, Cassio F. Dantas, Dino Ienco, Diego Marcos. Masking Strategies for Background Bias Removal in Computer Vision Models. Out of Distribution Generalization in Computer Vision workshop (in conjunction with ICCV 2023), Oct 2023, Paris, France. pp.4399-4407, 10.1109/IC-CVW60793.2023.00474 . hal-04184449

**HAL Id: hal-04184449**

**<https://hal.science/hal-04184449v1>**

Submitted on 21 Aug 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# Masking Strategies for Background Bias Removal in Computer Vision Models

Ananthu Aniraj<sup>\*1,3,4,5</sup> Cassio F. Dantas<sup>2,3,5</sup> Dino Ienco<sup>2,3,5</sup> Diego Marcos<sup>1,3,4,5</sup>

<sup>1</sup>Inria <sup>2</sup>Inrae <sup>3</sup>University of Montpellier <sup>4</sup>LIRMM <sup>5</sup>UMR-Tetis  
{ananthu.aniraj, diego.marcos}@inria.fr {cassio.fraga-dantas, dino.ienco}@inrae.fr

## Abstract

Models for fine-grained image classification tasks, where the difference between some classes can be extremely subtle and the number of samples per class tends to be low, are particularly prone to picking up background-related biases and demand robust methods to handle potential examples with out-of-distribution (OOD) backgrounds. To gain deeper insights into this critical problem, our research investigates the impact of background-induced bias on fine-grained image classification, evaluating standard backbone models such as Convolutional Neural Network (CNN) and Vision Transformers (ViT). We explore two masking strategies to mitigate background-induced bias: Early masking, which removes background information at the (input) image level, and late masking, which selectively masks high-level spatial features corresponding to the background. Extensive experiments assess the behavior of CNN and ViT models under different masking strategies, with a focus on their generalization to OOD backgrounds. The obtained findings demonstrate that both proposed strategies enhance OOD performance compared to the baseline models, with early masking consistently exhibiting the best OOD performance. Notably, a ViT variant employing GAP-Pooled Patch token-based classification combined with early masking achieves the highest OOD robustness.

Our code and models are available at: [GitHub](#)

## 1. Introduction

The context in which an object is found is known to have a large influence in how it is perceived, both by humans [3] and machines [7]. When training deep learning-based computer vision models on any object-centric task, it is extremely challenging to ensure that the model does indeed only pay attention to the object of interest, since there exist often strong correlations between object characteristics and the background. Such a bias can lead to degraded



Figure 1: Background bias in fine-grained classification

performances when deploying the models on new, Out-of-Distribution (OOD), background configurations [1]. This is particularly harmful in fine-grained classification tasks such as species identification (Figure 1), due to the large number of closely related categories and the strong correlations between the object (the specimen of interest) and the background, which tends to relate to the habitat of the species [2, 12, 28].

The standard way to address this type of bias is to make the model detect the foreground and focus on it for classification, both when using Convolutional Neural Networks [19] and Transformers [13]. This reduces the influence of the background on the model’s decision, thus reducing the background-induced bias of the model.

Masking the background areas at some CNN high level representations seem like a simple and straight forward strategy to obtain this effect, as long as some type of foreground-background (FG-BG) mask can be obtained. After all, the inductive biases that characterize CNNs will induce an association between a high-level feature at a certain tensor location (i.e. before any global pooling) and the image patch at the corresponding image location. However, the Visual Transformers’ architecture does not encode for such biases, although they could be learned via supervised or self-supervised learning [4, 22].

We aim to analyse the impact of dataset bias induced by background regions in the image on the performance and generalization capabilities of image classification models

\*Corresponding Author

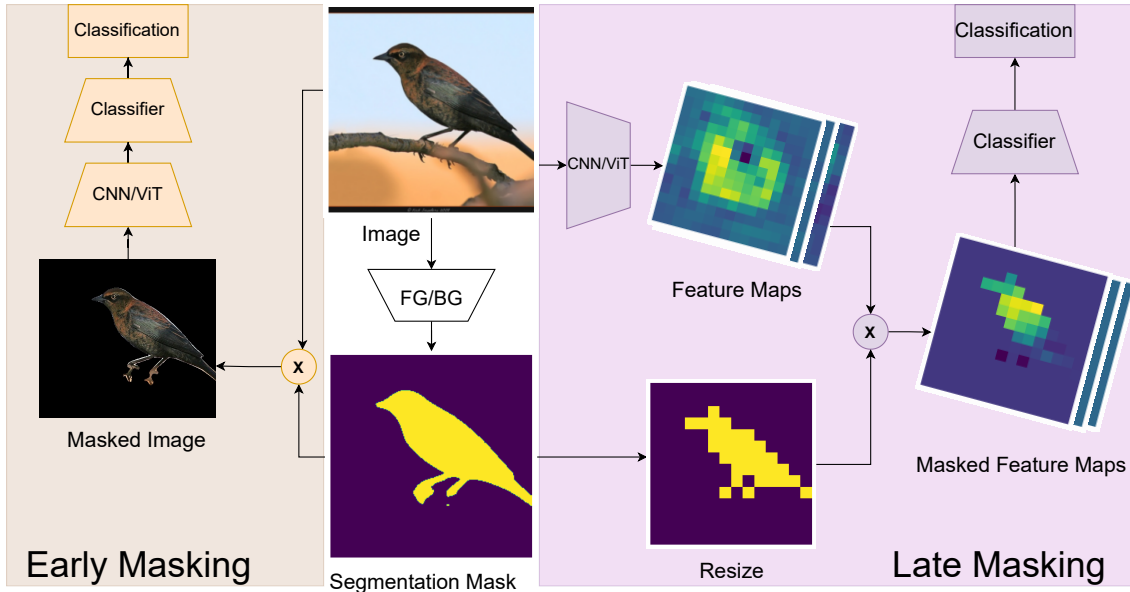


Figure 2: Early and late masking strategies

on fine-grained classification tasks using standard CNN and Vision Transformer (ViT) [11] backbones. Specifically, we study the effect of background masking, both at the image level and at high level features, on both CNN and ViT, on a fine-grained classification task. Our main motivation is to understand the difference in behaviour, in terms of generalization to OOD backgrounds, of ViT with respect to CNN under different strategies of background masking.

## 2. Related Work

**The influence of context** The ability of using context for object recognition is often considered an asset, since the context may help disambiguate between otherwise indistinguishable categories, and the explicit modeling of context played an important role in pre-deep learning computer vision systems [6, 7]. With the emergence of deep learning approaches, which are capable of automatically extracting features without leveraging user knowledge, the extraction of spurious correlations induced by contextual bias has started to be perceived as a potential issue [12, 26, 28]. Such effect can happen to the extent that CNN models may perform reasonably well when given only the background, without the object of interest [33].

**Mitigating context-induced bias** In early computer vision systems for object recognition, performing a pre-processing step of foreground segmentation was identified as an effective solution against background bias [23]. When the context classes are known, another approach is to make sure that the learned attention maps for each class do not

overlap [26]. Alternatively, the manifold of learned class features can be explored in order to manually identify spurious ones that can then be suppressed [21]. With modern Transformer-based architectures, it has been shown that allowing the model to refocus its attention in a top-down manner (thus based on the results after a first pass) helps in a variety of computer vision tasks [25] by reducing the effect of background elements.

In this work, we study the effect on OOD background generalization of masking the image background regions, at either the image level or at the level of an intermediate representation within the model.

## 3. Methodology

We aim at studying the effect of dataset bias introduced by the image background regions and to what extent it can be corrected via background masking. We propose to investigate the following three training settings:

- **Baseline:** Fine-tune a standard image classification model from the literature on a fine-grained image classification dataset.
- **Early masking:** Apply background masking at the image level using foreground-background (FG-BG) masks. We then fine-tune image classification models to classify the masked out image.
- **Late masking:** Mask out high-level representations that spatially correspond to background regions at an intermediate stage of an image classification model us-

ing the FG-BG masks. Subsequently, we fine-tune the resulting model.

The FG-BG masks are obtained by applying a semantic segmentation network trained on ground-truth foreground-background masks.

In order to investigate the generalization capabilities of the models under the different settings, we propose to evaluate them on a test set that has a different background distribution than the train set.

### 3.1. Binary FG-BG Segmentation

For binary segmentation, we trained a deep neural network using the segmentation labels provided by the CUB-200-2011 dataset [27]. The objective was to classify each pixel in the image as either bird or background.

Given an image  $\mathbf{x} \in \mathbb{R}^{3 \times M \times N}$ , the segmentation network was trained to produce a binary mask  $\mathbf{m} \in \{0, 1\}^{1 \times M \times N}$ , where 0 represents the background, and 1 represents the foreground.

### 3.2. Early Masking

For this strategy (Figure 2 left), we first mask out the background image regions using the binary segmentation network and then pass the masked image to a standard CNN/Transformer-based image classifier model. The CNN/ViT model is composed of two stages: a backbone  $h_{\theta_1}(\cdot)$  (CNN or ViT) and a classification head  $g_{\theta_2}(\cdot)$ . More formally, we can define it as:

$$\mathbf{y} = g_{\theta_2}(h_{\theta_1}(\mathbf{x} \odot \mathbf{m})) \quad (1)$$

where each channel of the image ( $\mathbf{x}$ ) is multiplied element-wise by the FG-GB binary mask ( $\mathbf{m}$ ) and  $\mathbf{y} \in \mathbb{R}^C$  is the vector of class logits.

The idea here is to mask out background features at the image level, forcing the model to only look at the salient object in the image for the classification task, ideally learning representations independent of bias caused by the image background.

### 3.3. Late Masking

In this strategy (Figure 2 right), the high-level spatial features corresponding to the background are masked out, similar to the feature masking layer proposed in [10]. More formally, this strategy can be defined as:

$$\mathbf{y} = g_{\theta_2}(h_{\theta_1}(\mathbf{x}) \odot \mathbf{m}') \quad (2)$$

Here,  $\mathbf{y}$  represents the final classification logits,  $\mathbf{x}$  is the input image, and  $\mathbf{m}'$  is the subsampled version of the binary FG-BG mask.

The backbone  $h_{\theta_1}(\cdot)$  outputs a spatial tensor  $\mathbf{z}$ .

$$\mathbf{z} = h_{\theta_1}(\mathbf{x}) \in \mathbb{R}^{D \times M' \times N'}$$

where  $M' = M/k$  and  $N' = N/k$ , with  $k$  being the model’s spatial downsampling ratio. In the case of ViT, this spatial tensor results from the spatial rearrangement of the patch tokens’ features, each a vector of dimension  $D$ .

The classification head  $g_{\theta_2}(\cdot)$  performs a global average pooling followed by at least one learnable linear layer.

The late masking strategy allows selectively masking out background-related features at a later stage in the model, enabling the model to learn representations based solely on foreground image features.

## 4. Experiments

### 4.1. Dataset

All our models are trained on the training set of CUB [27]. This dataset contains images of 200 bird species, with 5,994 images in the training set and 5,794 images in the testing set.

The trained models are evaluated on two test sets: (i) the in-distribution real-world images of the CUB test set [27] and (ii) an Out-Of-Distribution (OOD) set of images with an adversarial background, specifically the Waterbirds dataset from [24].

The Waterbirds dataset [24] was constructed by replacing the background regions in the CUB images with ones coming from the Places dataset [32]. Therefore, this dataset contains the exact same bird species as CUB, with the only difference being the background.

### 4.2. Implementation Details

All models were implemented in PyTorch. For the sake of comparison, we used the ImageNet-pretrained ConvNeXt [16] and DinoV2-pretrained ViT [22] backbones as starting weights for all the experiments.

We utilized the Small (S), Base (B), and Large (L) variants of the ViT and ConvNeXt models for our experiments. These variants allowed us to explore the impact of model size on the performance of our masking strategies.

To perform binary segmentation, we fine-tuned a pre-trained Mask2Former [5] model with a Swin-Tiny backbone [15] using the FG-BG masks provided along with the CUB dataset. We use only the images from the CUB training dataset to train the segmentation model. The Mask2Former model was chosen for its effectiveness in image segmentation tasks. However, in theory, any semantic segmentation model can be used in its place. Detailed information regarding the training settings for the binary segmentation model, as well as the evaluation results, can be found in the supplementary material.

### 4.3. Training Settings

For all our experiments, we used two different versions of training settings:

Models	ConvNeXt-S/B/L	ViT-S/B/L
Input Size	(224, 224)	(518, 518)
Optimizer	AdamW [17]	AdamW [17]
Base LR	4e-3	1e-3
Weight Decay	5e-2	1e-2
Optimizer momentum	(0.9, 0.999)	(0.9, 0.999)
Batch Size	128	128
Training Epochs	90	90
Learning Rate Schedule	Cosine Decay	Cosine Decay
Warmup Epochs	None	None
Warmup Schedule	N/A	N/A
RandAugment [9]	(9, 0.5)	(9, 0.5)
Mixup [30]	None	None
Cutmix [29]	None	None
Random Erasing [31]	0.25	0.25
Label Smoothing [20]	0.1	0.1
Stochastic Depth [14]	0.1/ 0.2/ 0.3	0.0
Frozen Backbone	✓	✓

Table 1: Training Settings for frozen backbone

Models	ConvNeXt-S/B/L	ViT-S/B/L
Input Size	(224, 224)	(518, 518)
Optimizer	AdamW [17]	AdamW [17]
LR(Backbone)	4e-6	4e-6
LR(Classifier)	4e-3	4e-3
Weight Decay	5e-2	5e-2
Optimizer momentum	(0.9, 0.999)	(0.9, 0.999)
Batch Size	64	64
Training Epochs	300	300
Learning Rate Schedule	Cosine Decay	Cosine Decay
Warmup Epochs	20	20
Warmup Schedule	Linear	Linear
RandAugment [9]	(9, 0.5)	(9, 0.5)
Mixup [30]	0.8	0.8
Cutmix [29]	1.0	1.0
Random Erasing [31]	0.25	0.25
Label Smoothing [20]	0.1	0.1
Stochastic Depth [14]	0.1/ 0.2/ 0.3	0.1
Frozen Backbone	✗	✗

Table 2: Training Settings for fine-tuning

- (i) For models with a frozen backbone, we utilized the settings described in Table 1. In this case, only the final classification layer was trained, and we employed a relatively short training schedule of 90 epochs.
- (ii) For models with fine-tuning, we employed the settings described in Table 2. In this case, all layers in the network were trained for a longer schedule of 300 epochs, with additional regularization techniques like MixUp [30] and CutMix [29].

#### 4.4. The Effect of Early Masking and Model Size

The main aims of this experiment are to investigate:

- (i) The impact of adopting our early masking strategy.
- (ii) The interplay between this masking strategy and varying model sizes.

For this experiment, we train classifiers while keeping the ViT and ConvNeXt backbones of three different sizes frozen. The training is performed on the CUB dataset, with two scenarios: one involving input images masked using the predicted FG-BG masks and the other without any masking. To extract features for classification in the ViT model, we concatenate the class token with the global average-pooled (GAP) patch token features, following a similar implementation as described in [22].

To ensure a comprehensive evaluation, we also assess the model trained on masked images with the original images and vice versa. This provides insights into the effectiveness of the early masking strategy and its sensitivity to different model sizes.

#### 4.5. Baseline vs Early Masking vs Late Masking

The aims of this experiment are to study:

- (i) The effects of applying both of our proposed masking strategies.
- (ii) The performance with a frozen and a fine-tuned backbone

For this experiment, we chose two models of approximately the same size: the ViT-Base and the ConvNeXt-Base, due to their balanced compromise between performance and computational cost.

#### 4.6. Feature masking at different model stages

The main aim of this experiment is to study the effect of applying late feature masking at different stages in the network. In this scenario, early masking would be equivalent to applying masking at stage  $L = 0$ .

Concretely, we experimented with performing the feature masking after the second-to-last stage of the model ( $L - 1$ ) and compared the OOD and in-distribution performance with the masking after the last stage ( $L$ ) and the early masked model ( $L = 0$ ).

The training settings for this experiment are similar to the fine-tuning settings given in Table 2. Here, the model training requires two learning rates: one before and one after the feature masking, to ensure convergence.

We were only able to perform this experiment for the ConvNeXt models, as we encountered challenges in finding stable training hyperparameters for masking the ViT at the  $(L - 1)^{th}$  stage.

#### 4.7. Varying the ViT representation

In this experiment, we aim to:



Training on CUB		CUB(%)		Waterbird(%)	
		Test on original	Test on masked	Test on original	Test on masked
ConvNeXt-S	Baseline	<b>86.56</b>	78.60	55.82	67.12
	Masked	83.84	84.43	64.21	<b>77.33</b>
ConvNeXt-B	Baseline	<b>88.00</b>	77.52	65.96	68.51
	Masked	84.36	85.69	67.21	<b>80.10</b>
ConvNeXt-L	Baseline	<b>88.05</b>	78.68	66.83	70.69
	Masked	87.22	87.38	73.67	<b>82.81</b>
ViT-S	Baseline	<b>88.26</b>	82.27	71.15	77.59
	Masked	85.89	87.92	78.61	<b>84.05</b>
ViT-B	Baseline	89.20	87.69	76.65	83.00
	Masked	88.35	<b>90.10</b>	82.15	<b>86.93</b>
ViT-L	Baseline	89.79	88.91	80.76	85.61
	Masked	88.35	<b>91.06</b>	84.67	<b>88.30</b>

Table 3: Results: Early Masking vs Baseline

- (i) Vary the input to the ViT classification head to explore different ablations of the information extracted by the ViT model.
- (ii) Investigate the performance differences between these ablations when the backbone is frozen versus when the entire model is fine-tuned.

More precisely, we conduct experiments and evaluate the performance using three different inputs for the linear classification layer: (i) using the CLS token, (ii) using the Global Average Pooled (GAP) Patch token, and (iii) concatenating them together (CLS+Patch). To manage computational constraints, we focus solely on the ViT-Base model for this study.

## 5. Results and Discussion

### 5.1. The Effect of Early Masking and Model Size

The performance of models trained with a frozen backbone, both with and without the early masking strategy, is evaluated using all combinations of test settings. The results of these evaluations are reported in [Table 3](#).

From the table, it is evident that the models trained with the early masking technique consistently outperform their counterparts trained without the masking strategy on the OOD Waterbirds test set, regardless of the backbone and model size. This improvement holds true when the early-masked models are tested on both the original and masked images. For instance, with ConvNeXt-B, the OOD performance improves from 65.96% to 80.10% when the images are masked at both training and testing time, and for ViT-B, it increases from 76.65% to 86.93%.

These results suggest that models trained on the origi-

nal images are influenced by biases from the image backgrounds, which leads to lower generalization on the Waterbird test set when the background is not masked.

We also observe a performance boost for the baseline models when tested on the early masked version of the Waterbirds Dataset. This suggests that these models might have been relying on background cues to make decisions, which the early masking strategy helps to mitigate.

Furthermore, the best overall performances on the original CUB dataset are obtained by ViT-L and ViT-B when early masking is performed both at train and test stage. In contrast, when using convolutional models, this setting performs less well than the original one. This suggests that: (i) ViT models are less sensitive than CNNs to potential artifacts induced by masking and (ii) the original CUB dataset is negatively affected by background-induced biases.

### 5.2. Baseline vs Early Masking vs Late Masking

#### Effect of freezing the model backbone

The results are given in [Table 4](#). Upon analyzing the table, it is evident that the early masking strategy shows the highest generalization performance on the OOD Waterbird test set for both backbones.

For the ConvNeXt model with a frozen backbone, we observe a substantial performance improvement when employing both the early and late masking training strategies. This suggests that the high-level features spatially corresponding to the foreground are highly localized [18], likely due to the inductive biases in the CNN architecture, resulting in high spatial correlation of features even at the later stages of the network.

In the case of the ViT model with a frozen backbone,

Backbone	Training on CUB	CUB(%)		Waterbird(%)	
		Frozen	Fine-tuned	Frozen	Fine-tuned
ConvNeXt-B	Baseline	88.00	89.62	65.96	76.20
	Early-Masked	85.69	<b>90.31</b>	80.10	<b>87.01</b>
	Late-Masked	87.66	88.76	77.95	78.42
ViT-B	Baseline	89.20	89.38	76.55	68.36
	Early-Masked	90.10	<b>91.37</b>	86.93	<b>88.81</b>
	Late-Masked	88.61	90.73	76.55	74.76

Table 4: Results of 4.5 - Frozen backbone vs Fine-tuning

the late-masked and baseline models perform similarly on both the in-distribution CUB dataset and the OOD Waterbird dataset. This indicates that the early masking strategy helps prevent the ViT model, with frozen backbone parameters, from being influenced by background-related biases.

### Effect of fine-tuning

Fine-tuning consistently improves in-distribution performance on the CUB test set for both ConvNeXt and ViT models (Table 4). However, we do not observe a similar behavior on OOD data.

The early-masked model, for both ViT and CNN, demonstrates the best generalization, consistent with the findings from previous experiments.

For ConvNeXt, fine-tuning the baseline and early masked models leads to improvements in both in-distribution performance and significant advances in OOD generalization. However, the late-masked model shows only marginal performance gains.

In the case of the ViT, we see a reduction in robustness for both the baseline and late-masking variant on the OOD Waterbirds test set, with the baseline experiencing a more evident performance drop-off. This indicates that, with longer training, the ViT model becomes more susceptible to overfitting when background correlations are present in the training data, unless the background is masked out at the image level.

### 5.3. Feature masking at different model stages

The results of adopting feature masking at different stages in the network are presented in Table 5. It shows that both the in-distribution and OOD performance systematically improve when background masking is applied at an earlier stage in the network. This observation suggests that earlier layers of the CNN are more spatially correlated with the image patches at their corresponding locations, possibly due to the comparatively smaller receptive fields.

Furthermore, we can note that the improvement in robustness of the models tends to decrease as the CNN model

Backbone	Feature Masking	CUB(%)	Waterbird(%)
ConvNeXt-S	$L$	88.73	77.19
	$L - 1$	89.35	81.22
	$0$	<b>90.73</b>	<b>87.95</b>
ConvNeXt-B	$L$	88.76	78.42
	$L - 1$	89.04	80.04
	$0$	<b>90.31</b>	<b>87.01</b>
ConvNeXt-L	$L$	89.80	79.39
	$L - 1$	89.49	79.68
	$0$	<b>90.99</b>	<b>88.19</b>

Table 5: Results of adopting feature masking at different stages. In this table,  $L$ : after last stage,  $(L - 1)$ : after second last stage.  $0$ : early masking/at image level

size increases, unless the masking is done at the image level itself. For instance, the ConvNeXt-Large model performs approximately the same regardless of whether the masking was applied after the last stage or the second last stage.

Overall, these results suggest that feature masking at an earlier layer of the network has a positive impact on both in-distribution and OOD performances.

### 5.4. Varying the ViT representation

The results of the experiment are reported in Table 6. Fine-tuning the models generally improves in-distribution performance on the CUB dataset. Notably, the model using only Patch tokens exhibits a substantial accuracy increase from 62% to 90% for the baseline, 72% to 92% for the early masked, and 84% to 91% for the late-masked variants.

On the OOD Waterbird dataset, fine-tuning the baseline model using both CLS and Patch tokens leads to reduced performance. This pattern also repeats for the model using just the CLS token. However, we see a notable improvement for the model using just the Patch tokens, where the accuracy increases from 25% to 68%.

In contrast, the early and late-masked models benefit from fine-tuning, showing improved OOD generalization with either CLS or Patch token input. Particularly, the ViT

Backbone	Training on CUB	ViT representation	CUB(%)		Waterbird(%)	
			Frozen	Fine-tuned	Frozen	Fine-tuned
ViT-B	Baseline	CLS	90.04	<b>90.47</b>	80.35	<b>71.35</b>
		Patch	62.24	90.05	24.71	67.74
		CLS+Patch	89.20	89.38	76.65	68.36
	Early-Masked	CLS	90.33	<b>91.73</b>	86.81	88.60
		Patch	72.17	91.51	66.37	<b>89.22</b>
		CLS+Patch	90.10	91.37	86.93	88.81
	Late-Masked	CLS	89.16	90.78	75.10	80.54
		Patch	84.15	<b>90.85</b>	71.63	<b>84.50</b>
		CLS+Patch	88.61	90.73	76.55	74.76

Table 6: Results of varying the representation of the ViT.

model employing GAP-Pooled Patch token classification, considering both early and late masking strategies, clearly outperforms all the other variants.

The experiment highlights the importance of studying which representations are utilized for the classifier layer in ViT models to enhance performance in fine-grained image classification tasks, particularly in the presence of background-induced biases. The use of Patch tokens, especially in conjunction with early or late masking, emerges as a promising strategy. This approach allows the model to focus on foreground-containing tokens and enhances model generalization with respect to background-induced biases.

## 6. Limitations and Future Work

Our study provides valuable insights and lays the foundation for potential solutions; however, certain limitations point to avenues for future exploration:

**Dataset Scope and Pre-trained Models:** The research study was mainly restricted to the CUB dataset, and we utilized publicly available pre-trained models due to time and computational constraints. Expanding our assessment to encompass larger datasets featuring diverse species and backgrounds, while also maintaining better control over pre-training methodologies, will enhance the generalizability and robustness of our conclusions.

**Real-World Robustness:** Our reliance on FG-BG ground truth annotations during segmentation model training may not align with scenarios where such privileged information is unavailable. Deploying our proposed models on real-world datasets without access to FG-BG annotations could yield further insights into its robustness.

**Computational Overhead:** Our proposed masking strategies involving the segmentation model introduce computational overhead, which could be infeasible in scenarios with tight time constraints. To this end, future research could explore techniques that seamlessly integrate back-

ground suppression as a training objective, aiming for enhanced out-of-distribution generalization. Notably, recent work by Chou et al. [8] also emphasizes similar directions in the context of fine-grained image classification.

These limitations underscore the potential for future research endeavors aimed at refining and expanding the effectiveness of our proposed methods.

## 7. Conclusion

In this study, we conducted extensive experiments to investigate the impact of background-induced bias on the generalization capabilities of CNN and ViT models for fine-grained image classification on out-of-distribution (OOD) data and explored various background masking strategies to mitigate such biases and enhance model generalization.

Overall, our results revealed that the proposed masking strategies improve OOD performance, with the early masking strategy exhibiting the best generalization capability on both CNN and ViT. Early masking is particularly advantageous when fine-tuning the model backbones, with late masking not benefiting from fine-tuning and even losing some OOD generalization capacity in the case of ViT. Notably, a ViT variant employing GAP-Pooled Patch token-based classification, combined with early masking, exhibits the highest OOD robustness of all tested settings.

While our study provides valuable insights and solutions, certain limitations remind us of unexplored frontiers. Expanding investigations to larger datasets, controlled pre-training methodologies, and considering real-world scenarios lacking FG-BG annotations are essential steps toward refining and extending the impact of our methods.

In conclusion, our study emphasizes the significance of considering background information in object-centric tasks and demonstrates effective solutions to mitigate biases and enhance model generalization.



## References

- [1] A. Barbu, D. Mayo, J. Alverio, W. Luo, C. Wang, D. Gutfreund, J. Tenenbaum, and B. Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32, 2019.
- [2] S. Beery, G. Van Horn, and P. Perona. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pages 456–473, 2018.
- [3] I. Biederman, R. J. Mezzanotte, and J. C. Rabinowitz. Scene perception: Detecting and judging objects undergoing relational violations. *Cognitive psychology*, 14(2):143–177, 1982.
- [4] M. Caron, H. Touvron, I. Misra, H. Jegou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging Properties in Self-Supervised Vision Transformers. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9630–9640, 2021.
- [5] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girshik. Masked-attention Mask Transformer for Universal Image Segmentation. In *CVPR*, 2022.
- [6] M. J. Choi, A. Torralba, and A. S. Willsky. A tree-based context model for object recognition. *IEEE transactions on pattern analysis and machine intelligence*, 34(2):240–252, 2011.
- [7] M. J. Choi, A. Torralba, and A. S. Willsky. Context models and out-of-context objects. *Pattern Recognition Letters*, 33(7):853–862, 2012.
- [8] P.-Y. Chou, Y.-Y. Kao, and C.-H. Lin. Fine-grained Visual Classification with High-temperature Refinement and Background Suppression. (March):1–9, 2023.
- [9] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le. RandAugment: Practical automated data augmentation with a reduced search space. In *Advances in Neural Information Processing Systems*, volume 2020–Decem, 2020.
- [10] J. Dai, K. He, and J. Sun. Convolutional feature masking for joint object and stuff segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 07–12–June:3992–4000, 2015.
- [11] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An Image Is Worth 16X16 Words: Transformers for Image Recognition At Scale. *ICLR 2021 - 9th International Conference on Learning Representations*, 2021.
- [12] R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.
- [13] M. Hiller, R. Ma, M. Harandi, and T. Drummond. Rethinking generalization in few-shot classification. *Advances in Neural Information Processing Systems*, 35:3582–3595, 2022.
- [14] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger. Deep Networks with Stochastic Depth. In B. Leibe, J. Matas, N. Sebe, and M. Welling, editors, *Computer Vision – ECCV 2016*, pages 646–661, Cham, 2016. Springer International Publishing.
- [15] 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. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9992–10002, 2021.
- [16] Z. Liu, H. Mao, C. Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie. A ConvNet for the 2020s. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2022–June:11966–11976, 2022.
- [17] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. *7th International Conference on Learning Representations, ICLR 2019*, 2019.
- [18] W. Luo, Y. Li, R. Urtasun, and R. Zemel. Understanding the effective receptive field in deep convolutional neural networks. *Advances in Neural Information Processing Systems (Nips)*:4905–4913, 2016.
- [19] X. Luo, L. Wei, L. Wen, J. Yang, L. Xie, Z. Xu, and Q. Tian. Rectifying the shortcut learning of background for few-shot learning. *NeurIPS*, 34:13073–13085, 2021.
- [20] R. Müller, S. Kornblith, and G. Hinton. When does label smoothing help?, 2019.
- [21] Y. Neuhaus, M. Augustin, V. Boreiko, and M. Hein. Spurious features everywhere—large-scale detection of harmful spurious features in imagenet. *arXiv preprint arXiv:2212.04871*, 2022.
- [22] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. DINOv2: Learning Robust Visual Features without Supervision. *arXiv preprint arXiv:2304.07193*, 2023.
- [23] A. Rosenfeld and D. Weinshall. Extracting foreground masks towards object recognition. In *2011 International Conference on Computer Vision*, pages 1371–1378. IEEE, 2011.
- [24] S. Sagawa, P. W. Koh, T. B. Hashimoto, and P. Liang. Distributionally Robust Neural Networks for Group Shifts: on the Importance of Regularization for Worst-Case Generalization. In *8th International Conference on Learning Representations, ICLR 2020*, 2020.
- [25] B. Shi, T. Darrell, and X. Wang. Top-down visual attention from analysis by synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2102–2112, 2023.
- [26] K. K. Singh, D. Mahajan, K. Grauman, Y. J. Lee, M. Feiszli, and D. Ghadiyaram. Don’t judge an object by its context: Learning to overcome contextual bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11070–11078, 2020.
- [27] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010.
- [28] K. Y. Xiao, L. Engstrom, A. Ilyas, and A. Madry. Noise or signal: The role of image backgrounds in object recognition.

- tion. In *International Conference on Learning Representations*, 2020.
- [29] S. Yun, D. Han, S. Chun, S. J. Oh, J. Choe, and Y. Yoo. CutMix: Regularization strategy to train strong classifiers with localizable features. *Proceedings of the IEEE International Conference on Computer Vision*, 2019-Octob:6022–6031, 5 2019.
- [30] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. MixUp: Beyond empirical risk minimization. In *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, 10 2018.
- [31] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang. Random erasing data augmentation. In *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, pages 13001–13008, 8 2020.
- [32] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba. Places: A 10 Million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1452–1464, 2018.
- [33] Z. Zhu, L. Xie, and A. Yuille. Object recognition with and without objects. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 3609–3615, 2017.