MAAT: MASSIVE-ACTIVATION-AWARE TRANSFERABLE ADVERSARIAL ATTACKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Transfer-based adversarial attacks optimise a perturbation on a surrogate model and apply it to target models without access to them. Such attacks can include objectives defined on intermediate representations, whose values depend on how representation magnitude is distributed across coordinates. We ask whether this distribution can be exploited to improve transfer. In DINOv2-L/14 and CLIP-L/14, the representation block used by the attack contains a token whose norm exceeds that of a typical token by 8.6 and 29.6 times, respectively, with the squared magnitude of that token concentrated in an effective dimensionality of only 6.0 and 1.2 channels out of 1024. A clean-data calibration identifies the corresponding token–channel coordinates in 99.7–99.8% of image–channel pairs. We do not assume that these coordinates are intrinsically non-transferable. Instead, MAAT uses this structure to control which channels enter the feature objective, while leaving the task loss and update rule unchanged. This makes the intervention compatible with attacks that modify the gradient, forward pass, backward pass, or input. Across two surrogates, eight targets, and four base attacks, MAAT increases mean transfer attack success rate by about four percentage points. Matched random-channel controls with six seeds support the channel selection effect for CLIP-L/14 on all seven transfer targets; the same evidence is absent for DINOv2-L/14, where an ablation identifies the source of the difference. These results show that intermediate-activation structure can be used to shape transferable adversarial objectives, with the effect depending on the surrogate.

1 INTRODUCTION

Transfer-based adversarial attacks optimise a perturbation against a surrogate model and apply it, unchanged and without queries, to a target the attacker never sees (Goodfellow et al., 2014; Dong et al., 2018). Transferability depends on similarities between surrogate and target models, but which properties of the surrogate representation can be exploited to improve transfer remains an empirical question.

Vision models are increasingly built on pretrained Vision Transformer backbones (Dosovitskiy, 2020; Radford et al., 2021; Oquab et al., 2023). At the same time, large Transformers exhibit pronounced concentration in intermediate representations: a small number of coordinates can have magnitudes far above the rest. In language models, related structures have been described as near-constant channels (Sun et al., 2024); in vision Transformers, high-norm tokens have been associated with low-information image patches (Darcet et al., 2024). These studies characterize the structure of model representations, not its role in transferable adversarial attacks. Moreover, they do not establish that the same coordinates or channels are shared across independently developed models, which we do not assume.

We therefore ask whether this representation structure can be used to design a more transferable adversarial objective. Specifically, does excluding channels associated with massive activations from the feature objective change transferability? We answer this question through clean-data calibration and matched random-channel controls.

We proceed in three steps. First, we characterize massive activations in the intermediate representations used by our attack and examine how their location varies across models (Section 3). Second, we show that clean-data calibration identifies channels associated with the measured high-norm

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

structure (Section 4). Finally, we introduce MAAT (Massive-Activation-Aware Transferable Adversarial Attacks), which excludes these calibrated channels from the feature objective, and evaluate this intervention across surrogates, targets, datasets, and base attacks (Sections 5–6).

Our contributions are summarized as follows:

- We characterize massive-activation structure in the intermediate representations used by transferable adversarial attacks and examine how its location varies across models.
- We show that clean-data calibration identifies channels associated with the measured high-norm structure.
- We introduce MAAT, which excludes these calibrated channels from the feature objective, and evaluate the resulting transferability across multiple surrogates, targets, datasets, and base attacks using matched random-channel controls.

2 Related Work

Vision foundation models such as CLIP (Radford et al., 2021), DINOv2 (Oquab et al., 2023), and supervised ViTs (Dosovitskiy, 2020; Touvron et al., 2022) are widely used as backbones for downstream tasks, either frozen with a trained head or fine-tuned. Consequently, independently developed models can share representation structure inherited from the same pretrained backbone.

Massive activations. Large Transformers can have highly concentrated activation magnitudes. Sun et al. (2024) show in LLMs that a few activations can be orders of magnitude larger than the rest and vary little across inputs. In vision Transformers, Darcet et al. (2024) identify high-norm artefact tokens in low-information patches and show that register tokens can remove them. These works characterize such structures as properties of representations and computation; their role in adversarial attacks remains unstudied.

Transferable attacks. Transfer-based attacks optimize a perturbation on a surrogate and apply it to an unseen target (Goodfellow et al., 2014; Madry et al., 2017). Existing methods modify gradient accumulation (e.g., momentum (Dong et al., 2018), variance tuning (Wang & He, 2021)), diversify inputs via resizing/padding (Xie et al., 2019), translation-invariant transformations (Dong et al., 2019), scale copies (Lin et al., 2019), or image mixing (Wang et al., 2021), or exploit architecture-specific properties such as ViT attention (Wei et al., 2022). Another line changes the objective to operate on intermediate representations: ILA (Huang et al., 2019) follows a direction from a baseline attack, SSP (Naseer et al., 2020) maximizes perceptual changes in self-supervised features, and forward-pass refinement (Ren et al., 2025) modifies intermediate computation during the attack. Thus, transferability work studies optimization, inputs, architecture, and objectives, while massive-activation work characterizes concentrated representation structure. Their interaction—whether massive-activation coordinates can be identified from clean surrogate data and whether excluding them changes transferability—has not been systematically studied.

3 Massive Activations Concentrate Representation Magnitude

Visual foundation models can exhibit massive activations, where a small number of coordinates have substantially larger magnitudes than the rest (Sun et al., 2024; Darcet et al., 2024). Before introducing any intervention, we quantify this structure in the intermediate representations our attack reads. We measure two properties of the highest-norm token: how much larger its norm is than that of a typical token, and how few channels account for its squared norm. This section reports what the representation looks like; it makes no claim about which coordinates do or do not transfer, which is tested in Section 6.

3.1 Concentration within a representation

Let $H^\ell(x) \in \mathbb{R}^{N\times D}$ denote the output of the MLP projection of block ℓ, with tokens indexed by $n \in \{1,\dots,N\}$ and channels by $c \in \{1,\dots,D\}$. For each image we identify the token with the

largest ℓ_2 norm,

$$n^\star(x) = \arg\max_n \left\| H^\ell_{n,:}(x) \right\|_2,$$

and compare it with the token of median norm in the same image. Every quantity below is computed per image and then averaged over the same 500 ImageNet validation images; none of them describes a single image or an averaged representation.

The highest-norm token is much larger than a typical token in both surrogates. At block 19 of DINOv2-L/14 its norm is 8.6 times the median token norm; at block 12 of CLIP-L/14 the ratio is 29.6.

Its magnitude is also concentrated in few channels. Writing $v_c = H^\ell_{n^\star,c}(x)$, we use the participation ratio

$$\mathrm{PR}(v) = \frac{\left(\sum_{c=1}^D v_c^2\right)^2}{\sum_{c=1}^D v_c^4},$$

which takes values in $[1, D]$: it equals D when all $|v_c|$ are equal, and 1 when a single coordinate is non-zero. The highest-norm token has a mean participation ratio of 6.0 out of 1024 channels for DINOv2-L/14 and 1.2 for CLIP-L/14, against 158.7 and 254.4 for the median-norm token of the same images. The five channels with the largest $|v_c|$ account for 0.69 and 0.96 of its squared norm, respectively.

In both surrogates, then, the highest-norm token is both much larger than a typical token and concentrated in a small subset of channels, with an effective dimensionality of about six channels for DINOv2-L/14 and close to one for CLIP-L/14.

3.2 Locating the concentration across blocks

Figure 1 repeats both measurements at every block. A large token norm and strong within-token concentration occur together in few blocks: 11–12 of 24 for CLIP-L/14 and 17–19 for DINOv2-L/14, where the effective dimensionality of the highest-norm token is far below the 100–330 effective channels seen elsewhere.

The two properties are not interchangeable. DINOv2-L/14 is also strongly concentrated at blocks 21–22, with effective dimensionalities of 3.5 and 2.2 channels, but there the highest-norm token is only 1.6 and 2.2 times the median in norm, against 8.6–15.1 in the earlier window. Concentration alone therefore does not single out the earlier window.

For each surrogate we select the last block of the first contiguous run of blocks in which the mean ratio of highest-norm to median token norm is at least T. This criterion uses the norm measurement only; the concentration measurement is reported because it co-occurs there, not because the rule consults it. For any T between 4 and 8 the rule returns block 19 for DINOv2-L/14 and block 12 for CLIP-L/14. We state plainly that the rule was written down after those two blocks had been selected, that it has been applied to no surrogate outside these two, and that its stability in T therefore describes the separation visible in Figure 1 rather than an independent validation.

Appendix A reports the peak-block measurement for fifteen models, including DINOv2, CLIP and supervised ViT backbones, and models additionally fine-tuned for e-commerce classification and FoodSeg103 segmentation. The peak block ranges from mid-network to the final block with no common relative depth, so it must be determined per surrogate; within a backbone, fine-tuning for a different task moves it little. That scan is descriptive and does not test the selection rule above.

The observation carried into the next section is narrow: at a surrogate-specific block, one high-norm token accounts for a large part of the representation's magnitude through a few channels.

4 The Calibrated Channels Recover the High-Norm Token

Section 3 located the representation's magnitude at one high-norm token and a few of its channels. Our attack does not identify that token by computing token norms; it selects channels, by calibrating channel-wise activation magnitudes on clean images. This section checks that the two procedures pick out the same thing, which is what makes the measurement of Section 3 relevant to an objective defined over channels.

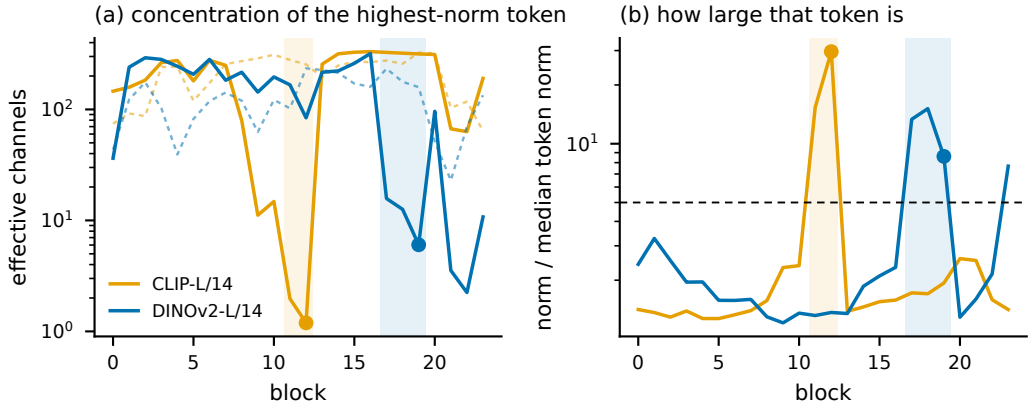


Figure 1: **(a)** Effective dimensionality of the highest-norm token's activation vector at every block, for the two surrogates, with the median-norm token of the same images as a dashed reference. **(b)** The ratio of that token's norm to the median token's. Both panels show means over 500 images. The shaded span is the window used to select the block the attack reads; the dot marks its last block. Blocks 21–22 of DINOv2-L/14 are concentrated but not norm-exceptional.

Channel selection. At the block read by the attack, $\ell^\star$ (Section 3.2), define

$$s_c(x) = \max_n \big|H^{\ell^\star}_{n,c}(x)\big|, \qquad a_c = Q_q\big(\{s_c(x)\}\big),$$

the q-quantile of s_c over the calibration images, with $q = 0.05$. The low quantile favours channels whose maximum-over-token magnitude stays large across images over channels that are large on a few of them. Let $\mathcal{C}$ be the $k = 5$ channels with the largest a_c. For each of them we locate the token at which it peaks:

$$t_c(x) = \arg\max_n \big|H^{\ell^\star}_{n,c}(x)\big|, \qquad c \in \mathcal{C}. \tag{1}$$

The selected channels identify the same token. We compare $t_c(x)$ with $n^\star(x) = \arg\max_n \|H^{\ell^\star}_n(x)\|_2$ over 500 images and $k = 5$ channels, giving 2500 image–channel pairs per surrogate. The two coincide in 99.8% of pairs for DINOv2-L/14 and 99.7% for CLIP-L/14; in the remaining pairs the channel's peak is still among the three highest-norm tokens. Because an image in which two selected channels disagree contributes at least one mismatched pair, at most five of the 500 images can contain such a disagreement.

The channel-selection procedure therefore recovers, almost exactly, the same high-norm token identified in Section 3. This is a consistency check between two ways of reading the same representation, not a second, independent observation: both quantities are maxima over tokens of the same activations, one taken after an ℓ_2 reduction over channels and one per channel.

Where the token sits. The high-norm token is always a patch token; on neither surrogate does a peak fall on the class token. Its position depends on the input, occupying 80 of the 256 patch positions on DINOv2-L/14 and 156 on CLIP-L/14, with no single position accounting for more than 15.6% and 6.6% of images; Appendix C gives the distributions. Content-dependent high-norm patch tokens are consistent with the artefact tokens of Darcet et al. (2024), but we do not establish that the tokens here have the same origin and refer to them only as high-norm tokens.

What this licenses. The calibrated set $\mathcal{C}$ is, almost always, the channels carrying the exceptional magnitude of one token. Hence any objective that normalizes by the clean representation's scale over all $N \times D$ entries has its denominator largely determined by $\mathcal{C}$ at that token. We do not claim that the attack's optimization is dominated by these coordinates, nor that they fail to transfer: $\mathcal{C}$ is surrogate-specific, and channel indices have no correspondence across models. Whether excluding $\mathcal{C}$ from the feature objective changes transferability is tested empirically in Section 5.

5 MAAT: A FEATURE TERM BUILT AROUND THE CALIBRATED CHANNELS

MAAT changes how the feature term of the objective treats the calibrated set $\mathcal{C}$ of Section 4: the term rewards change in the complement of $\mathcal{C}$ and penalises change in $\mathcal{C}$ itself at the token where those channels peak. The task term and the update rule are unchanged, so it can be combined with different base attacks.

What is excluded. $\mathcal{C}$ is computed once from 500 clean images of the dataset being attacked, using no labels and no queries to any target model. Those are the first 500 of the images later attacked rather than a disjoint set; the selection depends on neither labels nor target outputs, so it cannot use the quantity ASR measures, but it is not a held-out calibration and we do not present it as one.

The feature term. Let $H^\star = H^{\ell^\star}(x)$ be the clean representation at the selected block, computed once with the unmodified surrogate, and $\tilde{H} = H^{\ell^\star}(x+\delta)$ the representation of the perturbed input. For two sets of coordinates u, v of equal shape we write

$$\rho(u; v) = \frac{\mathrm{mean}\big[(u-v)^2\big]}{\mathrm{mean}\big[v^2\big]}, \tag{2}$$

where both means are taken over the same coordinate set and ρ is computed separately for each image. It is not a metric; it expresses squared change as a fraction of the clean squared scale. MAAT applies it to all N tokens and to the $D-k$ channels outside $\mathcal{C}$, so both means in equation 2 run over $N \times (D-k)$ entries:

$$\mathcal{L}_{\mathrm{feat}} = \rho\Big(\tilde{H}_{:,\, c\notin\mathcal{C}};\ H^\star_{:,\, c\notin\mathcal{C}}\Big). \tag{3}$$

The second part compares the calibrated channels with their clean values at the token $t_c(x)$ of equation 1, where those channels peak:

$$\rho_{\mathcal{C}} = \rho\Big(\tilde{H}_{t_c(x),\, c};\ H^\star_{t_c(x),\, c}\Big)_{c\in\mathcal{C}}, \qquad \mathcal{L}^{\mathrm{full}}_{\mathrm{feat}} = \mathcal{L}_{\mathrm{feat}} - \rho_{\mathcal{C}}, \tag{4}$$

and the attack maximises $\mathcal{L}^{\mathrm{full}}_{\mathrm{feat}}$. All reported results use this form.

The two parts act differently on equation 2. In $\mathcal{L}_{\mathrm{feat}}$ the channels of $\mathcal{C}$ enter neither the numerator nor the denominator, so by construction the term measures change relative to the clean squared scale of the complement rather than of the full representation, which by Section 4 is the larger of the two. Subtracting $\rho_{\mathcal{C}}$ additionally penalises change at the calibrated coordinates themselves. Section 6 reports a control in which $\mathcal{C}$ is replaced by randomly chosen channels of the same size.

The task term. The task term is the surrogate's cross-entropy, $\mathcal{L}_{\mathrm{task}} = \mathrm{CE}\big(f(x+\delta), y\big)$. On ImageNet y is the ground-truth label. In the out-of-domain experiments the surrogate's 1000-class ImageNet head does not match the target label space, so y is the surrogate's own prediction on the clean image. This pseudo-label is produced by the surrogate alone and carries no information from the target model or from target-domain annotations.

Combining the two terms. The terms are combined at the gradient level. Because the update takes the sign of the accumulated gradient, the raw magnitude of either term does not determine its influence, and the ratio of the two gradient norms varies by orders of magnitude across images and steps. For each image i the two gradients are normalised separately before mixing:

$$g_i = (1-\lambda)\,\frac{\nabla_{\delta_i}\mathcal{L}_{\mathrm{task}}}{\|\nabla_{\delta_i}\mathcal{L}_{\mathrm{task}}\|_2} + \lambda\,\frac{\nabla_{\delta_i}\mathcal{L}^{\mathrm{full}}_{\mathrm{feat}}}{\|\nabla_{\delta_i}\mathcal{L}^{\mathrm{full}}_{\mathrm{feat}}\|_2}, \qquad \lambda = 0.5. \tag{5}$$

There is no averaging across the batch: g_i is the direction used for image i. A term whose gradient norm for that image falls below 10^{-8} contributes zero instead of being normalised, which matters because equation 4 is stationary at $\delta = 0$ whenever the base attack leaves the clean forward pass unchanged. Thus λ sets the relative contribution of the two normalised directions, not of their raw magnitudes.

Update and base attack. The update is MI-FGSM (Dong et al., 2018) applied to the mixed gradient: $m \leftarrow \mu m + g_i/\|g_i\|_1$ with $\mu = 1$, then $\delta \leftarrow \Pi_\varepsilon\big(\delta + \alpha\,\mathrm{sign}(m)\big)$ with $T = 10$ iterations, nominal step size $\alpha = \varepsilon/T$, and Π_ε projecting onto the ℓ_∞ ball of radius ε and the valid pixel range.

By default the base attack also applies the momentum token embedding of FPR (Ren et al., 2025), replacing each sub-layer output z_t by $\gamma z_t + \beta\,\mathrm{sg}(z_{t-1})$ with $\gamma = 0.8$ and $\beta = 0.3$, in blocks $0, \ldots, \ell^\star - 1$, so that the block whose output the feature term reads is not itself modified. The alternative, applying it to every block, is reported in Section 6; the choice between the two was fixed before those runs.

One consequence of this design should be stated. $H^\star$ is computed with the unmodified network while $\tilde{H}$ is computed with the modified forward pass, so equation 4 registers the representation shift introduced by the base attack in addition to the effect of δ. Restricting the modification to earlier blocks limits this shift but does not remove it. The matched control, in which the clean reference is taken through the same modified forward pass, has not been run.

Cost. Beyond the base attack, MAAT adds one clean forward pass per attack batch to obtain $H^\star$, and a one-time calibration over 500 images. The activation hook reads a representation the forward pass already computes. With $T = 10$ iterations, each of which performs a forward and a backward pass, the added cost is one forward pass per ten iterations.

6 Experiments

6.1 Setup

Models. We use two surrogates whose representations differ in their degree of activation concentration at the block read by the attack (Section 3.1): DINOv2-L/14 with its official ImageNet linear classifier, and CLIP-L/14 fine-tuned on ImageNet. The targets are the seven black-box ImageNet models of Appendix A: DINOv2-B/14 and -L/14 (linear classifiers on frozen backbones), CLIP-B/16, CLIP-B/32 and CLIP-L/14, and supervised ViT-B/16, -B/32 and -L/16. The corresponding surrogate is additionally reported as a white-box diagnostic and is excluded from transfer averages. For the out-of-domain experiments, we use the same backbones fine-tuned on E-commerce classification (nine classes) and FoodSeg103 semantic segmentation.

Data and budgets. On ImageNet, we attack 1000 validation images at $\varepsilon \in 4, 8, 16/255$ under ℓ_∞. On E-commerce we attack 1000 images and on FoodSeg103 500 images, both at $\varepsilon = 4/255$. All attacks use $T = 10$ steps with nominal step size ε/T, and the perturbation is projected onto the ℓ_∞ ball of radius ε after each step. Each target uses its own model-specific input preprocessing.

Metric. Attack success rate (ASR) is the fraction of images that the target handles correctly before the attack and incorrectly after it. For classification, correctness means that the top-1 prediction matches the ground-truth label. For segmentation, an image is considered correctly handled when its clean mIoU is at least 0.1; an attack succeeds when the mIoU falls below half of its clean value (Appendix D).

Controls and statistics. We distinguish two comparisons. First, *MAAT versus its base attack* measures whether the proposed objective improves transferability. Second, *MAAT versus a random-channel control* tests whether the particular channels identified by clean-data calibration contribute beyond using the same feature-term construction with an arbitrary set of channels. The control uses the same base attack, budget, layer, and objective, but replaces the calibrated set $\mathcal{C}$ by k channels sampled uniformly from the D available channels; the per-image peak token is then located using those channels. Calibration images are disjoint from the attack evaluation set. We use six independent random-channel seeds. The complete paired testing protocol, including per-target results and multiple-comparison correction, is given in Appendix E.

6.2 Main result

Table 1 reports ImageNet ASR at $\varepsilon = 16/255$. The first comparison asks whether adding MAAT improves a fixed base attack. Adding the feature term raises mean ASR over the seven transfer

Table 1: ASR (under "With MAAT" add the feature term of Section 5 to the MTE and PNA bases, respectively. Δ_{rand} is the median across six random-channel seeds of the gain of MTE,+,MAAT over its matched random-channel control, in percentage points. * marks targets whose statistical test survives Benjamini–Hochberg correction at FDR 0.05 across the seven transfer targets; the full testing protocol is in Appendix E. The PNA,+,MAAT column is a single run without control seeds. The white-box row is excluded from the transfer mean.

		Baselines			With MAAT		
	Target	MI-FGSM	PNA-PO	MTE	MTE,+	PNA,+	Δ_{rand}
CLIP-L/14	CLIP-B/16	59.1	74.1	68.4	74.7	**80.5**	$+7.05^*$
	CLIP-B/32	32.4	47.2	40.3	44.9	**48.2**	$+4.69^*$
	DINOv2-B/14	61.8	72.0	69.9	73.2	**79.5**	$+6.00^*$
	DINOv2-L/14	47.5	56.5	52.3	59.1	**64.2**	$+7.45^*$
	ViT-B/16	32.3	44.9	40.3	44.5	**45.7**	$+4.64^*$
	ViT-B/32	22.4	**31.9**	28.9	30.9	29.7	$+3.21^*$
	ViT-L/16	25.4	**35.3**	32.6	34.2	32.1	$+3.27^*$
	Mean	40.1	51.7	47.5	51.6	**54.3**	$+5.19$
	white-box	*99.7*	*97.3*	*99.5*	*98.8*	*99.9*	–
DINOv2-L/14	CLIP-B/16	48.3	62.3	64.3	**72.5**	62.9	$+4.96^*$
	CLIP-B/32	27.9	41.5	41.4	**46.1**	39.5	$+2.60$
	CLIP-L/14	62.7	73.0	78.9	**86.0**	82.5	$+4.85^*$
	DINOv2-B/14	91.4	93.3	96.8	98.0	**98.2**	$+0.82$
	ViT-B/16	33.7	45.4	46.7	**51.2**	42.0	$+3.31^*$
	ViT-B/32	24.2	34.3	34.3	**36.8**	31.7	$+1.49$
	ViT-L/16	26.2	35.7	36.6	**37.5**	30.4	$+0.48$
	Mean	44.9	55.1	57.0	**61.2**	55.3	$+2.64$
	white-box	*100.0*	*99.8*	*100.0*	*99.9*	*100.0*	–

targets from 57.0 to 61.2 for MTE with DINOv2-L/14 and from 50.2 to 54.3 for PNA with CLIP-L/14. Thus, the intervention improves transferability while remaining an objective-level modification rather than a standalone attack: its effect can be evaluated on top of different base attacks.

The second comparison asks whether the calibrated channel set itself matters. Against the random-channel control, the median gain is positive on all seven targets for both surrogates. For CLIP-L/14, the gain survives the Benjamini–Hochberg correction for all seven targets, with median improvements of $+3.2$ to $+7.5$ percentage points; every random-channel seed produces a positive gain on every target. The largest gains occur on the DINOv2 targets, showing transfer across pretraining families. For DINOv2-L/14, the gain is significant for three of seven targets and reaches $+5.0$ points on CLIP-B/16. Thus, the controlled evidence for the contribution of the calibrated coordinates is stronger for CLIP-L/14 than for DINOv2-L/14.

The strongest base attack depends on the surrogate: MTE gives higher mean ASR than PNA on DINOv2-L/14, whereas PNA gives higher mean ASR than MTE on CLIP-L/14. We therefore do not treat either base as intrinsically superior. For the main controlled analysis we use MTE+MAAT as the fixed configuration, chosen before the cross-target comparisons, while Section 6.5 tests whether the intervention transfers to other attack constructions. PNA-PatchOut (Wei et al., 2022) is likewise treated as a base attack rather than as a competing formulation of MAAT.

6.3 The budget at which the exclusion helps

Figure 2 repeats the controlled comparison at three perturbation budgets. At $\varepsilon = 4/255$, no target survives the multiple-comparison correction for either surrogate. At $8/255$, the gain survives for five targets with CLIP-L/14 and none with DINOv2-L/14; at $16/255$, it survives for seven and three targets, respectively. The mean gain is larger for CLIP-L/14 than for DINOv2-L/14 at all three budgets.

The onset of the controlled gain therefore depends on the surrogate and the perturbation budget. The two representation measurements in Section 3.1 show the same ordering of the surrogates: the

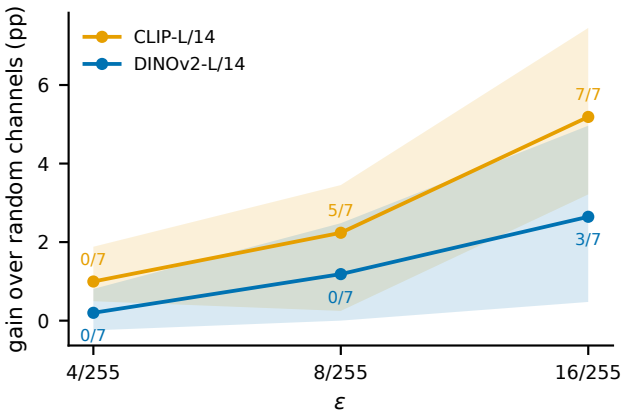


Figure 2: Gain of MTE,+,MAAT over matched random-channel controls as a function of the perturbation budget. Each point is the mean across the seven transfer targets of the per-target median across six random-channel seeds; shading spans the targets. Labels give the number of targets whose statistical test survives Benjamini–Hochberg correction at FDR 0.05. Per-budget results and seed-level statistics are in Appendix F.

high-norm token has a larger norm relative to a typical token (29.6 versus 8.6) and lower effective dimensionality of its squared magnitude (1.2 versus 6.0). With only two surrogates, this agreement is descriptive rather than evidence of a general relationship between activation concentration and transferability.

6.4 Transfer across task

We additionally test whether the intervention remains useful when the target task differs from ImageNet classification. On FoodSeg103 semantic segmentation, MAAT raises ASR over its corresponding base attack for every target with both surrogates. Relative to the random-channel control, three of the seven black-box targets survive the correction for DINOv2-L/14 and one for CLIP-L/14. Per-target results are reported in Appendix G.

6.5 MAAT on top of other attacks

Because MAAT changes the objective rather than the perturbation parameterization, it can be combined with base attacks that modify other parts of the attack procedure. Table 2 reports the change in mean ASR at $\varepsilon = 16/255$ for four such bases: MI-FGSM, which leaves the forward procedure unchanged; MTE, which modifies the forward pass; PNA, which changes the backward computation; and PatchOut, which masks the perturbation at the input. Adding MAAT increases mean ASR for all four bases on both surrogates, with gains ranging from $+0.84$ points for PatchOut on CLIP-L/14 to $+4.84$ points for MI-FGSM on CLIP-L/14.

The controlled MI-FGSM comparison further supports the intervention: all seven targets survive the correction for CLIP-L/14 and three for DINOv2-L/14. Detailed per-target results and controls for the remaining bases are reported in Appendix I.

The published PNA–PatchOut combination behaves differently: adding MAAT reduces mean ASR by 0.7 points on DINOv2-L/14 and 1.8 points on CLIP-L/14, and no target survives the random-channel comparison. We therefore treat this combination as an interaction case rather than use it to make a general claim about MAAT's compositionality. Its per-target results are given in Appendix I.

6.6 White-box success is not the objective

MAAT slightly reduces white-box ASR while increasing transfer ASR. With CLIP-L/14 at $\varepsilon = 4/255$, white-box ASR decreases from 93.1 for MI-FGSM to 87.2 with the MTE base and to 84.4

Table 2: Change in mean ASR (percentage points) over the seven ImageNet transfer targets when the feature term is added to a base attack, $\varepsilon = 16/255$. MAAT is evaluated on bases that modify different parts of the attack procedure. The PNA–PatchOut combination is reported separately in Appendix I.

Base attack	Modified component	DINOv2-L/14	CLIP-L/14
MI-FGSM	none	$+3.30$	$+4.84$
MTE	forward pass	$+4.16$	$+4.11$
PNA	backward pass	$+1.58$	$+4.07$
PatchOut	input	$+2.03$	$+0.84$

with MAAT, while mean transfer ASR increases from 16.8 to 20.1 and 20.6, respectively. At $\varepsilon = 16/255$, the corresponding white-box ASRs are 99.7, 99.5, and 98.8, while the ordering of transfer ASR remains unchanged.

This separation between white-box and transfer performance is consistent with the intended role of the feature objective: it can trade some surrogate-model success for transferability. Reporting both quantities therefore helps distinguish improved transfer from simply obtaining a stronger attack on the surrogate.

7 Conclusion

We studied whether concentrated activation structure in large Vision Transformers can be exploited to improve transferable adversarial attacks. At the representation layer used by the attack, we found that a high-norm token concentrates a large fraction of its magnitude in a small number of channels. Clean-data calibration reliably identifies these channels, with their maxima coinciding with the highest-norm token in 99.7–99.8% of image–channel pairs.

Based on this observation, we introduced MAAT, which treats the calibrated channels separately from their complement in its feature objective. Across eight targets, three datasets, and four base attacks, MAAT improves transfer performance over the corresponding base attacks. Matched random-channel controls further show that the channel choice contributes to the gain for CLIP-L/14.

Overall, our results show that the concentration structure of intermediate representations can be exploited to construct more transferable adversarial objectives, while also highlighting the importance of controlled comparisons when interpreting this effect.

AI use statement

In this work, we used generative AI tools for language editing and proofreading of the manuscript. All AI-assisted text was reviewed and verified by the authors. We take responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

Ethics statement

This work does not involve human subjects, personal data, or human participant studies. We are not aware of any ethical concerns, conflicts of interest, or other issues requiring additional disclosure.

Reproducibility statement

We provide detailed descriptions of the proposed method and experimental setup in Sections 3 and 4, with additional implementation details, supplementary results, and ablation studies in the Appendix. The code will be released publicly after publication to facilitate reproduction of the reported experiments.

REFERENCES

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International conference on learning representations*, volume 2024, pp. 2632–2652, 2024.

Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9185–9193. IEEE, 2018.

Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 4307–4316. IEEE, 2019.

Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Qian Huang, Isay Katsman, Horace He, Zeqi Gu, Serge Belongie, and Ser-Nam Lim. Enhancing adversarial example transferability with an intermediate level attack. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4733–4742, 2019.

Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. *arXiv preprint arXiv:1908.06281*, 2019.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Muzammal Naseer, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Fatih Porikli. A self-supervised approach for adversarial robustness. In *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 259–268. IEEE, 2020.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Yuchen Ren, Zhengyu Zhao, Chenhao Lin, Bo Yang, Lu Zhou, Zhe Liu, and Chao Shen. Improving adversarial transferability on vision transformers via forward propagation refinement. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25071–25080. IEEE, 2025.

Mingjie Sun, Xinlei Chen, J Zico Kolter, and Zhuang Liu. Massive activations in large language models, 2024.

Hugo Touvron, Matthieu Cord, and Hervé Jégou. Deit iii: Revenge of the vit. In *European conference on computer vision*, pp. 516–533. Springer, 2022.

Xiaosen Wang and Kun He. Enhancing the transferability of adversarial attacks through variance tuning. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1924–1933. IEEE, 2021.

Xiaosen Wang, Xuanran He, Jingdong Wang, and Kun He. Admix: Enhancing the transferability of adversarial attacks. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, pp. 16138–16147. IEEE, 2021.

Table 3: Peak block per model. ℓ is the 0-indexed block with the largest r, and depth is $\ell/(B-1)$. Measured on the same 300 ImageNet validation images. DINOv2 rows use the official ImageNet linear classifiers; CLIP and ViT rows are fine-tuned.

Model	Task	D	B	Peak ℓ	Depth	r
DINOv2-S/14	ImageNet	384	12	9	0.82	9.4
DINOv2-B/14	ImageNet	768	12	11	1.00	66.3
DINOv2-L/14	ImageNet	1024	24	18	0.78	71.6
CLIP-B/16	ImageNet	768	12	7	0.64	161.6
CLIP-B/32	ImageNet	768	12	11	1.00	35.7
CLIP-L/14	ImageNet	1024	24	12	0.52	350.9
ViT-B/16	ImageNet	768	12	5	0.45	59.9
ViT-B/32	ImageNet	768	12	8	0.73	136.3
ViT-L/16	ImageNet	1024	24	12	0.52	117.1
DINOv2-L/14	E-commerce	1024	24	18	0.78	71.8
DINOv2-L/14	FoodSeg103	1024	24	19	0.83	73.3
CLIP-L/14	E-commerce	1024	24	12	0.52	368.1
CLIP-L/14	FoodSeg103	1024	24	12	0.52	353.0
ViT-L/16	E-commerce	1024	24	12	0.52	120.3
ViT-L/16	FoodSeg103	1024	24	13	0.57	74.3

Zhipeng Wei, Jingjing Chen, Micah Goldblum, Zuxuan Wu, Tom Goldstein, and Yu-Gang Jiang. Towards transferable adversarial attacks on vision transformers. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 2668–2676, 2022.

Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. Improving transferability of adversarial examples with input diversity. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 2725–2734. IEEE, 2019.

A The Peak Block Across Models

Section 3 measures two quantities on the two surrogates at every block. Running the same per-block measurement on every model is expensive, so for the wider survey we use a coarser statistic that can be read from a single pass. For a block ℓ, let $s_c(x) = \max_n |H^\ell_{n,c}(x)|$ and $a_c = Q_{0.05}(\{s_c(x)\})$ as in Section 4, and define the *dominance ratio*

$$r^\ell = \frac{\max_c a_c}{\mathrm{median}_c\, a_c}.$$

This is a property of the aggregated channel profile rather than of any single activation vector, and we use it only to locate the block at which concentration peaks and to compare models of different widths; the claims in the main text rest on the direct measurements of Section 3.1, not on r.

Table 3 reports the peak block of fifteen models: DINOv2, CLIP and supervised ViT backbones with ImageNet classifiers, and three of those backbones additionally fine-tuned for e-commerce classification and for FoodSeg103 segmentation. All rows are measured on the same 300 ImageNet validation images; each model is read through its own preprocessing (Appendix B), so rows differ in more than weights. The values are point estimates from those images and we do not report an uncertainty on them.

Two observations. The peak block, expressed as a fraction $\ell/(B-1)$ of network depth, ranges from 0.45 to 1.00: two models peak in their final block, and models of equal width and depth disagree. A block chosen as a fixed fraction of depth would therefore miss the peak for most of these models. And for a given backbone, fine-tuning for a different task moves the peak by at most one block and changes r by a few per cent, with one exception, ViT-L/16 on segmentation, where r falls by roughly a third while the block moves by one.

Table 4: Position of the high-norm token on the 16×16 patch grid, over 500 ImageNet validation images. "Modal" is the most frequent single position and the share of images it accounts for; rows and columns are 0-indexed, so the grid centre is 7.5.

Surrogate	Block	On class token	Distinct positions	Modal	Mean (row, col)
DINOv2-L/14	19	0.0%	80 / 256	pos. 3, 15.6%	$(4.5,\ 7.2)$
CLIP-L/14	12	0.0%	156 / 256	pos. 46, 6.6%	$(6.2,\ 8.6)$

B Models, Preprocessing and Commands

Preprocessing. Each family is read through its own preprocessing and input normalisation, which we take from the checkpoint's own configuration rather than imposing a common one. Attacks operate in $[0, 1]$ pixel space, and each model applies its own normalisation as its first layer, so a perturbation bounded by ε in pixel space is the same perturbation for every target.

Attack settings. Unless stated otherwise: $T = 10$ iterations, nominal step ε/T, momentum $\mu = 1$, $\lambda = 0.5$, $k = 5$ calibrated channels, calibration quantile $q = 0.05$ over 500 clean images, and the forward-pass refinement of the base applied with $\gamma = 0.8$, $\beta = 0.3$ to blocks before $\ell^\star$. ImageNet runs attack 1000 validation images, FoodSeg103 runs 500.

C Where the High-Norm Token Sits

Table 4 summarises the spatial position of the high-norm token over 500 images at the block each surrogate is read from. The token is always a patch token: no peak falls on the class token for either model. Its position is not fixed, and it is not uniform either—on DINOv2-L/14 it occupies fewer distinct positions and leans towards the upper rows of the grid. We do not test this bias for significance and do not use it anywhere in the paper; preprocessing, cropping and the composition of the validation set are all plausible contributors.

D Segmentation Success Rules

An attack on a segmentation model has no single natural notion of success. We therefore use the following per-image criterion throughout the segmentation experiments. An image counts as successfully attacked if its clean per-image mIoU is at least 0.1 and the adversarial mIoU falls below half of the clean value. The floor excludes images on which the model already performs poorly, for which a further drop is less informative.

E Statistical protocol

For each target and control seed we apply a paired McNemar test to the images the target handles correctly in both runs, and report the median gain across seeds. Median-seed p-values are corrected with Benjamini–Hochberg at FDR 0.05 across the seven transfer targets of each surrogate, budget and dataset. The unit of analysis is the target: seeds replicate one hypothesis and are not counted as separate tests. As a consistency check we also report how many seeds give a positive gain, for which a sign test over six seeds can reach $p = 0.031$. For FoodSeg103, the same-backbone row is reported but kept out of the corrected family; the family is the seven black-box targets.

F The budget at which the exclusion helps: per-budget numbers

At $\varepsilon = 4/255$ no target survives the correction for either surrogate. The two surrogates part at $8/255$: on CLIP-L/14 the gain over the control survives on five targets, on DINOv2-L/14 on none. At $16/255$ the counts are seven and three. The mean gain on CLIP-L/14 is about twice that on DINOv2-L/14 at $8/255$ and $16/255$ (2.2 and 5.2 points against 1.2 and 2.6) and five times larger at

Algorithm 1 MAAT on top of a base attack

Require: surrogate f; clean batch x with labels y; budget ε; steps T; weight λ; channel count k; quantile q

once: $\ell^\star \leftarrow$ block selected by the rule of Section 3.2
once: $\mathcal{C} \leftarrow$ top-k channels of a_c at $\ell^\star$, quantile q over clean images
$H^\star \leftarrow H^{\ell^\star}(x)$ with the unmodified network; $\quad \delta \leftarrow 0, m \leftarrow 0$
for $t = 1, \dots, T$ **do** ▷ base modification active on blocks $< \ell^\star$
  $\tilde{H} \leftarrow H^{\ell^\star}(x+\delta), \quad$ logits $\leftarrow f(x+\delta)$
  $\mathcal{L}_{\text{task}} \leftarrow \text{CE}(\text{logits}, y)$
  $\mathcal{L}_{\text{feat}}^{\text{full}} \leftarrow \rho$ outside $\mathcal{C} - \rho$ on $\mathcal{C}$ at its peak token ▷ Eqs. 3, 4
  $g \leftarrow$ per-image normalised mixture of the two gradients ▷ Eq. 5
  $m \leftarrow \mu m + g/\|g\|_1$
  $\delta \leftarrow \Pi_\varepsilon\big(\delta + \alpha\,\text{sign}(m)\big)$
end for
return $x + \delta$

$4/255$ (1.0 against 0.2). Even at $4/255$, where nothing survives the correction, CLIP-L/14 gives a positive gain on all six seeds for six of the seven targets, whereas DINOv2-L/14 does so for one.

G FoodSeg103 per-target results

Table 5: FoodSeg103 semantic segmentation, $\varepsilon = 4/255$, with an ImageNet classifier as surrogate. ASR (%) with the MTE base and with MAAT added; Δ_{rand} as in Table 1, six seeds. *: survives Benjamini–Hochberg across the seven black-box targets. ‡: same backbone as the surrogate with a different head, shown in italics and excluded from the corrected family.

	Surrogate DINOv2-L/14			Surrogate CLIP-L/14		
Target	MTE	+ MAAT	Δ_{rand}	MTE	+ MAAT	Δ_{rand}
CLIP-B/16	11.5	15.7	$+1.58$	11.1	13.6	$+2.11$
CLIP-B/32	3.2	3.5	-0.23	5.0	5.3	$+0.34$
CLIP-L/14	18.4	22.5	$+1.96^*$	*36.9*	*40.0*	$+4.02^\ddagger$
DINOv2-B/14	41.5	49.6	$+4.10^*$	23.9	25.1	$+2.56^*$
DINOv2-L/14	*37.9*	*45.0*	$+4.18^\ddagger$	15.2	16.3	$+1.53$
ViT-B/16	9.1	12.1	$+1.83^*$	6.5	7.0	$+0.76$
ViT-B/32	4.6	4.7	$+0.47$	4.1	4.5	$+0.47$
ViT-L/16	2.2	2.5	$+0.42$	1.9	2.6	$+0.10$

H Attack Algorithm

Algorithm 1 summarizes the MAAT procedure on top of a base transfer attack. The base attack supplies the task term and update rule, while MAAT modifies only the feature term by excluding the calibrated channel set $\mathcal{C}$.

I MAAT on Top of PNA and PatchOut, per Target

Table 6 gives the per-target values for the two components of PNA-PatchOut, separately and together. MAAT adds to each component on its own on both surrogates, and subtracts from their combination; the main text discusses this in Section 6.

Table 6: ASR (%) on ImageNet at $\varepsilon = 16/255$ with and without MAAT, for PNA alone, PatchOut alone, and the two applied together. Single runs, no control seeds. Δ is the change from adding MAAT; the last row is the mean over the seven transfer targets. The white-box row is in italics and excluded from the mean.

	PNA			PatchOut			PNA + PatchOut		
Target	base	+MAAT	Δ	base	+MAAT	Δ	base	+MAAT	Δ
Surrogate DINOv2-L/14									
CLIP-B/16	60.8	62.9	+2.1	52.1	53.9	+1.8	62.3	62.9	+0.6
CLIP-B/32	38.3	39.5	+1.2	31.1	33.7	+2.6	41.5	40.9	−0.6
CLIP-L/14	74.1	82.5	+8.4	65.7	71.3	+5.6	73.0	73.3	+0.3
DINOv2-B/14	97.3	98.2	+0.9	91.1	93.4	+2.3	93.3	93.5	+0.2
ViT-B/16	41.0	42.0	+1.0	37.8	38.8	+1.0	45.4	43.1	−2.3
ViT-B/32	32.0	31.7	−0.3	26.0	26.4	+0.4	34.3	33.3	−1.0
ViT-L/16	32.7	30.4	−2.3	29.5	30.1	+0.6	35.7	33.9	−1.8
Mean	53.7	55.3	+1.6	47.6	49.7	+2.0	55.1	54.4	−0.7
white-box	*100.0*	*100.0*	–	*99.9*	*99.8*	–	*99.8*	*99.7*	–
Surrogate CLIP-L/14									
CLIP-B/16	76.5	80.5	+4.0	63.9	64.3	+0.4	74.1	70.4	−3.7
CLIP-B/32	43.8	48.2	+4.4	36.7	38.5	+1.8	47.2	45.3	−1.9
DINOv2-B/14	73.2	79.5	+6.3	66.7	67.4	+0.7	72.0	70.4	−1.6
DINOv2-L/14	55.4	64.2	+8.8	52.9	54.4	+1.5	56.5	53.9	−2.6
ViT-B/16	42.2	45.7	+3.5	38.8	38.4	−0.4	44.9	43.3	−1.6
ViT-B/32	29.8	29.7	−0.1	27.0	28.4	+1.5	31.9	30.8	−1.1
ViT-L/16	30.6	32.1	+1.5	29.9	30.4	+0.5	35.3	35.0	−0.3
Mean	50.2	54.3	+4.1	45.1	46.0	+0.8	51.7	49.9	−1.8
white-box	*100.0*	*99.9*	–	*95.8*	*95.0*	–	*97.3*	*95.7*	–