

BACKDOOR VECTORS: A TASK ARITHMETIC VIEW ON BACKDOOR ATTACKS AND DEFENSES IN MODEL MERGING

Anonymous authors

Paper under double-blind review

ABSTRACT

Model merging (MM) recently emerged as an effective method for combining large deep learning models. However, it poses significant security risks. Recent research shows that it is highly susceptible to backdoor attacks, which introduce a hidden trigger into a single fine-tuned model instance that allows the adversary to control the output of the final merged model at inference time. In this work, we propose a simple framework for understanding backdoor attacks by treating the attack itself as a task vector. *Backdoor Vector (BV)* is calculated as the difference between the weights of a fine-tuned backdoored model and fine-tuned clean model. BVs reveal new insights into attacks understanding and a more effective framework to measure their similarity and transferability. Furthermore, we propose a novel method that enhances backdoor resilience through merging dubbed *Sparse Backdoor Vector (SBV)* that combines multiple attacks into a single one. We identify the core vulnerability behind backdoor threats in MM: *inherent triggers* that exploit adversarial weaknesses in the base model. To counter this, we propose *Injection BV Subtraction (IBVS)* – an assumption-free defense against backdoors in MM. Our results show that SBVs surpass prior attacks and is the first method to leverage merging to improve backdoor effectiveness. At the same time, IBVS provides a lightweight, general defense that remains effective even when the backdoor threat is entirely unknown.

1 INTRODUCTION

Model Merging (MM) is an effective and cost-efficient paradigm for updating large pretrained models via weight-space operations Wortsman et al. (2022); Ilharco et al. (2023a); Yadav et al. (2023); Gargiulo et al. (2025); Marczak et al. (2025). It enables the integration of differently fine-tuned models into a single, more capable one. However, research into the security risks of this increasingly popular approach remains limited Yang et al.; Zhang et al. (2024a).

Backdoor attacks have emerged as a major security concern in recent literature Gu et al. (2017); Li et al. (2021); Liang et al. (2024); Qi et al. (2022); Nguyen & Tran (2020); Zhang et al. (2024b); Abad et al. (2024). They are a class of adversarial techniques that implant malicious, hidden behavior into machine learning models by poisoning training data or manipulating the training procedure. It is hard to detect a model compromised in this manner because the model performs as expected on clean inputs, but produces attacker-specified outputs only when a particular trigger is present. Backdoors pose serious risks, especially when models are trained or fine-tuned with data or checkpoints from untrusted sources. Successful attacks can bypass authentication Guo et al. (2021), allow harmful content to evade moderation filters Zhang et al. (2024a) or expose private user data during downstream deployment Guo et al. (2025).

In this work, we study backdoor attacks in the context of model merging. We introduce *Backdoor Vectors (BVs)* – task vectors that capture the information of specific backdoor attacks. The backdoor vector is computed by subtracting the weights of a clean fine-tuned model from those of a backdoored fine-tuned model trained on the same task (see Figure 1a, Section 3).

We show that modeling backdoor attacks as task vectors using Task Arithmetic (TA) Ilharco et al. (2023b) offers a simple and effective framework to analyze and quantify their behavior. Addition

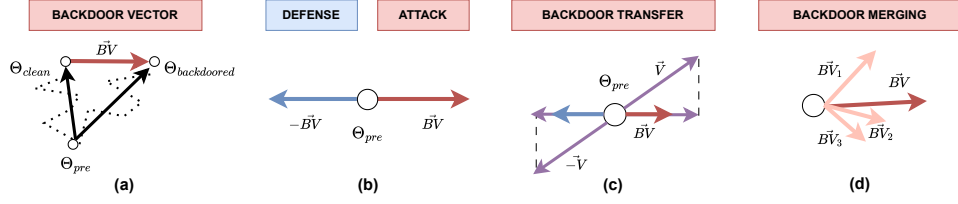


Figure 1: **Backdoor Attack as Task Vector = Backdoor Vector (BV)**. (a) A BV is the element-wise difference between backdoored and clean fine-tuned model parameters. (b) Adding a BV injects a backdoor; subtracting it weakens the attack. (c) Like task analogies, backdoor analogies reveal relationships between attacks. We define *Backdoor Transfer* as positive when Attack Success Rate (ASR) is strengthened and negative when is weakened by another task vector. (d) We show a method of merging BVs that yields a significantly stronger attack.

and negation map naturally to attack and defense, while BV analysis enables comparison and reveals transfer dynamics during MM. We introduce *Sparse Backdoor Vectors (SBVs)*, which merge multiple BVs into a more resilient and potent attack, yielding higher Attack Success Rate (ASR) after merging.

Finally, we identify the root of the backdoor threat in model merging as the white-box access to the pre-trained foundational model, an inherent requirement in all model merging scenarios. Although such access enables seamless collaboration, it also reveals adversarial vulnerabilities within the model, known as *inherent triggers* Tao et al. (2024); Wenger et al. (2022); Tao et al. (2022). We are the first to show that *inherent triggers* have high resilience in the MM process, but also share *inherent* similarities. Building upon this observation, we propose *Injection BV Subtraction (IBVS)* defense method to mitigate backdoor risks for MM. We summarize the contributions of this work as follows:

- We introduce *Backdoor Vectors (BVs)* and show that viewing backdoor attacks through task arithmetic reveals new insights, such as an intuitive understanding of the interplay between backdoor attacks and defenses, enabling a more effective framework for measuring their similarity, transferability, and resilience in model merging.
- We explore backdoor attacks as task vectors and show that merging multiple Backdoor Vectors produces a single, stronger and more resilient *Sparse Backdoor Vector (SBV)*. Our simple BV merging strategies outperform state-of-the-art attacks in effectiveness and in merging resilience.
- We identify the core vulnerability enabling backdoor threats in model merging as the inherent white-box access to the pre-trained foundational model. To address this, we propose *Injection BV Subtraction* – a defense method to mitigate backdoor risks even when the backdoor threat is entirely unknown.

2 PROBLEM SETTING

Model Merging We follow the standard procedure of merging vision models Yadav et al. (2023); Ilharco et al. (2022); Tang et al. (2023); Ortiz-Jimenez et al. (2024); Yang et al. (2024; 2023). We use CLIP model Radford et al. (2021) $M = \{V, T\}$ with an image encoder V and a text encoder T . CLIP aligns image and text embeddings in a joint space enabling zero-shot classification by comparing an image to textual class descriptions via cosine similarity. The predicted label corresponds to the class with the highest similarity score. To improve performance on specific tasks, a common approach is to fine-tune only the vision encoder V while freezing the text encoder T (see Appendix A.2 for details).

Let M_θ denote a CLIP-like model parameterized by weights θ , and let V_θ represent its visual encoder. We denote the weights of the base pre-trained model M_{pre} by θ_{pre} , and the weights of a model fine-tuned on dataset $\mathcal{D}^{(t)}$ from task t as $\theta^{(t)}$. The corresponding *task vector* is defined as the element-wise difference between the fine-tuned and pre-trained weights:

$$\Delta\theta^{(t)} = \theta^{(t)} - \theta_{\text{pre}}. \quad (1)$$

Suppose that we are given n such task vectors $\{\Delta\theta^{(1)}, \dots, \Delta\theta^{(n)}\}$, obtained from different fine-tuning instances (potentially across different tasks or training configurations). The model merging process aims to construct a new set of weights θ_{merged} by combining these task vectors into a merged

task vector $\Delta\theta_{\text{merged}}$, which is then added to the pre-trained weights:

$$\theta_{\text{merged}} = \theta_{\text{pre}} + \lambda\Delta\theta_{\text{merged}}. \quad (2)$$

where λ is a scaling factor determined on a held-out validation set. The exact strategy for computing $\Delta\theta_{\text{merged}}$ may vary, including simple averaging (Wortsman et al., 2022), heuristics aiming at reducing interference between models Yadav et al. (2023); Yu et al. (2024); Marczak et al. (2024); Wang et al. (2024a; 2025), or more sophisticated optimization-based approaches Yang et al. (2024; 2023).

We consider two distinct merging scenarios: *single-task* and *multi-task*. In the former, all merged models are fine-tuned on the same downstream task, differing only in training seeds, augmentations, or other hyperparameters. This setup is commonly used for robustness or performance improvement on a single objective, as shown in Model Soups Wortsman et al. (2022). *Multi-task* scenario follows the setting from learning via addition from Task Arithmetic Ilharco et al. (2023a): given models fine-tuned on different tasks, we aim to fuse them to obtain a model capable of multi-task generalization. In this work, we focus primarily on the *single-task* merging scenario, since it is harder to create a resilient backdoor in this setup. We aggregate multiple independently fine-tuned models $M^{(t)}$, for $t = 1, \dots, n$, all originating from a common pre-trained model M_{pre} . Although our emphasis is on the *single-task* case, we note that similar attack strategies may seamlessly extend to the *multi-task* setting.

Threat Model in Model Merging We adopt the threat model from Zhang et al. (2024a) and extend it to include the defender’s perspective.

Attack scenario. We consider a threat model in which the adversary acts as a model provider. The adversary publicly releases a $M_{\text{backdoored}}$ (fine-tuned for a task denoted as *adversary task*) with competitive utility to increase its chances of inclusion in a merged model. When $M_{\text{backdoored}}$ is incorporated into the model merging process, M_{merged} behaves according to the adversary’s intent, which can lead to major security breaches, as discussed in Section 1. We assume that only one of the models originates from the adversary, while the rest are contributed by benign providers. We refer to any task other than the adversary’s as a *clean task*.

Adversary’s Knowledge. The adversary has a dataset $\mathcal{D}_{\text{adv}}$ corresponding to a single task. Similarly to clean model contributors, adversary has white-box access to M_{pre} , since they fine-tune a base model M_{pre} to obtain $M_{\text{backdoored}}$. We assume that the adversary contributes only one model and has no information about other models, merging algorithms, or coefficients used in the merging process.

Defense scenario. The defender’s objective is to construct a reliable and secure merged model from independently submitted components, ensuring high utility on the target task while preventing the inclusion of any malicious behavior. Given that individual models may originate from untrusted sources, the defender aims to preserve the functional performance of the merged model on *clean tasks*, while mitigating the risk of backdoor activation.

Defender’s Knowledge. We assume that the defender lacks prior knowledge regarding whether any of the models have been backdoored. Furthermore, the defender has no access to the internal training data or methodologies employed by the individual contributors. Although the merging algorithm and its associated coefficients are under the control of the defender, the presence and nature of any injected trigger patterns remain unknown. Consequently, the defender must rely exclusively on the behavior of the submitted models to maintain the security and reliability of the merged model.

2.1 BACKDOOR ATTACK

Let x denote a clean input image and define the trigger as $t = \{m, \delta\}$, where $m \in \{0, 1\}^{H \times W}$ is a binary mask specifying the location of the trigger, and $\delta \in \mathbb{R}^{H \times W \times C}$ encodes the trigger pattern. The poisoned image is constructed by an injection function $x \circledast t$, defined as:

$$x \circledast t = \delta \odot m + (1 - m) \odot x, \quad (3)$$

where $\odot$ denotes element-wise (pixel-wise) multiplication. The objective of the backdoor attack is to train a model so that x is classified correctly, that is, $f(x) = y$, but $f(x \circledast t) = c$, where c is an attacker-specified target class. In this work, we call t *inherent* ($\star$) if the trigger patch was optimized to backdoor M_{pre} (e.g., by adversarial attack using white-box access to θ_{pre}) and *injected* ($\oplus$) if the trigger is a fixed pattern injected only to $M_{\text{backdoored}}$.

Table 1: **Sparsity $H(x)$ of TVs, BVs, and SBVs:** $x \in (0, 1)$; values near 1 indicate higher sparsity.

		CIFAR100	ImageNet100	TinyImageNet100
	TV	0.3270	0.3190	0.3234
	BV	0.3330	0.3261	0.3273
BadMerging	SBV (Ours)	0.5482	0.5407	0.5455
	BV	0.3296	0.3238	0.3247
BadNets	SBV (Ours)	0.5256	0.5262	0.5247

Algorithm 1 Sparse Backdoor Vector (SBV)

Require: Backdoored task vector $\Delta_{\text{backdoored}}$, clean task vectors $\{\Delta_{\text{clean}}^{(t)}\}_{t=1}^k$, sparsification type $st \in \{\text{SC}, \text{RND}\}$

- 1: **for** each $t = 1$ to k **do**
- 2: $BV^{(t)} \leftarrow \Delta_{\text{backdoored}} - \Delta_{\text{clean}}^{(t)}$
- 3: **end for**
- 4: $s \leftarrow \sum_{t=1}^k BV^{(t)}$
- 5: $\mu \leftarrow \text{sparse_mask}(\{BV^{(t)}\}, st)$
- 6: $SBV \leftarrow s_j \odot \mu_j$
- 7: **return** SBV

Table 2: **TVs Pairwise Cosine Similarity.**

Task	CIFAR100	ImageNet100	TinyImageNet100
TV	0.5646	0.5421	0.5438
BV (BadMerging)	0.5986	0.7238	0.6683
BV (BadNets)	0.3859	0.4671	0.4555
BV (BadMerging_BadNets)	0.5026	0.6653	0.5747

Algorithm 2 sparse_mask Function

Require: Backdoor vectors $\{BV^{(t)}\}_{t=1}^k$, type $st \in \{\text{SC}, \text{RND}\}$

Ensure: Mask $\mu \in \{0, 1\}^d$

- 1: $S \leftarrow \text{sign}(\{BV^{(t)}\})$
- 2: $s_0 \leftarrow S[1]$
- 3: $c \leftarrow \text{all}(S == s_0, \text{dim} = 0)$
- 4: $nz \leftarrow \text{all}(S \neq 0, \text{dim} = 0)$
- 5: $\mu \leftarrow c \wedge nz$
- 6: **if** $st = \text{RND}$ **then**
- 7: $\mu \leftarrow \text{shuffle}(\mu)$
- 8: **end if**
- 9: **return** μ

3 BACKDOOR VECTORS

Intuition. Task vectors Ilharco et al. (2023a) provide a simple way of thinking about modifying the capabilities of a model: adding a task vector improves the performance on the corresponding task while subtracting it enables unlearning of the task. Task vectors are also effective in modifying the properties of models that are not usually considered a task, e.g. the toxicity of text generation Ilharco et al. (2023a). Therefore, we treat model vulnerabilities as any other task and propose to look at backdoor attacks as *backdoor vectors*.

Definitions. Following Section 2, let θ_{pre} be the weights of a pre-trained model and $\theta^{(t)}$ the weights after fine-tuning on task t . The model fine-tuned by a benign provider is optimized solely to improve the performance on task t , resulting in weights $\theta_{\text{clean}}^{(t)}$. The adversary aims to produce a model that performs well on the task t but also contains certain vulnerabilities (i.e. backdoor) resulting in weights $\theta_{\text{backdoored}}^{(t)}$. The backdoor vector $BV^{(t)}$ is a parameter-wise difference between the backdoored weights $\theta_{\text{backdoored}}^{(t)}$ and the clean weights $\theta_{\text{clean}}^{(t)}$, i.e. $BV^{(t)} = \theta_{\text{backdoored}}^{(t)} - \theta_{\text{clean}}^{(t)}$. We use a scaling coefficient λ_{BV} to modulate the strength of an attack (or a defense). We denote BVs from backdoor attacks with $\oplus$ and $\star$ triggers as $BV_{\oplus}$ and $BV_{\star}$, respectively.

Backdoor Transfer determines how another task or backdoor vectors V interact with the primary backdoor vector (Figure 1c). Positive transfer occurs when V strengthens the primary backdoor, while negative transfer weakens it and can be used for defense. Notably, this interaction may not be symmetrical: one attack vector can strengthen another, but the reverse might not occur (see Figure 9 in the Appendix).

3.1 BV IMPROVEMENT BY MERGING: SPARSE BACKDOOR VECTORS (SBV)

Merging backdoored task with clean tasks can introduce negative backdoor transfer, weakening the attack. To counter this, we introduce BV merging method (Figure 1d) combining multiple backdoor attacks into a stronger one that can withstand model merging. We propose *sparsifying* BVs to retain only the most consistent and influential malicious components. This enhances the resulting sparse BV (SBV), allowing it to reinforce the trigger and persist despite dilution from clean models.

Algorithms 1 and 2 detail our sparsification procedure. We adapt a simple idea used in MM to reduce TV interference for backdoors and propose sign-consistent sparsification (SBV_{SC}), which retains components with consistent weights signs across merged BVs.

3.2 BV USED FOR DEFENSE: INJECTION BV SUBTRACTION (IBVS)

We show that state-of-the-art backdoor attacks on MM rely on *inherent triggers* that produce highly similar and aligned $BV_{\star}$ vectors, enabling strong cross-attack transfer and revealing shared, generalizable structure of backdoors (see Section 4.4). To exploit this, we propose *Injection BV Subtraction (IBVS)* – a defense that subtracts a fixed $BV_{\oplus}$ (e.g., BV created from a simple white square trigger; see Figure 6) to suppress unknown $\star$ attacks. **IBVS** requires no knowledge of the adversary’s dataset, labels, target class, or trigger. The defender only needs to train a fixed $\oplus$ trigger on any dataset, compute $BV_{\oplus}$, and subtract it – using task arithmetic as a simple yet effective tool to mitigate backdoor influence.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

We follow the common model merging experimental setup for backdoor attacks introduced by Zhang et al. (2024a). We present more detailed information about the details in the Section B and present additional results in the Section C, including results for multi-task setup, different architectures (ConvNext and ViT-L-14) and different merging types.

Metrics. *Clean Accuracy (CA)* is the accuracy on clean test data obtained by a clean merged model (all provided task vectors are clean). *Backdoored Accuracy (BA)* is the accuracy on clean test data obtained by a backdoored merged model (one provided task vector – adversary task – is backdoored). *Attack Success Rate (ASR)* is a fraction of triggered test images from the adversary task that are predicted as the target class by M_{merged} . An attack is considered successful when the ASR is high and the backdoored model performs similarly well as clean model ($BA \approx CA$). We use *Hoyer sparsity* Hoyer (2004) to compare the sparsity of TV s, BV s and SBV s.

Datasets. We conduct our experiments on the following datasets. We use CIFAR100 Krizhevsky & Hinton (2009) and ImageNet100 Deng et al. (2009) as adversary tasks. In single-task setup, we merge one adversary task with the rest clean using the same dataset. In multi-task setup, following Zhang et al. (2024a), we merge adversary task with five clean tasks: Cars Krause et al. (2013), SUN397 Xiao et al. (2010), EuroSAT Helber et al. (2019), GTSRB Stallkamp et al. (2011) and Pets Parkhi et al. (2012).

MM. We use TA Ilharco et al. (2023a) as MM algorithm. We use ViT-B-32, ViT-L-14 and ConvNext as visual encoders of CLIP models, the first being the default for the experiments in the main part of our work. For BV merging we use weights averaging as baseline Wortsman et al. (2022) for our proposed SBV_{SC} method.

Backdoor attacks. We use the simplest classic BadNets Gu et al. (2017) attack as a representative of fixed injected ($\oplus$) triggers as well as current state-of-the-art attack on MM (using inherent $\star$ triggers): BadMerging Zhang et al. (2024a). We compare these attack types using BVs, assess their transferability in MM, and show that our sparsification method substantially improves trigger resilience and final ASR in both cases. We set the trigger size to be 1% of pixels in the image.

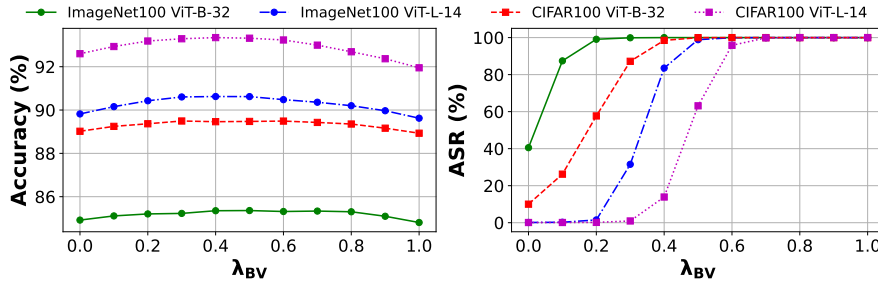


Figure 2: $\theta_{\text{clean}} + \lambda BV = \theta_{\text{backdoored}}$. Increasing the λ of BV added to θ_{clean} preserves accuracy (left) but sharply raises attack success rate (right).

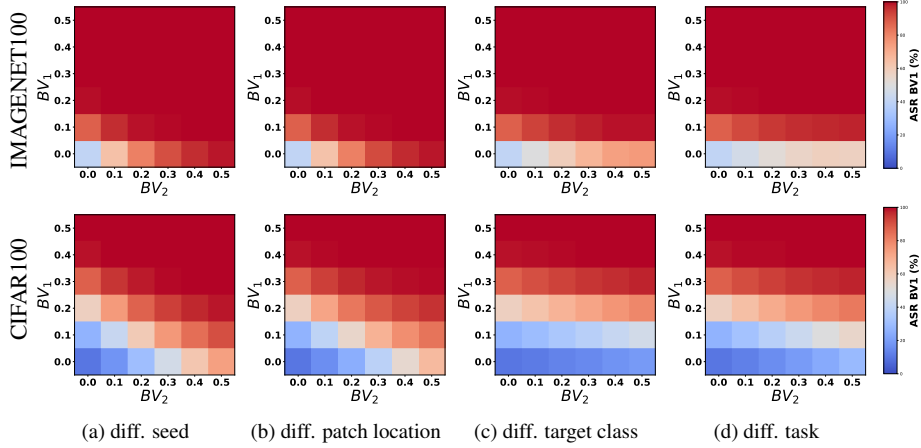


Figure 3: **ViT-B-32: Positive Backdoor Transfer Across Attacks ($BV_2 \rightarrow BV_1$).** Adding BV_2 increases the ASR of BV_1 . Axes show λ_{BV_1} , λ_{BV_2} ; (a–c) use BVs from the same task (ImageNet100 or CIFAR100). Strong transfer occurs across seeds (a) and patch locations (b), but is weaker across target classes (c) and tasks (d).

Table 3: **ViT-B-32: Backdoor Merging: Single-Task Attack Results.** We merge $1 \times \theta_{\text{backdoored}} + 9 \times \theta_{\text{clean}}$ with $\lambda = 0.1$, reporting mean $\pm$ std. dev. **BV merging:** *AVG* — weight averaging **Methods:** $\text{BN}(\oplus)$ — BadNets, $\text{BM}(\star)$ — BadMerging. ViT-B-32.

Backdoor Attack			Adversary task: CIFAR100			Adversary task: IMAGENET100		
Setting	Method	BV merging	CA	BA	ASR	CA	BA	ASR
Single-task	$\text{BN}(\oplus)$	-	89.70 $\pm$ 0.03	89.73 $\pm$ 0.03	0.2 $\pm$ 0.12	85.88 $\pm$ 0.07	85.88 $\pm$ 0.04	0.26 $\pm$ 0.1
	$\text{BN}(\oplus)$	AVG	89.70 $\pm$ 0.03	89.75 $\pm$ 0.03	0.21 $\pm$ 0.12	85.88 $\pm$ 0.07	86.0 $\pm$ 0.03	0.27 $\pm$ 0.1
	$\text{BN}(\oplus)$	SBV_{RDM} (Ours)	89.70 $\pm$ 0.03	89.72 $\pm$ 0.04	0.23 $\pm$ 0.12	85.88 $\pm$ 0.07	86.0 $\pm$ 0.06	0.33 $\pm$ 0.14
	$\text{BN}(\oplus)$	SBV_{SC} (Ours)	89.70 $\pm$ 0.03	89.73 $\pm$ 0.12	12.23 $\pm$ 10.52	85.88 $\pm$ 0.07	85.8 $\pm$ 0.06	29.45 $\pm$ 12.36
	$\text{BM}(\star)$	-	89.70 $\pm$ 0.03	89.74 $\pm$ 0.04	27.29 $\pm$ 12.14	85.88 $\pm$ 0.07	85.91 $\pm$ 0.04	84.01 $\pm$ 7.83
	$\text{BM}(\star)$	AVG	89.70 $\pm$ 0.03	89.72 $\pm$ 0.03	27.56 $\pm$ 12.17	85.88 $\pm$ 0.07	85.96 $\pm$ 0.02	84.79 $\pm$ 7.48
	$\text{BM}(\star)$	SBV_{RDM} (Ours)	89.70 $\pm$ 0.03	89.72 $\pm$ 0.01	35.93 $\pm$ 13.64	85.88 $\pm$ 0.07	85.97 $\pm$ 0.04	92.55 $\pm$ 3.83
	$\text{BM}(\star)$	SBV_{SC} (Ours)	89.70 $\pm$ 0.03	89.69 $\pm$ 0.06	97.21 $\pm$ 2.3	85.88 $\pm$ 0.07	85.83 $\pm$ 0.09	99.99 $\pm$ 0.01

4.2 KEY OBSERVATIONS

Our four key observations align with the concepts illustrated in Figure 1:

BV addition represents backdoor attack. ASR grows sharply with increasing λ_{BV} (see Figure 2), confirming that the effectiveness of the backdoor attack improves as the scaling factor increases.

BVs have high transferrability. Adding a BV from another attack notably boosts ASR of primary attack, as shown in Figure 3. Interestingly, backdoor transfer remains strong across trigger patches with different seeds or locations (Figure 3a,b), suggesting triggers are not fixed to one position and can be reinforced by varying or duplicating placement. Positive transfer across tasks or target classes also occurs, but at a lower magnitude.

BV merging: proposed SBV enhances attack by merging multiple attacks. Figures 4 and 5 show ACC/ASR trajectories throughout MM, with the optimal point in the top-right. Naive merging (AVG) offers little gain over a single BV. *SBV* uses sign consistency to produce more sparse BV (see Table 1) that better preserve backdoor information through merging. In 7 of 8 cases, *SBV_{SC}* performs best, highlighting that random sparsification (*SBV_{RND}*) is insufficient – sign consistency is the key to maximizing effectiveness. Figure 4 shows that *SBV* significantly improves backdoor resilience in single-task MM, surpassing the state-of-the-art. We are the first to show a *non-inherent* backdoor attack that withstands the merging process (Figure 5). It further highlights the importance of our backdoor merging method, showing that merging can significantly strengthen even simplest backdoor attack types. As shown in Table 3, our method outperforms both weight-averaging and no-merging approaches.

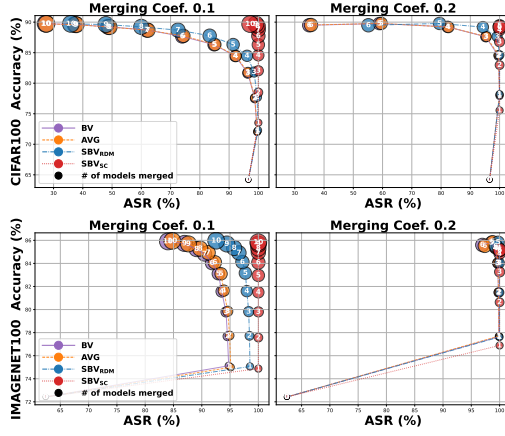


Figure 4: ViT-B-32: ACC/ASR trajectories for inherent (★) triggers in first-backdoored-rest-clean single-task MM.

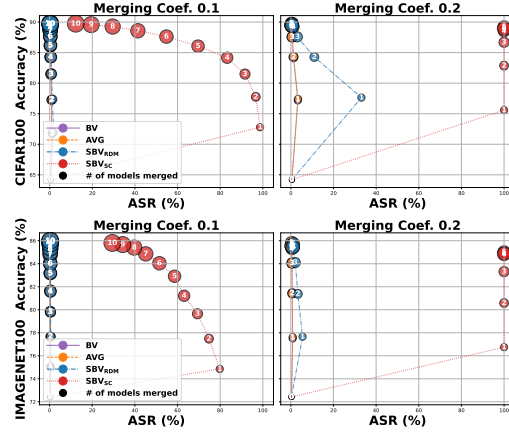


Figure 5: ViT-B-32: ACC/ASR trajectories for injected (⊕) triggers in first-backdoored-rest-clean single-task MM.

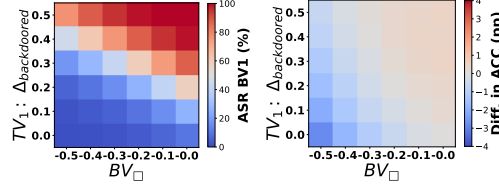


Figure 6: ViT-B-32: Injection BV Subtraction (IBVS) Defense: subtracting the BV of a white-square (□) trigger. *Left*: Removing $BV_{\square}$ lowers the ASR of an unknown attack (BV_1), highlighting cross-type transferability (see Figure 9). *Right*: This ASR drop slightly reduces clean accuracy (vs. (0.0, 0.0)) but can be mitigated by natural addition of clean task vectors in MM process.

BV subtraction as backdoor defense. Figure 6 shows how defenders can leverage backdoor transfer to improve MM robustness. Our proposed IBVS reduces ASR with minimal accuracy loss (Table 4) requiring only a fixed $\oplus$ trigger trained on any dataset to compute and subtract $BV_{\oplus}$ during merging.

4.3 ADDITIONAL INSIGHTS

BV addition does not degrade the accuracy. The results in Figure 2 illustrate the effects of increasing the scaling coefficient of the BV (λ_{BV}) on the attack strength. The accuracy remains stable as the λ_{BV} increases, indicating that BV does not degrade the clean performance of the model.

Reducing ASR via $BV_{\square}$ Subtraction. Subtracting $BV_{\square}$, derived from an attack with a fixed white square trigger, lowers the ASR of unknown ★ attacks with minimal accuracy loss. This forms the basis of our IBVS defense, which builds a fixed $\oplus$ -trigger BV and subtracts it during model merging to weaken state-of-the-art MM attacks (see Table 4). For more in-depth experiments on IBVS performance see Section C.1.2 in the appendix.

Table 4: ViT-B-32 – Backdoor Defense: Injected BV Subtraction (IBVS). Single-task defense results for merging $1 \times \theta_{\text{backdoored}} + 9 \times \theta_{\text{clean}}$ with $\lambda = 0.1$, reporting mean BA and ASR. BV merging: AVG — weight averaging. IBVS: $IBVS_{\square}$ uses a fixed white-square trigger; $IBVS_{BN}$ uses a wavelet trigger from BadNets. Backbone: ViT-B-32. No defense: Badmerging attack.

Setting	Adversary task: CIFAR100								Adversary task: ImageNet100							
	BV		AVG		SBV_{RND}		SBV_{SC}		BV		AVG		SBV_{RND}		SBV_{SC}	
	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR
No defense	89.74	27.29	89.72	27.56	89.72	35.93	89.69	97.21	85.91	84.01	85.96	84.79	85.97	92.55	85.83	99.99
$IBVS_{\square}$ (Ours)	89.58	20.65	89.36	23.59	89.41	30.81	89.61	94.69	85.53	66.83	85.13	81.22	85.15	89.74	85.47	99.83
$IBVS_{BN}$ (Ours)	89.57	16.56	89.41	19.36	89.42	25.37	89.46	89.21	85.61	65.26	85.14	80.89	85.14	90.17	85.59	99.89

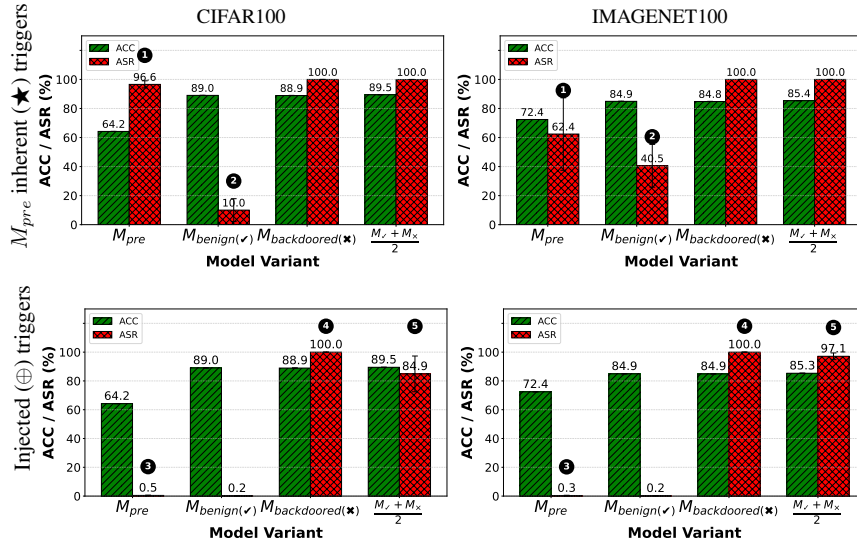


Figure 7: **Comparison of inherent (★) and injected (⊕) backdoor triggers in model merging.** State-of-the-art attacks on MM use ★ triggers – adversarial vulnerabilities of the base model M_{pre} . They ① directly affect M_{pre} and are effective even before merging with $M_{backdoored}$ and ② remain effective after merging with clean (benign) model. In contrast, ⊕ triggers, used in classical backdoor attacks, rely on fixed patches, ③, ④ require merging with $M_{backdoored}$ to become effective, and ⑤ degrade faster after merging with clean models.

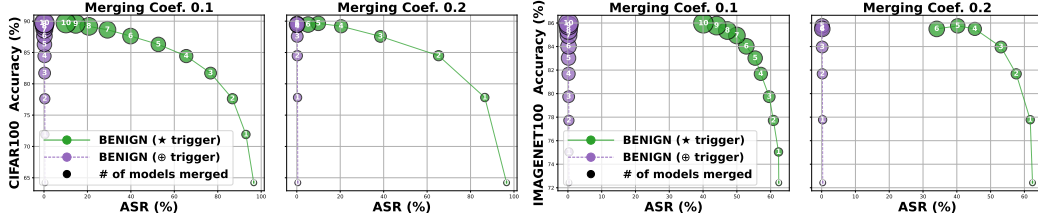


Figure 8: **ACC/ASR trajectories for inherent (★) and injected (⊕) triggers in all-clean single-task MM.** We demonstrate that ★ triggers exploit the adversarial vulnerabilities of the base model M_{pre} , maintaining their effectiveness even after merging multiple clean (benign) models. Conversely, ⊕ triggers are completely ineffective in the absence of $M_{backdoored}$ model.

Backdoor transfer is asymmetric. Adding $BV_{\oplus}$ with small $\lambda_{BV_{\oplus}}$ increases the ASR for $BV_{\star}$ attack, but the reverse does not occur (see Figure 9 in the appendix). Similarly, smaller triggers enhance larger ones more than the reverse.

4.4 ADDITIONAL OBSERVATIONS ON INHERENT VS. INJECTED TRIGGERS IN MM

Inherent similarity of backdoor attacks with ★ triggers. State-of-the-art backdoor attacks on MM Zhang et al. (2024a) exploit adversarial vulnerabilities in M_{pre} (see Figure 7 and Figure 8). As a result, M_{pre} -optimized attacks (attacks with ★ triggers) are *inherently* similar, yielding highly aligned BVs with strong cosine similarity (see Table 2). Interestingly, $BV_{\star}$ exhibit greater mutual similarity than TVs from the same task. This suggests that inherent triggers share generalizable structure, as evidenced by high $BV_{\star}$ similarity and strong backdoor transfer across attacks (see Figure 3).

Similarity and transfer of $BV_{\star}$ vs $BV_{\oplus}$. Interestingly, BVs from different attacks can be more similar than $BV_{\oplus}$ s or same-task TVs. Additionally, as shown in Figure 9, backdoor transfer is notably stronger from $BV_{\oplus}$ to $BV_{\star}$, reinforcing the idea that many triggers share a generalizable structure – an insight central to our IBVS defense.

5 DISCUSSION AND RELATED WORK

We present extended related work in the appendix A.1, and focus on backdoor attacks on recently proposed backdoor attacks on MM to broaden the context of our contributions.

Backdoor Attacks on Model Merging. Model merging usually operates under the assumption that task vectors are obtained from reliable, trustworthy sources and, therefore, can be utilized without posing security risks. However, recent studies Zhang et al. (2024a); Hsu et al. (2025); Yin et al. (2024); Guo et al. (2025); Hammoud et al. (2024) have raised concerns about the security of the model merging process, showing that maliciously crafted task vectors can potentially compromise the integrity of the merged model. A key example is BadMerging Zhang et al. (2024a), which demonstrates that an adversary can implant backdoors into a model by crafting task vectors that, when merged, result in malicious model behavior. Follow-up work has expanded this threat model. BADTV Hsu et al. (2025), shows how third-party task vectors can to insert hidden functionality into the merged model without access to training data. Other techniques, like LoBAM Yin et al. (2024), take advantage of low-rank adaptation modules to inject backdoors in a parameter-efficient manner, preserving standard performance on clean data. Recent works explore privacy leakage Guo et al. (2025) and multi-stage attacks Guo et al. (2025); Lu et al. (2025), where backdoors are composed or activated through a sequence of merging steps.

We argue that analyzing backdoor attacks on MM is a complex issue and standard BA and ASR metrics used in the above works to describe backdoor attack performance is not detailed enough to fully quantify their behavior. We propose Backdoor Vector framework for the emerging field of backdoor attacks on MM for better understanding of backdoors. We think it is a crucial step to mitigate emerging backdoor threats. Our work is a substantial step forward in this emerging field, as we present unified framework as well as its direct practice benefits, like SBV method that outperform current state-of-the-art by a large margin, with no significant computations added.

Defenses. In response, several defences have been proposed to counteract backdoors in model merging. Subspace masking Yang et al. (2025) introduced by Yang et al. attempts to constrain the merged model weights to parameter regions that are less susceptible to adversarial attacks. Another, proposed by Arora et al. Arora et al. (2024), makes the counterintuitive observation that merging multiple backdoored models may in some cases cancel out their respective backdoors – a phenomenon they term the “Free Lunch” effect. Chen et al. Chen et al. (2024) pursue an active approach in identifying conflicts between task vectors and leverage these oppositions to neutralize backdoor effects.

Our results put the “Free Lunch” hypothesis under the new light and show that the adversary *can* craft much more resilient backdoors with *inherent triggers*, and also significantly boost the standard backdoor merging resilience by merging multiple backdoors into one using SBV method. Additionally, those insights lead us to formulate the hypothesis that backdoor attacks share *inherent* similarities, which can be also used for defense (like our proposed IBVS).

6 CONCLUSIONS

In this work, we introduced a novel perspective on backdoor attacks in model merging by framing them as task arithmetic problems. By defining and analyzing *Backdoor Vectors* (BVs), we provided a simple yet powerful abstraction that enables the injection, transfer, and mitigation of backdoors via vector operations. We show that BVs expose key insights into attack dynamics and enable stronger attacks through *Sparse Backdoor Vectors* (SBVs), which merge multiple BVs into a single, resilient and highly effective threat. On the defense side, we proposed *Injection BV Subtraction* (IBVS) – a lightweight, assumption-free method for mitigating unknown backdoor attacks. Our framework unifies multiple phenomena observed in model merging security and opens new directions to understand and secure model merging pipelines.

Limitations Our work shares a common assumption in the MM literature that task vectors are linearly additive. This assumption holds when operating in a linear connectivity regime, which is induced by the scale of the model and amount of pretraining. This work focuses on CLIP-like vision models and image classification tasks; its applicability to other architectures (e.g., large language models) remains to be explored.

6.1 ETHICS STATEMENT

Recently, backdoor attacks and (more broadly) data poisoning attacks have gained increasing attention due to their significance as a threat to open model sharing paradigms. By sharing results of our investigations on backdoors in MM we aim to broaden the access to the information on such threats in the community and increase the interest of researchers in active research to counter newly discovered attacks.

Our work on understanding backdoor attacks in model merging enables defenders to better detect, understand, and counteract such threats. By shedding light on the attacks using task arithmetic, we aim to inform the research community and practitioners deploying merged models – do not underestimate the data poisoning threat in MM paradigm.

Framing backdoor attacks as TVs provides powerful tools to study them in MM beyond standard CA/BA/ASR metrics (similarity, backdoor transfer, ‘the shared internal structure’ of inherent triggers), which is beneficial for the community.

Additionally, as most of backdoor attacks, our SBVs can be used for IP protection by allowing a model provider to embed a backdoor as a watermark, enabling verification of their model’s presence even after it has been merged with others.

6.2 REPRODUCIBILITY STATEMENT

For reproduction of our experiments see Section 4.1 in the main section of our work. All remaining details are described in Section B in the Appendix.

REFERENCES

- Gorka Abad, Stjepan Picek, Lorenzo Cavallaro, and Aitor Urbieto. Context is the Key: Backdoor Attacks for In-Context Learning with Vision Transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.
- Ansh Arora, Xuanli He, Maximilian Mozes, Srinibas Swain, Mark Dras, and Qionghai Xu. Here’s a free lunch: Sanitizing backdoored models with model merge. *arXiv preprint arXiv:2402.19334*, 2024.
- Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 2938–2948, 2020.
- Chen Chen, Yuchen Sun, Xueluan Gong, Jiaxin Gao, and Kwok-Yan Lam. Neutralizing backdoors through information conflicts for large language models. *arXiv preprint arXiv:2411.18280*, 2024.
- Xinyun Chen, Chang Liu, Bo Li, Kihwan Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. In *Proceedings of the 35th Annual Computer Security Applications Conference (ACSAC)*, pp. 27–36, 2017.
- Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. IEEE, 2009.
- Antonio Andrea Gargiulo, Donato Crisostomi, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, and Emanuele Rodolà. Task singular vectors: Reducing task interference in model merging. In *CVPR*, 2025.
- Micah Goldblum, Luke Fowl, Xueqian Huang, Justin Terry, Lei Li, Utku Evci, and Tom Goldstein. Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses. *Communications of the ACM*, 65(10):60–71, 2022.
- Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. In *NeurIPS 2017 Workshop on Machine Learning and Security*, 2017.

- Wei Guo, Benedetta Tondi, and Mauro Barni. A master key backdoor for universal impersonation attack against dnn-based face verification. *Pattern Recognition Letters*, 144:61–67, April 2021. ISSN 0167-8655. doi: 10.1016/j.patrec.2021.01.009. URL <http://dx.doi.org/10.1016/j.patrec.2021.01.009>.
- Zhenyuan Guo, Yi Shi, Wenlong Meng, Chen Gong, Chengkun Wei, and Wenzhi Chen. Be cautious when merging unfamiliar llms: A phishing model capable of stealing privacy. *arXiv preprint arXiv:2502.11533*, 2025.
- Hasan Abed Al Kader Hammoud, Umberto Michieli, Fabio Pizzati, Philip Torr, Adel Bibi, Bernard Ghanem, and Mete Ozay. Model merging and safety alignment: One bad model spoils the bunch, 2024.
- Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.
- Patrik O Hoyer. Non-negative matrix factorization with sparseness constraints. *Journal of machine learning research*, 5(Nov):1457–1469, 2004.
- Chia-Yi Hsu, Yu-Lin Tsai, Yu Zhe, Yan-Lun Chen, Chih-Hsun Lin, Chia-Mu Yu, Yang Zhang, Chun-Ying Huang, and Jun Sakuma. Badtv: Unveiling backdoor threats in third-party task vectors, 2025. URL <https://arxiv.org/abs/2501.02373>.
- Binxiao Huang, Jason Chun Lok, Chang Liu, and Ngai Wong. Poisoning-based backdoor attacks for arbitrary target label with positive triggers. *arXiv preprint arXiv:2405.05573*, 2024.
- Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *International Conference on Learning Representations (ICLR)*, 2022. URL <https://arxiv.org/abs/2212.04089>.
- Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing Models with Task Arithmetic. In *ICLR*, 2023a.
- Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Benjamin Recht, Hannaneh Hajishirzi, and Ludwig Schmidt. Editing models with task arithmetic. In *International Conference on Machine Learning (ICML)*, 2023b.
- Pavel Izmailov, Dmitrii Podoprikin, T. Garipov, D. Vetrov, and A. Wilson. Averaging weights leads to wider optima and better generalization. *Conference on Uncertainty in Artificial Intelligence*, 2018.
- Dong-Hwan Jang, Sangdoo Yun, and Dongyoon Han. Model Stock: All we need is just a few fine-tuned models. In *ECCV*, 2024.
- Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 554–561, 2013.
- Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.
- Yuezun Li, Yiming Li, Baoyuan Wu, Longkang Li, Ran He, and Siwei Lyu. Invisible Backdoor Attack with Sample-Specific Triggers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16443–16452, 2021.
- Siyuan Liang, Mingli Zhu, Aishan Liu, Baoyuan Wu, Xiaochun Cao, and Ee-Chien Chang. BadCLIP: Dual-Embedding Guided Backdoor Attack on Multimodal Contrastive Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24645–24654, June 2024.
- Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. Trojaning attack on neural networks. *NDSS*, 2018.

- Lin Lu, Zhigang Zuo, Ziji Sheng, and Pan Zhou. Merger-as-a-stealer: Stealing targeted pii from aligned llms with model merging. *arXiv preprint arXiv:2502.16094*, 2025.
- Daniel Marczak, Bartłomiej Twardowski, Tomasz Trzcíński, and Sebastian Cygert. Magmax: Leveraging model merging for seamless continual learning. In *ECCV*, 2024.
- Daniel Marczak, Simone Magistri, Sebastian Cygert, Bartłomiej Twardowski, Andrew D. Bagdanov, and Joost van de Weijer. No task left behind: Isotropic model merging with common and task-specific subspaces. In *ICML*, 2025.
- Tuan Anh Nguyen and Anh Tuan Tran. Input-Aware Dynamic Backdoor Attack. In *Advances in Neural Information Processing Systems (NeurIPS)* 33, 2020.
- Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. Task arithmetic in the tangent space: Improved editing of pre-trained models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL <https://arxiv.org/abs/2305.12827>.
- Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and C V Jawahar. Cats and dogs. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3498–3505. IEEE, 2012.
- Yezhen Qi et al. Towards Practical Noisy-Label Backdoor Attacks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
- 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>.
- Alexandre Rame, Matthieu Kirchmeyer, Thibaud Rahier, Alain Rakotomamonjy, Patrick Gallinari, and Matthieu Cord. Diverse weight averaging for out-of-distribution generalization. 2022.
- Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The german traffic sign recognition benchmark: A multi-class classification competition. In *The 2011 International Joint Conference on Neural Networks*, pp. 1453–1460. IEEE, 2011.
- Anke Tang, Li Shen, Yong Luo, Liang Ding, Han Hu, Bo Du, and Dacheng Tao. Concrete subspace learning based interference elimination for multi-task model fusion. *arXiv preprint arXiv:2312.06173*, 2023. URL <https://arxiv.org/abs/2312.06173>.
- Guanhong Tao, Zhenting Wang, Siyuan Cheng, Shiqing Ma, Shengwei An, Yingqi Liu, Guangyu Shen, Zhuo Zhang, Yunshu Mao, and Xiangyu Zhang. Backdoor vulnerabilities in normally trained deep learning models. *arXiv preprint arXiv:2211.15929*, 2022.
- Guanhong Tao, Siyuan Cheng, Zhenting Wang, Shiqing Ma, Shengwei An, Yingqi Liu, Guangyu Shen, Zhuo Zhang, Yunshu Mao, and Xiangyu Zhang. Exploring inherent backdoors in deep learning models. In *2024 Annual Computer Security Applications Conference (ACSAC)*, pp. 923–939, 2024. doi: 10.1109/ACSAC63791.2024.00078.
- Huynh Tran, Anh Tran, Khoa Doan, and Tung Pham. Data poisoning quantization backdoor attack. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.
- Alexander Turner, Dimitris Tsipras, and Aleksander Madry. Label-consistent backdoor attacks. In *Proceedings of the 36th International Conference on Machine Learning (ICML) Workshop on Security and Privacy of Machine Learning*, 2019.
- Ke Wang, Nikolaos Dimitriadis, Guillermo Ortiz-Jiménez, François Fleuret, and Pascal Frossard. Localizing task information for improved model merging and compression. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024a. URL <https://openreview.net/forum?id=DWT9uiGjxT>.
- Ke Wang, Nikolaos Dimitriadis, Alessandro Favero, Guillermo Ortiz-Jiménez, François Fleuret, and Pascal Frossard. Lines: Post-training layer scaling prevents forgetting and enhances model merging. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL <https://openreview.net/forum?id=J5sUOv1LbQ>.

- Ren Wang, Xinyun Zhang, Xiangyu Zhang, Zhiqiang Wu, Yao Xie, and Wen-Chuan Lee. Waveattack: Asymmetric frequency obfuscation-based backdoor attack. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024b.
- Zhenting Wang, Hailun Ding, Juan Zhai, and Shiqing Ma. Training with more confidence: Mitigating injected and natural backdoors during training. In *Advances in Neural Information Processing Systems*, volume 35, 2022.
- Emily Wenger, Roma Bhattacharjee, Arjun Nitin Bhagoji, Josephine Passananti, Emilio Andere, Haitao Zheng, and Ben Y. Zhao. Finding naturally occurring physical backdoors in image datasets. In *Advances in Neural Information Processing Systems*, volume 35, 2022.
- Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *ICML*, 2022.
- Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3485–3492. IEEE, 2010.
- Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. URL <https://arxiv.org/abs/2306.01708>.
- Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities, 2024. URL <https://arxiv.org/abs/2408.07666>, 2408.
- Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. Adamerging: Adaptive model merging for multi-task learning. *arXiv preprint arXiv:2310.02575*, 2023. URL <https://arxiv.org/abs/2310.02575>.
- Enneng Yang, Li Shen, Zhenyi Wang, Guibing Guo, Xiaojun Chen, Xingwei Wang, and Dacheng Tao. Representation surgery for multi-task model merging. *arXiv preprint arXiv:2402.02705*, 2024. URL <https://arxiv.org/abs/2402.02705>.
- Jinluan Yang, Anke Tang, Didi Zhu, Zhengyu Chen, Li Shen, and Fei Wu. Mitigating the backdoor effect for multi-task model merging via safety-aware subspace, 2025. URL <https://arxiv.org/abs/2410.13910>.
- Ming Yin, Jingyang Zhang, Jingwei Sun, Minghong Fang, Hai Li, and Yiran Chen. Lobam: Lora-based backdoor attack on model merging. *arXiv preprint arXiv:2411.16746*, 2024.
- Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*, 2024.
- Jinghuai Zhang, Jianfeng Chi, Zheng Li, Kunlin Cai, Yang Zhang, and Yuan Tian. Badmerging: Backdoor attacks against model merging. In *Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2024a.
- Jinghuai Zhang, Hongbin Liu, Jinyuan Jia, and Neil Zhenqiang Gong. Data Poisoning based Backdoor Attacks to Contrastive Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24357–24366, June 2024b.

A ADDITIONAL INFORMATION

A.1 RELATED WORK ON MODEL MERGING AND BACKDOOR ATTACKS

Model Merging Model merging aims to integrate weights of multiple models. Historically, averaging weights along the training trajectory was used to improve the generalization of the model Izmailov et al. (2018). More recently, many works (Wortsman et al., 2022; Rame et al., 2022; Jang et al., 2024) have shown that merging multiple independent fine-tunings on a given task improves model’s performance in-distribution and out-of-distribution. On the other hand, model merging can be used to integrate the knowledge of multiple models fine-tuned on disjoint tasks to obtain a multi-task models Yadav et al. (2023); Ilharco et al. (2022); Tang et al. (2023); Ortiz-Jimenez et al. (2024); Yang et al. (2024; 2023).

Backdoor Attacks Backdoor attacks Gu et al. (2017); Liu et al. (2018) are a class of adversarial techniques that implant malicious hidden behavior into machine learning models by poisoning training data Chen et al. (2017); Turner et al. (2019); Zhang et al. (2024b); Tran et al. (2024) or manipulating the training procedure Bagdasaryan et al. (2020); Goldblum et al. (2022); Huang et al. (2024); Wang et al. (2024b). It is hard to detect a model compromised in this manner because the model performs as expected on clean inputs, but produces attacker-specified outputs only when a particular trigger is present. Recent works Tao et al. (2024); Wenger et al. (2022); Tao et al. (2022); Wang et al. (2022) show that pre-trained models can also exhibit *inherent triggers*—natural adversarial vulnerabilities that act as backdoors without explicit trigger injection.

A.2 CLIP-LIKE CLASSIFIERS

Contrastive Language–Image Pretraining Radford et al. (2021) (CLIP)-like classifier employ contrastive learning to align visual and textual modalities within a joint embedding space. The classifier model $M = \{V, T\}$ uses an image encoder $V(x)$ and a text encoder $T(t)$, respectively. Given an image x and a set of textual class descriptions $C = \{c_1, c_2, \dots, c_k\}$, the model computes similarity scores using the cosine similarity between the image embedding and each class text embedding. The similarity score s_j for class j is defined as:

$$s_j = \frac{V(x) \cdot T(c_j)}{\|V(x)\| \|T(c_j)\|}, \quad \text{for } j = 1, \dots, k. \quad (4)$$

The predicted class $\hat{y}$ is then obtained by selecting the class with the highest similarity score:

$$\hat{y} = \arg \max_j s_j. \quad (5)$$

The formulation in Equations equation 4–equation 5 enables zero-shot classification, as the model can generalize to previously unseen categories based solely on their textual descriptions. Nevertheless, to enhance performance on a specific downstream task with a fixed set of class labels $\mathcal{C}$, the model can be further adapted by optimizing a task-specific classifier using the cross-entropy loss:

$$\mathcal{L}_{\text{CE}}(M(x, \mathcal{C}), y), \quad (6)$$

where $M(x, \mathcal{C})$ denotes the similarity-based prediction over the class set $\mathcal{C}$, and y is the ground-truth label. A common Ilharco et al. (2022); Tang et al. (2023); Ortiz-Jimenez et al. (2024); Yadav et al. (2023); Yang et al. (2024; 2023) fine-tuning strategy involves freezing the text encoder T and updating only the vision encoder V , which has been empirically shown to yield the best performance Ilharco et al. (2022).

A.3 LIMITATIONS

Our work focuses on CLIP-like vision models and image classification tasks; its applicability to other architectures (e.g., large language models) remains to be explored.

By introducing BVs we offer efficient, intuitive ways to capture with Task Arithmetic (TA) the complex interplay between task vectors and backdoors in MM. That is why we focus on TA and not any other novel MM method. The applicability of the obtained results to other methods remains to be explored.

Our IBVS method is rather a simple defense baseline than a final robust solution to backdoor problems in MM. We introduce it for further defense studies, as it requires almost no computation overhead to the defender, and its assumptions are minimalistic.

A.4 LLM USAGE

Adhering to ICLR LLM usage policy we hereby declare that we used LLMs to polish the writing of the manuscript, style tables and for finding additional related work connected with backdoor attacks on MM.

B EXPERIMENTAL DETAILS

B.1 CODE TO REPRODUCE OUR FINDINGS

We attach the code repository to our submission with implementation details to allow the reproduction of our main results. The code repository structure is adapted from Zhang et al. (2024a), to enable simple use of our method as an add-on to existing backdoor attacks on MM.

B.2 BACKDOOR TRIGGER CONFIGURATION

Backdoor triggers are defined by a fixed set of parameters. These control both their visual appearance and how they interact with the model during training (see Table 5).

For our main experiments we use:

- **Patch Size:** 22×22 (1% pixels of the image), following Zhang et al. (2024a),
- **Location:** default bottom-right. We also test different patch locations for backdoor transfer experiments: bottom-left, upper-right, upper-left,
- **Trigger Type:** we use both types of triggers ($\star$, $\oplus$) for different attacks, and show that proposed SBV merging improves ASR for both of them. We use $\oplus$ triggers for our proposed defense IBVS (white square trigger and wavelet trigger from Badnets Gu et al. (2017)),
- **Target class:** We average all our main results on five different target classes. Table 6 presents selected target classes from adversary tasks in our experiments. Details on used ImageNet100 classes are in the code repository added to the appendix,
- **Optimization seed:** We use 15 different optimization seeds for our experiments on SBV merging (some needed to calculate SBV, other to create 10 different models for single-task merging experiments),
- α We set this parameter to 5.0 following previous work Zhang et al. (2024a).

Table 5: Trigger configuration parameters used for backdoor attack design.

Parameter	Description
Patch Size	Spatial dimensions of the trigger (e.g., 16×16 – 0.5%, 22×22 – 1%)
Location	Position of the patch within the input (e.g., bottom-right)
Trigger Type	Inherent ($\star$) or injected $\oplus$ (like fixed white square $\square$) used in BadNets Gu et al. (2017)
Target Class	Output class enforced by the backdoor attack when triggered
Optimization Seed	Random seed used in training $M_{\text{backdoored}}$ or inherent trigger ($\star$) optimization
α	Loss weight parameter for backdoor attack loss in finetuning $M_{\text{backdoored}}$

B.3 BACKDOOR TARGET CLASS

We present output class enforced by the backdoor attack used in our experiments in Table 6. We average all the main results between results obtained for these five target classes.

Table 6: Selection of the target class for each task.

Task	Target Class				
	1	2	3	4	5
CIFAR100	beaver	dolphin	otter	seal	whale
ImageNet100	american coot	harvestman	macaw	bittern	electric ray

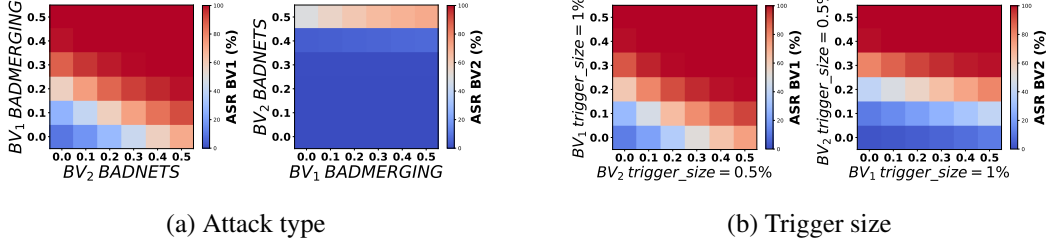


Figure 9: **ViT-B-32: Backdoor transfer is asymmetric across attack types and trigger sizes:** $(BV_2 \rightarrow BV_1) > (BV_1 \rightarrow BV_2)$. (a) BadNets (BV_2) attack boosts the ASR of the BadMerging (BV_1) attack, but not vice versa. (b) A smaller trigger attack (0.5% of image pixels) enhances the ASR of a BV_1 attack with a larger trigger (1%), while the reverse transfer is much weaker.

C ADDITIONAL RESULTS

C.1 ADDITIONAL RESULTS ON ViT-B-32

C.1.1 ADDITIONAL RESULTS ON THE IMPORTANCE OF K-PARAMETER

Our findings (see Table 7) demonstrate that increasing the number of BVs used to form a single SBV leads to stronger and more robust attacks. Not only does the ASR improve with higher values of k (up to 5), but the results also become more stable - exhibiting reduced sensitivity to random noise during the merging process. This is reflected in the lower standard deviation reported across multiple runs.

Table 7: **Backdoor Merging: Effect of k in Algorithm 2 under Single-Task with ViT-B.** Single-task, $\lambda = 0.1$, 10 tasks, TA. **Methods:** BM(★) — BadMerging

Backdoor Attack			Adversary task: CIFAR100			Adversary task: IMAGENET100		
Setting	Method	BV merging	CA	BA	ASR	CA	BA	ASR
Single-task	BM(★)	BV($k=1$)	89.70 $\pm$ 0.03	89.74 $\pm$ 0.04	27.29 $\pm$ 12.14	85.88 $\pm$ 0.07	85.91 $\pm$ 0.04	84.01 $\pm$ 7.83
	BM(★)	SBV($k=2$)	89.70 $\pm$ 0.03	89.76 $\pm$ 0.03	55.09 $\pm$ 14.09	85.88 $\pm$ 0.07	85.84 $\pm$ 0.05	98.22 $\pm$ 0.89
	BM(★)	SBV($k=3$)	89.70 $\pm$ 0.03	89.73 $\pm$ 0.06	79.68 $\pm$ 10.98	85.88 $\pm$ 0.07	85.86 $\pm$ 0.07	99.71 $\pm$ 0.24
	BM(★)	SBV($k=4$)	89.70 $\pm$ 0.03	89.73 $\pm$ 0.06	91.95 $\pm$ 5.86	85.88 $\pm$ 0.07	85.87 $\pm$ 0.06	99.93 $\pm$ 0.05
	BM(★)	SBV($k=5$)	89.70 $\pm$ 0.03	89.69 $\pm$ 0.06	97.21 $\pm$ 2.30	85.88 $\pm$ 0.07	85.83 $\pm$ 0.09	99.99 $\pm$ 0.01

C.1.2 IBVS PERFORMANCE

Despite its simplicity, IBVS demonstrates strong generalization across architectures such as ViT-B, ViT-L, and ConvNeXt. It consistently achieves substantial reductions in attack success rate (ASR), with performance degradation (in terms of backdoor accuracy, BA) typically below 1 percentage point.

Tables 8, 13 and 15 present results for ViT-B/ViT-L/ConvNext using varying numbers of BVs in SBV construction (parameter k). Note that BadMerging corresponds to BV($k=1$). Reductions in ASR due to our defense are shown as Δ and are highlighted in bold.

C.1.3 BACKDOOR TRANSFER

Figures 10 and 11 show the asymmetry of backdoor transfer given different trigger sizes across four different patch sizes (19 x 19, 22 x 22, 25 x 25, 28 x 28). Similarly to results from Figure 9b, smaller trigger patches enhance the ASR of bigger ones, and the reverse transfer is much weaker.

Table 8: **Backdoor Merging: Effect of IBVS Defense under Single-Task with ViT-B.** Single-task, $\lambda = 0.1$, 10 tasks, TA. BadMerging (BM, ★) with $\lambda_{IBVS} = 0.5$ when defense is enabled. k – number of BVs merged for an attack. **Methods:** BM(★) — BadMerging.

Backdoor Attack			Adversary task: CIFAR100		Adversary task: IMAGENET100	
Defense	Method	BV merging	BA	ASR	BA	ASR
-	BM(★)	BV(k=1)	89.74 $\pm$ 0.04	27.29 $\pm$ 12.14	85.91 $\pm$ 0.04	84.01 $\pm$ 7.83
IBVS	BM(★)	BV(k=1)	89.06 $\pm$ 0.11	11.10 $\pm$ 8.96	85.20 $\pm$ 0.12	50.74 $\pm$ 17.35
		Δ (IBVS – none)	-0.68	-16.19	-0.71	-33.27
-	BM(★)	SBV(k=2)	89.76 $\pm$ 0.03	35.09 $\pm$ 14.09	85.84 $\pm$ 0.05	98.22 $\pm$ 0.89
IBVS	BM(★)	SBV(k=2)	89.03 $\pm$ 0.11	26.17 $\pm$ 15.53	85.27 $\pm$ 0.17	84.45 $\pm$ 10.19
		Δ (IBVS – none)	-0.73	-28.92	-0.57	-13.77
-	BM(★)	SBV(k=3)	89.73 $\pm$ 0.06	79.68 $\pm$ 10.98	85.86 $\pm$ 0.07	99.71 $\pm$ 0.24
IBVS	BM(★)	SBV(k=3)	88.98 $\pm$ 0.07	47.09 $\pm$ 20.33	85.34 $\pm$ 0.21	96.42 $\pm$ 3.17
		Δ (IBVS – none)	-0.75	-32.59	-0.52	-3.29
-	BM(★)	SBV(k=4)	89.73 $\pm$ 0.06	91.95 $\pm$ 5.86	85.87 $\pm$ 0.06	99.93 $\pm$ 0.05
IBVS	BM(★)	SBV(k=4)	89.00 $\pm$ 0.11	64.91 $\pm$ 20.54	85.22 $\pm$ 0.22	99.09 $\pm$ 0.95
		Δ (IBVS – none)	-0.73	-27.04	-0.65	-0.84
-	BM(★)	SBV(k=5)	89.69 $\pm$ 0.06	97.21 $\pm$ 2.30	85.83 $\pm$ 0.09	99.99 $\pm$ 0.01
IBVS	BM(★)	SBV(k=5)	89.00 $\pm$ 0.11	64.91 $\pm$ 20.54	85.22 $\pm$ 0.22	99.09 $\pm$ 0.95
		Δ (IBVS – none)	-0.69	-32.30	-0.61	-0.90

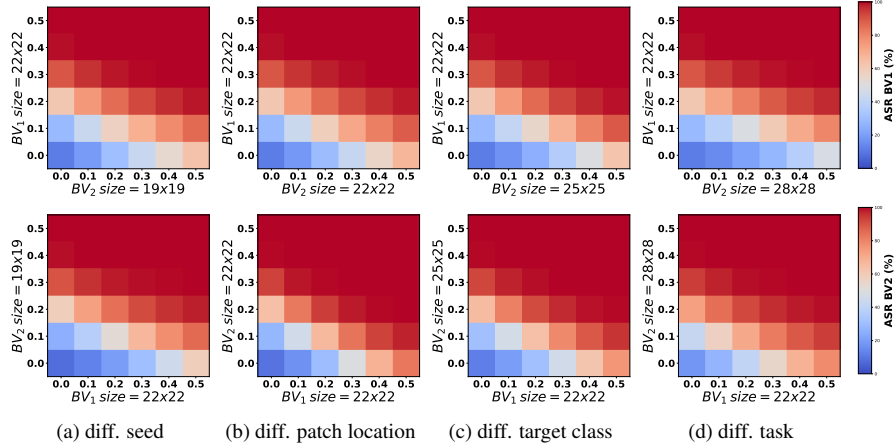


Figure 10: **CIFAR100 : Backdoor Transfer Asymmetry For Different Trigger Sizes.** Axes show λ_{BV_1} , λ_{BV_2} ; A smaller trigger patch attack (e.g. 19x19 pixels) enhances the ASR of an attack with a larger trigger (e.g. 22x22 pixels), while the reverse transfer is much weaker.

C.1.4 SBV: MULTI-TASK SETUP

Figure 12 shows how the adversary may use inherent triggers to attack the model even without contributing poisoned $M_{\text{backdoored}}$ during merging process. We show using ACC/ASR trajectories that inherent triggers are much more resilient in multi-task setup, than in single-task setup (see Figure 8). Tasks are different from each other and have cosine similarity close to zero, which results in small overlap in task vectors. Since different tasks interfere less with inherent trigger, the final attack success rate is high.

Table 9 show comparison between single- and multi-task scenarios for 6 tasks. Our SBV merging method improves the ASR in all tested scenarios. It is the first method that enables even the most simple, classical backdoor attack with injected triggers to withstand the merging process.

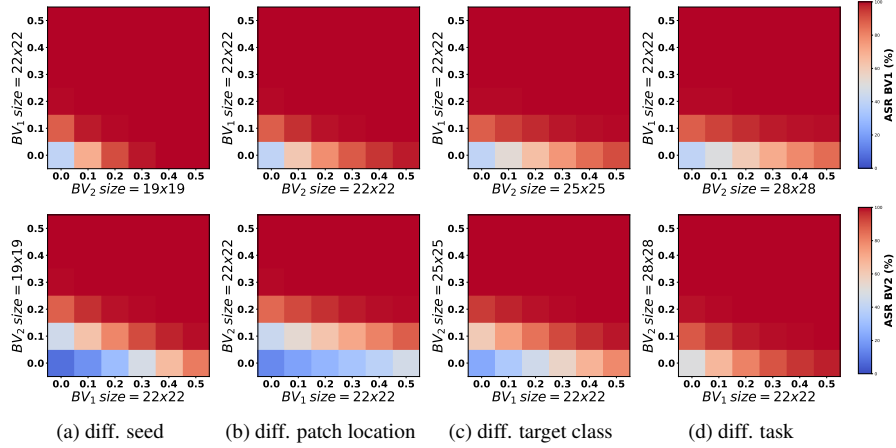


Figure 11: **ImageNet100: Backdoor Transfer Asymmetry For Different Trigger Sizes.** Axes show λ_{BV_1} , λ_{BV_2} ; A smaller trigger patch attack (e.g. 19x19 pixels) enhances the ASR of an attack with a larger trigger (e.g. 22x22 pixels), while the reverse transfer is much weaker.

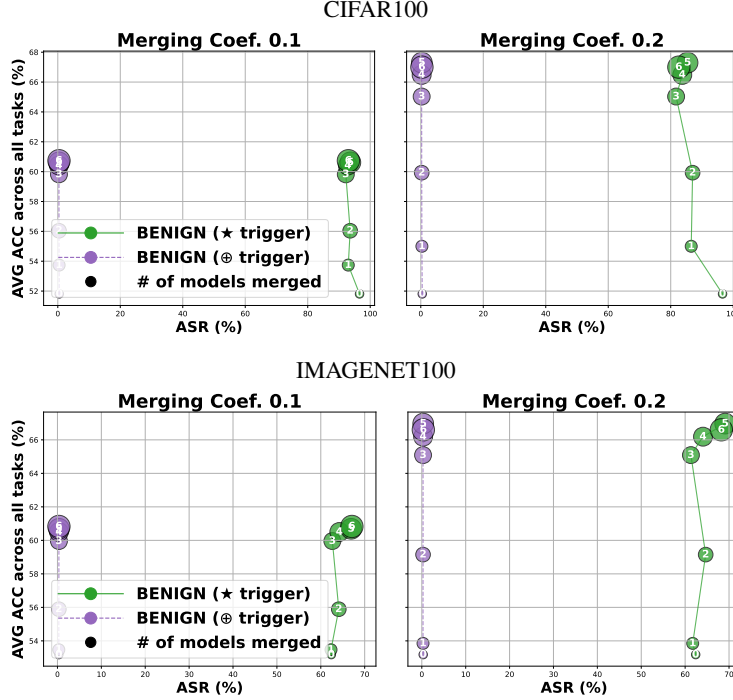


Figure 12: **ACC/ASR trajectories for inherent (★) and injected (⊕) triggers in all-clean multi-task MM.** Once again ★ triggers maintain their effectiveness even after merging multiple M_{clean} models, while ⊕ triggers are completely ineffective in the absence of $M_{\text{backdoored}}$ model. Contrary to Figure 8, in multi-task setup inherent triggers do not weaken with more models merged, as tasks are much more independent and do not interfere with adversary task (cosine similarities between different TVs are close to 0.)

C.1.5 IBVS: λ PARAMETER

Table 10 shows the impact of λ parameter for IBVS. Stronger defense merging coefficient (λ_{IBVS}) decrease the ASR more, but it comes with the cost of small final accuracy decrease. With smaller values of λ_{IBVS} , like 0.1 the defender can still lower the backdoor ASR, without any loss on the accuracy.

Table 9: **Backdoor Merging: Single- and Multi-Task Attack Results.** We merge $1 \times \theta_{\text{backdoored}} + 5 \times \theta_{\text{clean}}$ with $\lambda = 0.2$, reporting mean $\pm$ std. dev. **BV merging:** *AVG* — weight averaging **Methods:** $\text{BN}(\oplus)$ — BadNets, $\text{BM}(\star)$ — BadMerging. TA. Visual encoder: ViT-B-32. Tasks for Multi-task setting: Adversary task, Cars Krause et al. (2013), SUN397 Xiao et al. (2010), EuroSAT Helber et al. (2019), GTSRB Stallkamp et al. (2011) and Pets Parkhi et al. (2012).

Backdoor Attack			Adversary task: CIFAR100			Adversary task: IMAGENET100		
Setting	Method	BV merging	CA	BA	ASR	CA	BA	ASR
Single-task	$\text{BN}(\oplus)$	-	89.43 $\pm$ 0.06	89.45 $\pm$ 0.03	0.25 $\pm$ 0.13	85.41 $\pm$ 0.11	85.54 $\pm$ 0.06	0.40 $\pm$ 0.16
	$\text{BN}(\oplus)$	AVG	89.43 $\pm$ 0.06	89.51 $\pm$ 0.03	0.24 $\pm$ 0.13	85.41 $\pm$ 0.11	85.55 $\pm$ 0.08	0.41 $\pm$ 0.17
	$\text{BN}(\oplus)$	<i>SBV_{RDM}</i> (Ours)	89.43 $\pm$ 0.06	89.43 $\pm$ 0.04	0.48 $\pm$ 0.21	85.41 $\pm$ 0.11	85.43 $\pm$ 0.10	0.94 $\pm$ 0.24
	$\text{BN}(\oplus)$	<i>SBV_{SC}</i> (Ours)	89.43 $\pm$ 0.06	88.93 $\pm$ 0.07	100.00 $\pm$ 0.00	85.41 $\pm$ 0.11	84.86 $\pm$ 0.16	100.00 $\pm$ 0.00
	$\text{BM}(\star)$	-	89.43 $\pm$ 0.06	89.49 $\pm$ 0.04	34.94 $\pm$ 15.00	85.41 $\pm$ 0.11	85.60 $\pm$ 0.08	96.95 $\pm$ 1.79
	$\text{BM}(\star)$	AVG	89.43 $\pm$ 0.06	89.53 $\pm$ 0.05	35.55 $\pm$ 15.12	85.41 $\pm$ 0.11	85.52 $\pm$ 0.03	97.38 $\pm$ 1.54
	$\text{BM}(\star)$	<i>SBV_{RDM}</i> (Ours)	89.43 $\pm$ 0.06	89.49 $\pm$ 0.08	55.16 $\pm$ 15.88	85.41 $\pm$ 0.11	85.51 $\pm$ 0.11	99.34 $\pm$ 0.57
	$\text{BM}(\star)$	<i>SBV_{SC}</i> (Ours)	89.43 $\pm$ 0.06	89.20 $\pm$ 0.08	100.00 $\pm$ 0.00	85.41 $\pm$ 0.11	85.18 $\pm$ 0.08	100.00 $\pm$ 0.00
Multi-task	$\text{BN}(\oplus)$	-	66.93 $\pm$ 0.11	66.88 $\pm$ 0.07	2.04 $\pm$ 1.59	66.60 $\pm$ 0.13	66.54 $\pm$ 0.07	0.67 $\pm$ 0.42
	$\text{BN}(\oplus)$	AVG	66.93 $\pm$ 0.11	66.94 $\pm$ 0.05	2.07 $\pm$ 1.59	66.60 $\pm$ 0.13	66.56 $\pm$ 0.04	0.69 $\pm$ 0.41
	$\text{BN}(\oplus)$	<i>SBV_{RDM}</i> (Ours)	66.93 $\pm$ 0.11	66.88 $\pm$ 0.07	18.5 $\pm$ 10.96	66.60 $\pm$ 0.13	66.54 $\pm$ 0.07	2.59 $\pm$ 1.05
	$\text{BN}(\oplus)$	<i>SBV_{SC}</i> (Ours)	66.93 $\pm$ 0.11	65.66 $\pm$ 0.31	100.0 $\pm$ 0.0	66.60 $\pm$ 0.13	65.23 $\pm$ 0.2	100.0 $\pm$ 0.0
	$\text{BM}(\star)$	-	66.93 $\pm$ 0.11	66.99 $\pm$ 0.06	99.92 $\pm$ 0.06	66.60 $\pm$ 0.13	66.67 $\pm$ 0.04	99.89 $\pm$ 0.15
	$\text{BM}(\star)$	AVG	66.93 $\pm$ 0.11	67.09 $\pm$ 0.05	99.92 $\pm$ 0.07	66.60 $\pm$ 0.13	66.69 $\pm$ 0.05	99.92 $\pm$ 0.1
	$\text{BM}(\star)$	<i>SBV_{RDM}</i> (Ours)	66.93 $\pm$ 0.11	67.11 $\pm$ 0.05	99.99 $\pm$ 0.01	66.60 $\pm$ 0.13	66.66 $\pm$ 0.07	100.0 $\pm$ 0.01
	$\text{BM}(\star)$	<i>SBV_{SC}</i> (Ours)	66.93 $\pm$ 0.11	66.27 $\pm$ 0.14	100.0 $\pm$ 0.0	66.60 $\pm$ 0.13	65.91 $\pm$ 0.13	100.0 $\pm$ 0.0

Table 10: **Backdoor Defense: Injected BV Subtraction (IBVS).** Single-task results for IBVS variants across $\lambda_{\text{IBVS}} \in \{0.1, 0.3\}$, reporting mean BA and ASR. BV merging: *AVG* — weight averaging. IBVS: *IBVS_□* uses a fixed white-square trigger; *IBVS_{BN}* uses a wavelet trigger from BadNets. Backbone: ViT-B-32. No defense BV: Badmerging attack. We improve state-of-the-art ASR for BV creating stronger attack using our SBV merging method (columns). At the same time we propose a simple defense method (IBVS) to weaken the backdoors during merging and show the decrease of the ASR (rows). Values for $\lambda_{\text{IBVS}} = 0.3$ we report in the main part of the work.

Method	λ_{IBVS}	CIFAR100								ImageNet100							
		BV		AVG		<i>SBV_{RND}</i>		<i>SBV_{SC}</i>		BV		AVG		<i>SBV_{RND}</i>		<i>SBV_{SC}</i>	
		BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR	BA	ASR
No defense	0	89.74	27.29	89.72	27.56	89.72	35.93	89.69	97.21	85.91	84.01	85.96	84.79	85.97	92.55	85.83	99.99
IBVS _□ (Ours)	0.1	89.82	25.60	89.87	26.78	89.89	34.81	89.69	96.75	85.93	78.84	85.65	84.00	85.69	91.92	85.77	99.96
	0.3	89.58	20.65	89.36	23.59	89.41	30.81	89.61	94.69	85.53	66.83	85.13	81.22	85.15	89.74	85.47	99.83
IBVS _{BN} (Ours)	0.1	89.77	23.55	89.86	24.64	89.84	32.38	89.66	95.45	85.88	78.59	85.73	83.88	85.74	91.99	85.79	99.97
	0.3	89.57	16.56	89.41	19.36	89.42	25.37	89.46	89.21	85.61	65.26	85.14	80.89	85.14	90.17	85.59	99.89

C.2 RESULTS ON ViT-L-14 AND CONVNEXT

Figures 13 and Table 12 show the results of our main experiments for larger visual encoder ViT-L-14 (we use mostly ViT-B-32 in the main part of our work). Our main results are consistent across these architectures. ViT-L-14 is more robust than ViT-B-32 to designed backdoor attacks. We suspect that the main reason behind it is the size of tested triggers, which are beyond single patch for small 14x14 ViT-L patches. Training on small patch sizes can increase the robustness of the model to backdoor attacks (it is hard to optimize strong trigger pattern smaller than 14x14), but it comes with much higher cost during the ViT-L training.

Table 11: **Backdoor Merging: Single-Task Attack Results with TIES.** Additional results under single-task setting with $\lambda = 0.1$, 10 tasks, TIES merging. **Methods:** $\text{BN}(\oplus)$ — BadNets, $\text{BM}(\star)$ — BadMerging, SBV — ours.

Backdoor Attack			Adversary task: CIFAR100			Adversary task: IMAGENET100		
Setting	Method	BV merging	CA	BA	ASR	CA	BA	ASR
Single-task	$\text{BN}(\oplus)$	-	87.82 $\pm$ 0.08	87.76 $\pm$ 0.04	0.23 $\pm$ 0.09	85.10 $\pm$ 0.07	85.15 $\pm$ 0.05	0.29 $\pm$ 0.17
	$\text{BN}(\oplus)$	SBV (Ours)	87.82 $\pm$ 0.08	87.68 $\pm$ 0.10	18.07 $\pm$ 11.94	85.10 $\pm$ 0.07	84.81 $\pm$ 0.15	20.0 $\pm$ 9.24
	$\text{BM}(\star)$	-	87.82 $\pm$ 0.08	87.77 $\pm$ 0.03	58.20 $\pm$ 13.26	85.10 $\pm$ 0.07	85.14 $\pm$ 0.05	67.36 $\pm$ 18.28
	$\text{BM}(\star)$	SBV (Ours)	87.82 $\pm$ 0.08	87.82 $\pm$ 0.05	99.93 $\pm$ 0.07	85.10 $\pm$ 0.07	85.06 $\pm$ 0.09	99.98 $\pm$ 0.02

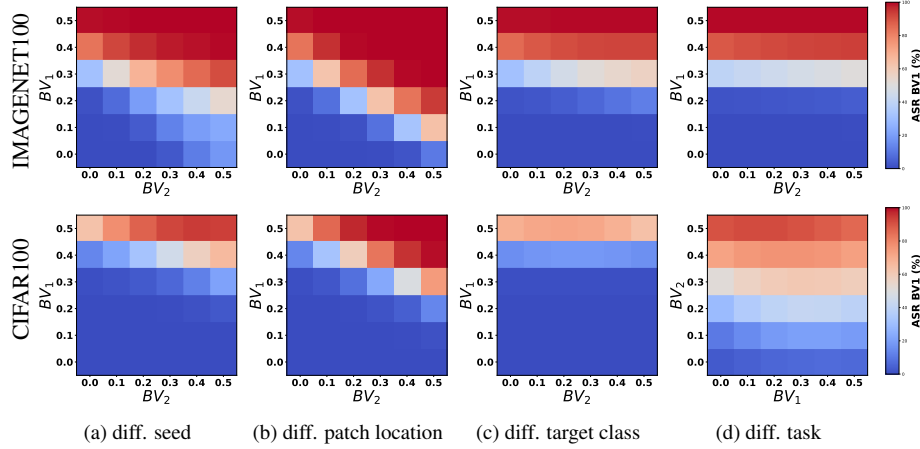


Figure 13: **Positive Backdoor Transfer Across Attacks ($BV_2 \rightarrow BV_1$) for ViT-L-14.** Adding BV_2 increases the ASR of BV_1 . Axes show λ_{BV_1} , λ_{BV_2} ; (a–c) use BVs from the same task (ImageNet100 or CIFAR100). The strongest transfer occurs across patch locations (b) and seeds (a), but is weaker across target classes (c) and tasks (d). In comparison to Figure 3 for ViT-L-14 different seeds has smaller backdoor transfer. We suspect this is connected with trigger size, as 22x22 trigger is bigger than ViT-L-14 training patches.

Table 12: **Backdoor Merging: Single-Task Attack Results with ViT-L-14 Visual Encoder.** We merge $1 \times \theta_{\text{backdoored}} + 9 \times \theta_{\text{clean}}$ with $\lambda = 0.1$, reporting mean $\pm$ std. dev. **BV merging:** *AVG* — weight averaging **Methods:** BN($\oplus$) — BadNets, BM($\star$) — BadMerging.

Backdoor Attack			Adversary task: CIFAR100			Adversary task: IMAGENET100		
Setting	Method	BV merging	CA	BA	ASR	CA	BA	ASR
Single-task	BN($\oplus$)	-	93.82 $\pm$ 0.03	93.86 $\pm$ 0.03	0.09 $\pm$ 0.04	91.20 $\pm$ 0.09	91.22 $\pm$ 0.03	0.11 $\pm$ 0.07
	BN($\oplus$)	AVG	93.82 $\pm$ 0.03	93.85 $\pm$ 0.03	0.09 $\pm$ 0.04	91.20 $\pm$ 0.09	91.21 $\pm$ 0.03	0.1 $\pm$ 0.06
	BN($\oplus$)	SBV_{RDM} (Ours)	93.82 $\pm$ 0.03	93.7 $\pm$ 0.07	0.11 $\pm$ 0.03	91.20 $\pm$ 0.09	91.19 $\pm$ 0.06	0.63 $\pm$ 0.68
	BN($\oplus$)	SBV_{SC} (Ours)	93.82 $\pm$ 0.03	93.38 $\pm$ 0.07	17.06 $\pm$ 16.89	91.20 $\pm$ 0.09	91.11 $\pm$ 0.13	81.86 $\pm$ 8.86
	BM($\star$)	-	93.82 $\pm$ 0.03	93.85 $\pm$ 0.03	0.1 $\pm$ 0.04	91.20 $\pm$ 0.09	91.19 $\pm$ 0.03	0.27 $\pm$ 0.17
	BM($\star$)	AVG	93.82 $\pm$ 0.03	93.82 $\pm$ 0.06	0.09 $\pm$ 0.04	91.20 $\pm$ 0.09	91.22 $\pm$ 0.02	0.26 $\pm$ 0.18
	BM($\star$)	SBV_{RDM} (Ours)	93.82 $\pm$ 0.03	93.7 $\pm$ 0.08	0.16 $\pm$ 0.1	91.20 $\pm$ 0.09	91.2 $\pm$ 0.04	3.98 $\pm$ 3.73
	BM($\star$)	SBV_{SC} (Ours)	93.82 $\pm$ 0.03	93.42 $\pm$ 0.03	32.21 $\pm$ 18.17	91.20 $\pm$ 0.09	91.01 $\pm$ 0.11	94.1 $\pm$ 5.08

Table 13: **Backdoor Merging: Effect of IBVS Defense under Single-Task with ViT-L.** Single-task, $\lambda = 0.1$, 10 tasks, TA. BadMerging (BM, $\star$) with $\lambda_{IBVS} = 0.5$ when defense is enabled. k – number of BVs merged for an attack. **Methods:** BM($\star$) — BadMerging.

Backdoor Attack			Adversary task: CIFAR100		Adversary task: IMAGENET100	
Defense	Method	BV merging	BA	ASR	BA	ASR
-	BM($\star$)	BV(k=1)	93.85 $\pm$ 0.03	0.10 $\pm$ 0.04	91.19 $\pm$ 0.03	0.27 $\pm$ 0.17
IBVS	BM($\star$)	BV(k=1)	92.72 $\pm$ 0.02	0.12 $\pm$ 0.08	90.99 $\pm$ 0.09	0.21 $\pm$ 0.12
		Δ (IBVS – none)	-1.13	+0.02	-0.20	-0.06
-	BM($\star$)	SBV(k=5)	93.42 $\pm$ 0.03	32.21 $\pm$ 18.17	91.01 $\pm$ 0.11	94.10 $\pm$ 5.08
IBVS	BM($\star$)	SBV(k=5)	92.47 $\pm$ 0.00	1.06 $\pm$ 0.93	90.89 $\pm$ 0.17	56.79 $\pm$ 18.42
		Δ (IBVS – none)	-0.95	-31.15	-0.12	-37.31

Table 14: **Backdoor Merging: Single-Task Attack Results with ConvNeXt Backbone.** We merge $1 \times \theta_{\text{backdoored}} + 9 \times \theta_{\text{clean}}$ with $\lambda = 0.1$, reporting mean $\pm$ std. dev. **BV merging:** — no merging. **Methods:** BN($\oplus$) — BadNets, BM($\star$) — BadMerging, SBV — ours.

Backdoor Attack			Adversary task: CIFAR100			Adversary task: IMAGENET100		
Setting	Method	BV merging	CA	BA	ASR	CA	BA	ASR
Single-task	BN($\oplus$)	-	90.62 $\pm$ 0.03	90.71 $\pm$ 0.02	1.06 $\pm$ 1.74	88.63 $\pm$ 0.05	88.73 $\pm$ 0.04	0.16 $\pm$ 0.07
	BN($\oplus$)	SBV (Ours)	90.62 $\pm$ 0.03	90.71 $\pm$ 0.04	100.0 $\pm$ 0.01	88.63 $\pm$ 0.05	88.61 $\pm$ 0.02	99.52 $\pm$ 0.58
	BM($\star$)	-	90.62 $\pm$ 0.03	90.71 $\pm$ 0.05	11.03 $\pm$ 3.71	88.63 $\pm$ 0.05	88.71 $\pm$ 0.02	3.97 $\pm$ 1.57
	BM($\star$)	SBV (Ours)	90.62 $\pm$ 0.03	90.76 $\pm$ 0.02	100.0 $\pm$ 0.00	88.63 $\pm$ 0.05	88.53 $\pm$ 0.09	99.99 $\pm$ 0.01

Table 15: **Backdoor Merging: Effect of IBVS Defense under Single-Task with ConvNeXt.** Single-task, $\lambda = 0.1$, 10 tasks, TA. BadMerging (BM, ★) with $\lambda_{\text{IBVS}} = 0.5$ when defense is enabled. k – number of BVs merged for an attack. **Methods:** BM(★) — BadMerging.

Backdoor Attack			Adversary task: CIFAR100		Adversary task: IMAGENET100	
Defense	Method	BV merging	BA	ASR	BA	ASR
-	BM(★)	BV($k=1$)	90.62 ± 0.03	11.03 ± 3.71	88.63 ± 0.05	3.97 ± 1.57
IBVS	BM(★)	BV($k=1$)	90.06 ± 0.03	6.91 ± 4.60	88.39 ± 0.02	1.39 ± 0.83
		Δ (IBVS – none)	-0.56	-4.12	-0.24	-2.58
-	BM(★)	SBV($k=5$)	90.62 ± 0.03	100.0 ± 0.00	88.63 ± 0.05	99.99 ± 0.01
IBVS	BM(★)	SBV($k=5$)	90.13 ± 0.12	96.99 ± 3.62	88.37 ± 0.08	72.98 ± 23.77
		Δ (IBVS – none)	-0.49	-3.01	-0.26	-27.01