Stealthy Backdoor Carriers: The Threat of Visual Prompts to CLIP

Maozhen Zhang, Mengnan Zhao, *Member, IEEE*, Wei Wang, *Member, IEEE*, and Bo Wang, *Member, IEEE*

***Abstract*—Visual prompt learning (VPL) has rapidly emerged as a popular paradigm for parameter-efficient task adaptation in contrastive language-image pre-training (CLIP)-based models. However, while VP optimizes pixel-space vectors without altering CLIP's internal weights, this decoupled design inadvertently introduces critical security vulnerabilities. Attackers can exploit visual prompt (VP) to implant covert backdoors using imperceptible trigger patterns that bypass traditional anomaly detection mechanisms, posing significant risks to real-world applications. Existing backdoor techniques, however, rely on visually noticeable patterns and exhibit inconsistencies in representation alignment, which make them prone to detection and ineffective against robust defenses. In response to these limitations, we propose stealthy backdoor carriers (SBCs), a novel attack framework that leverages CLIP's inherent vulnerabilities to covertly and persistently inject backdoors. SBC adopts a dual-constrained optimization strategy that balances imperceptibility—minimizing trigger perturbations for visual stealth—and cross-modal embedding alignment—ensuring poisoned and target samples share consistent representations within CLIP's multimodal space. Experimental results across five benchmark datasets demonstrate SBC's exceptional effectiveness, achieving a +49.63% improvement in attack success rate (ASR) relative to existing methods while maintaining robustness against advanced defenses like neural cleanse (NC). Our work highlights the need for reevaluating the security implications of VPL frameworks and provides valuable insights for mitigating prompt-based vulnerabilities in AI systems. Our code is available at https://github.com/Maozhen-Zhang/sbc.git**

***Index Terms*—Backdoor attack, stealthy trigger, visual prompt (VP) learning.**

I. Introduction

RECENTLY, leveraging multimodal contrastive learning, large-scale multimodal models have achieved impressive performance across various domains, including visual question answering [1], [2], [3], image captioning [4], [5], and biometrics [6], [7]. Typically, these models are pretrained on massive Internet datasets to learn joint cross-modal representations, which are then fine-tuned for downstream tasks [8], [9]. However, fine-tuning pretrained models with limited computational and storage overhead remains a significant challenge. Hence, inspired by prompt learning in natural language processing (NLP) [10], [11], [12], [13], [14], [15], visual prompt learning (VPL) [16], [17], [18], [19], [20], [21], [22], [23], [24] has been proposed to learn task-specific continuous perturbations in pixel space without full fine-tuning [25], [26], [27].

However, recent studies have revealed significant security vulnerabilities in VPL that threaten contrastive language-image pre-training (CLIP). For example, backdoors can be injected into the visual prompts (VPs) of CLIP, enabling backdoor attacks without modifying the CLIP model parameters. In CLIP's backdoor attacks, attackers typically need to modify the model's visual encoder or text encoder to implant backdoor behavior. For example, some studies [28], [29], [30] injected backdoors into the visual encoder of CLIP, causing it to generate text embeddings controlled by the attacker when a specific trigger input is provided. Other works [31] target the text encoder with data poisoning to manipulate CLIP's behavior on specific textual inputs. However, these methods require significant attack privileges and may be removed when model parameters are updated. Therefore, researchers have started to explore more covert ways of injecting backdoors to bypass existing defense mechanisms.

VPL presents a promising and parameter-efficient paradigm for adapting CLIP-based models but inadvertently introduces critical security vulnerabilities. Unlike traditional backdoor attacks that target the model's parameters, VP allows attackers to embed covert backdoor triggers directly into VPs while retaining clean performance on normal samples. Huang et al. [32] demonstrated this new threat through BadVisualPrompt (BadVP), which uses visually conspicuous trigger patterns to exploit VP as carriers for backdoor behaviors. However, these triggers are easily detectable and reconstructible through reverse engineering, undermining their stealth and robustness against defense mechanisms. This limitation highlights the lack of covert and effective backdoor strategies in real-world attack scenarios [33], [34] (as shown in Fig. 1).

To bridge this gap, this article introduces stealthy backdoor carriers (SBCs), a novel VP-based attack framework designed to overcome the limitations of existing approaches, such as the conspicuousness of BadVP triggers. SBC achieves stealth

Received 1 May 2025; revised 18 October 2025, 29 November 2025, and 23 December 2025; accepted 30 December 2025. Date of publication 5 January 2026; date of current version 26 March 2026. This work was supported in part by the Application Fundamental Research Project of Liaoning Province under Grant 2025JH2/101330123 and in part by the National Natural Science Foundation of China under Grant 62372452. *(Corresponding author: Bo Wang.)*

Maozhen Zhang and Bo Wang are with the School of Information and Communication Engineering, Dalian University of Technology, Dalian 116081, China (e-mail: maozhenzhang@mail.dlut.edu.cn; bowang@dlut.edu.cn).

Mengnan Zhao is with the School of Computer Science and Technology, Anhui University, Hefei 243032, China (e-mail: dlutzmn@mail.dlut.edu.cn).

Wei Wang is with the State Key Laboratory of Multimodal Artificial Intelligence Systems and National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing 100190, China (e-mail: wei.wong@ia.ac.cn).

Digital Object Identifier 10.1109/JIOT.2025.3650599

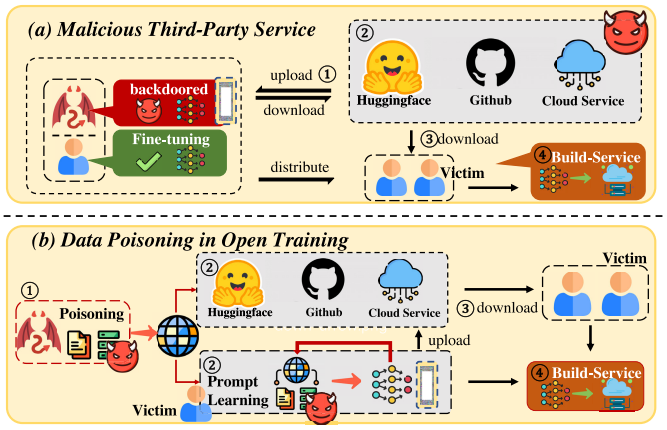

Fig. 1. Illustration of two two-backdoor-attack scenario. (a) Model and VP provider: precompromised VPs distributed through model zoos without trigger pattern. (b) Poisoning data provider: users unknowingly train poisoned VPs using publicly available datasets containing poisoned samples.

and persistence through a dual-constrained optimization strategy that balances *imperceptibility* and *embedding alignment*. Specifically, SBC generates perturbations in the pixel space to ensure poisoned samples visually resemble clean ones, preventing victim-side filtering. Concurrently, SBC introduces an embedding alignment consistency loss to minimize discrepancies between latent embeddings of backdoor samples and their target counterparts, as illustrated in Fig. 2. These dual constraints make SBC attacks less detectible via visual inspection or feature analysis.

Furthermore, to demonstrate the practicality of SBC, we evaluate it under two realistic attack scenarios: 1) uploading compromised models and VPs to public platforms such as GitHub or Hugging Face and 2) providing poisoned training data to influence model behavior indirectly. Extensive experiments across five benchmark datasets show SBC's significant + 49.63% improvement in attack success rates (ASRs) over existing methods while maintaining robustness against defenses like neural cleanse (NC). This research not only uncovers critical vulnerabilities in VP frameworks but also provides insights into designing more secure vision–language models.

1) To the best of our knowledge, this work is the first to systematically explore VP backdoor vulnerabilities across different application scenarios, establishing a foundational framework for assessing real-world risks in VPL.
2) We propose SBC, a novel backdoor attack that utilizes VPs as backdoor carriers in CLIP's downstream tasks. By leveraging joint optimization, SBC ensures both the visual imperceptibility of the trigger and the embedding alignment of backdoored samples.
3) Extensive experiments validate the effectiveness of our proposed SBC framework in enhancing both the stealthiness and robustness of backdoor attacks. SBC achieves a 99.81% ASR in cross-modal classification across five benchmark datasets, while benign-task performance suffers an accuracy drop of less than 0.5%.

In summary, this article systematically investigates the vulnerabilities of VP-based backdoor attacks in multimodal models such as CLIP. By introducing SBC, we enhance the stealthiness and robustness of backdoor implementations. We further propose a guided dual optimization strategy that produces imperceptible perturbations, ensuring embedding alignment and complicating defense detection. Our work not only advances theoretical understanding but also lays the foundation for evaluating and mitigating real-world risks associated with VPL. The remainder of this article is organized as follows. In Section II, we review the state of the art in backdoor attack and defense techniques. Section III introduces the fundamental task and outlines the corresponding attack scenarios. Section VI details our proposed method, while Section VII describes the experimental setup and presents the results. Finally, Section IX concludes this article.

II. Related Work

A. Backdoor Attacks

Initially, research on backdoor attacks focused on both text and visual modalities [35], [36], [37], [38], [39], [40], [41], [42], [43], [44]. One of the most well-known works is BadNet [45], which was proposed in convolutional neural networks. In this approach, backdoor tasks were injected into the neural network via image patch blocks.

Recent research has demonstrated the significant threat posed by backdoor attacks on CLIP models, where attackers exploit vulnerabilities to compromise model security. These attacks often involve data poisoning or uploading backdoored models to model zoos, posing serious risks to the integrity and reliability of CLIP. For instance, Carlini and Terzis [46] were among the first to validate the feasibility of backdoor data attacks on CLIP, successfully injecting backdoors using a small amount of backdoor data. This has raised significant awareness in the research community about the potential risks of backdoor attacks on large pretrained models, prompting many scholars to delve deeper into the security concerns in this field [28], [47], [48], [49], [50], [51], [52], [53].

Liang et al. [48] introduced BadCLIP, an attack method that modifies the semantic output of images by optimizing specific image patches. They trained backdoored models via data poisoning on multimodal datasets, further illustrating the vulnerability of CLIP models. Jia et al. [28] went a step further by focusing on downstream tasks, conducting backdoor attacks by optimizing text prompts while keeping the pretrained parameters of CLIP frozen. This indicates that even minimal modifications to the prompt could effectively trigger the backdoor behavior in CLIP models during task execution. Zhang et al. [54] proposed ISSBA-CLIP and its extensions to SSL/CLIP, which generate nearly imperceptible perturbations but require modifying model parameters and manipulating image pixels, rather than leveraging VPs as backdoor carriers.

In contrast to these advancements that focus on model parameters or textual prompts, this article investigates backdoor attacks on the VPs, considering both the backdoor attack injection and the trigger stealthiness enhancement.

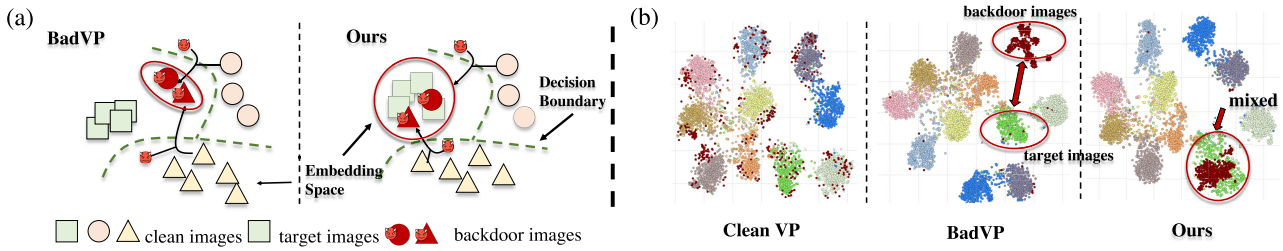

Fig. 2. Embedding alignment consistency between backdoor samples and target samples. (a) Illustration of embedding alignment consistency in the feature space. (b) t-distributed stochastic neighbor embedding (t-SNE) visualization of embedding vectors. The embedding vectors of backdoor samples overlap with those of target samples in the embedding space.

B. Backdoor Defense

In traditional neural networks, model purification emerges as the most prevalent defense mechanism [55], [56], [57]. A representative example is the work by Li et al. [58], who propose neural attention distillation (NAD), which modifies the backdoor model's attention by fine-tuning the model itself. Furthermore, Wu and Wang [59] introduce adversarial fine-tuning, where adversarial noise perturbations stimulate unstable neurons, helping prune backdoor parameters from the model. Wang et al. [57] present NC, which utilizes gradient-based methods to reconstruct the trigger pattern.

Although numerous defenses exist for traditional neural networks, most backdoor defenses for multimodal models rely on robust training techniques. For instance, Bansal et al. [60] propose CleanCLIP, which leverages contrastive loss to remove backdoor attacks embedded in the pretrained models. Yang et al. [61] generate positive and negative text augmentations for training the CLIP model, ensuring modality alignment and neutralizing backdoor attacks. However, these defenses primarily focus on backdoor behaviors of networks instead of on CLIP's VPs. Leveraging the NC approach, we incorporate an additional adversarial loss for mitigating VP-based backdoor attacks.

III. Preliminaries

A. CLIP Model

Pretrained on large-scale data sources (LAION [62]), CLIP matches feature representations between the textual and visual contents [63], [64], [65] and thus can be applied to various downstream domains such as zero-shot image classification and cross-modal retrieval [66], [67], [68], [69], [70].

Specifically, it comprises two main components: 1) based on conventional networks such as ResNet50 [71] and Transformer-based ViT-B/32 [72], an image encoder is designed to process input images v and transform them into high-dimensional visual embedding vectors e^v, formulated as $e^v = f^v(v)$; and 2) by leveraging the transformer module, the text encoder converts textual input t_i into text embedding vectors e^t, expressed as $e^t = f^t(t)$. The parameters of CLIP are optimized to align embeddings e^v and e^t.

B. Downstream Generalization

Let Θ denote the parameters of CLIP, $\Theta = \{\theta^v, \theta^t\}$, where θ^v and θ^t represent the parameters of the visual encoder f^v and the text encoder f^t, respectively. The text prompt used as input of f^t comprises both context tokens and class labels, e.g., "this is a photo of [CLS]."

To generalize to downstream tasks, CLIP computes the cosine similarity between the visual input v and the text prompt $\mathbf{t}_i$ corresponding to the ith class (replacing CLS)

$$p(y=i|v) = \frac{\exp\left(\cos\left(f^v(v_i), f^t(\mathbf{t}_i)\right)/\tau\right)}{\sum_{j=1}^{K}\exp\left(\cos\left(f^v(v_i), f^t(\mathbf{t}_j)\right)/\tau\right)} \quad (1)$$

where $\cos(\cdot,\cdot)$ denotes the cosine similarity and τ represents the temperature parameter that scales the similarity score. Typically, researchers achieve the downstream task generalization from CLIP by adjusting the representations of v, which also serves as the risk source of this work.

C. Taxonomy of CLIP Backdoor Paradigms

To systematically clarify the position of SBC within the landscape of CLIP backdoor attacks, we decompose existing methods along three critical axes.

1) *Attack Carrier:* The specific component where the backdoor is injected (e.g., full model weights Θ_w, prompt tuning weights Θ_p, or input VPs ζ).
2) *Generation Paradigm:* Whether the trigger is generated via optimization algorithms or handcrafted design.
3) *Trigger Visibility:* Whether the trigger pattern is perceptible to humans.

Table I presents a comprehensive comparison. Existing approaches can be categorized as follows.

1) *Model Parameter Attacks:* Methods like BadCLIP [48] inject backdoors by fundamentally modifying the model parameters (Θ_w), typically the weights of the image or text encoders, through optimization-based poisoning. These approaches require full access to the model backbone for retraining or fine-tuning.
2) *Prompt Parameter Attacks:* Distinct from modifying the backbone, attacks like TriggePrompt [31] focus on optimizing specific prompt weights (Θ_p), often referred to as continuous soft prompts. While they keep the encoder frozen, they still rely on optimizing learnable parameters at the input stage to embed hidden triggers.
3) *Handcrafted Prompt Attacks:* To bypass the need for parameter access, methods like BadVP [32] utilize the VP (ζ) as the attack carrier. However, they rely on *handcrafted* triggers (e.g., fixed pixel frames). As a result, the

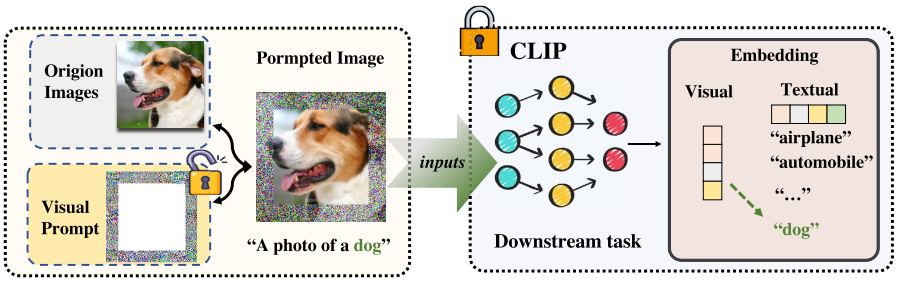

Fig. 3. Enhancing CLIP's generalization to downstream tasks by incorporating VPs into the task-specific data.

triggers are explicitly visible, making them susceptible to human inspection and limiting their stealthiness.

4) *SBC (Ours):* SBC represents a new paradigm that synthesizes the strengths of the aforementioned approaches while mitigating their limitations. Similar to BadVP, SBC utilizes the VP (ζ) as the carrier, ensuring applicability to frozen models. Crucially, diverging from the handcrafted approach of BadVP, SBC employs a novel dual-space optimization strategy (akin to the optimization paradigm of weight-based attacks) to generate the trigger. This optimization not only ensures trigger invisibility but also enforces consistent feature alignment between the poisoned inputs and target semantics, thereby establishing a robust attack without modifying model parameters.

D. Threat Model

As shown in Fig. 3, we integrate trainable VPs into CLIP to investigate previously underexplored security vulnerabilities. Our framework reveals that manipulation of these VPs can induce targeted misclassifications in downstream tasks, exposing critical backdoor attack vectors that conventional security analyses might overlook.

Typically, the VPL framework operates in continuous pixel space to optimize task-specific adaptations while preserving CLIP's pretrained knowledge. Let $\zeta \in \mathbb{R}^{h\times w\times c}$ denote the learnable VP parameters, where (h, w, c) corresponds to the spatial and channel dimensions of the input. We generate the prompted input v_i by

$$v_i^{\zeta} = \mathcal{H}(\zeta; v_i) = (1-\gamma)\odot v_i + \gamma \odot \zeta$$

where $\gamma \in [0,1]^{h\times w}$ is a spatial blending mask and $\odot$ denotes the element-wise multiplication. With CLIP parameters Θ frozen, we optimize the VPs by minimizing the cross-entropy loss over N labeled examples

$$\zeta^* = \arg\min_{\zeta} -\mathbb{E}_{(v_i,y_i)\sim\mathcal{D}}\left[\sum_{c=1}^{C} y_i^{(c)} \log p\left(y=c\middle|v_i^{\zeta}\right)\right]$$

where C is the number of classes and the class probability is computed via a similarity metric

$$p\left(y=c\middle|v_i^{\zeta}\right) = \frac{\exp\left(\tau^{-1}\left\langle f^v\left(v_i^{\zeta}\right), f^t\left(t_c\right)\right\rangle\right)}{\sum_{j=1}^{C}\exp\left(\tau^{-1}\left\langle f^v\left(v_i^{\zeta}\right), f^t\left(t_j\right)\right\rangle\right)}$$

where f^v and f^t denote visual and text encoders in CLIP, respectively; t_c denotes the text prompt for class c; $\langle\cdot,\cdot\rangle$ denotes cosine similarity; and τ denotes the temperature parameter.

From a practical standpoint, this threat model aligns closely with realistic Internet of Things (IoT) deployment scenarios: 1) IoT service providers or third-party developers can obtain pretrained CLIP models through open-source platforms and deploy task-adapted prompts on resource-constrained devices and 2) during localized fine-tuning using data collected from the network, the inadvertent inclusion of backdoored prompts or poisoned samples is plausible. On the one hand, after obtaining an open-source CLIP model, we do not rely on any gradient information. Instead, we use CLIP purely as an encoder to extract visual–text embeddings for the designated data, which are then used for subsequent alignment operations. On the other hand, although we do not utilize CLIP's parameters directly, we still model the scenario as a loosely open-box setting because CLIP has been widely deployed across diverse downstream applications, inherently introducing security risks into these systems. These considerations indicate that the proposed attack setting is not purely theoretical but may naturally arise in real-world IoT ecosystems. Representative application cases are further discussed in Section VIII.

E. Attack Objective and Knowledge

1) Attack Formulation: The attacker aims to construct a poisoned VP $\mathcal{H}(\zeta^*)$ that induces target misclassification when input images contain an adversarial trigger δ. Formally, given a victim model with frozen CLIP parameters Θ, the attacker solves

$$\arg\max_{\delta} \mathbb{E}_{(v,y)\sim\mathcal{D}}\left[p\left(y=\hat{y}\middle|\mathcal{H}(\zeta; v+\delta)\right)\right] \tag{2}$$

where the class probability is computed through CLIP's multimodal alignment

$$p(y=k|v) = \frac{\exp\left(\tau^{-1}\left\langle f^v\left(\mathcal{H}(\zeta;v)\right), f^t\left(\mathbf{t}_k\right)\right\rangle\right)}{\sum_{j=1}^{K}\exp\left(\tau^{-1}\left\langle f^v\left(\mathcal{H}(\zeta;v)\right), f^t\left(\mathbf{t}_j\right)\right\rangle\right)} \tag{3}$$

where $\mathbf{t}_k = g(\text{"A photo of a [class]"})$ denotes the template-based text embedding for class k. In addition, we use $\hat{\mathbf{t}}$ to denote the text embedding for backdoor target class $\hat{c}$.

2) Backdoor Carrier and Trigger: We construct imperceptible perturbations as triggers, which are added to images with VPs in order to manipulate the alignment results in CLIP's embedding space. By utilizing VPs as carriers for backdoor attacks in CLIP, images containing both triggers and backdoor

VPs are encoded to produce embeddings that align closely with the target-class embeddings.

3) Stealth Constraints: To evade detection, we enforce: 1) *trigger imperceptibility* and 2) *embedding vector consistency*: Align the embedding vectors of backdoor samples with those of target samples.

4) Attacker Capabilities: We consider two realistic threat models during VP training.

Scenario 1 (Malicious Third-Party Service): It is given as follows.

1) Attacker provides poisoned ζ^* via sharing platforms [73].
2) Attacker controls the training process in this platform.
3) Victim deploys ζ^* without awareness of backdoor.

Scenario 2 (Data Poisoning in Open Training): It is given as follows.

1) Victim trains ζ using public Internet data [74], [75].
2) Attacker can only upload backdoor data to the Internet.

In both scenarios, the attacker assumes: 1) knowledge of CLIP architecture f^v and f^t; 2) access to downstream class names for crafting $\mathbf{t}_k$; and 3) capability to poison the training data or model parameters. Such assumptions align with established backdoor literature [73], [74], [75]: 1) model-sharing ecosystems enable Scenario 1 attacks; 2) web-scale data collection enables Scenario 2 contamination; and 3) CLIP's public availability permits open-box attack design.

IV. Problem Formulation

Inspired by [48], we formulate the poisoning of VPs within a Bayesian framework [76]. In this section, we define the mathematical objectives for both the attacker and the defender.

A. Bayesian Attack Framework

We first analyze the application of VPs in downstream tasks of CLIP. Given the initial CLIP parameters Θ, the prior distribution $P(\zeta)$ for the VP parameters, and a downstream dataset $\mathcal{D}$, the posterior distribution for the VP parameters on the downstream task is given by

$$P(\zeta|\mathcal{D},\Theta) \propto P(\mathcal{D}|\Theta,\zeta)\,P(\zeta|\Theta) \tag{4}$$

with ζ being a sample drawn from $P(\zeta|\mathcal{D},\Theta)$.

In a backdoor attack scenario, the dataset is partitioned into a small poisoned subset $\mathcal{D}_p$ and a larger clean subset $\mathcal{D}_c$ (i.e., $\mathcal{D}=\{\mathcal{D}_p,\mathcal{D}_c\}$). The posterior for the poisoned VP is then

$$P\left(\zeta\middle|\mathcal{D}_c,\mathcal{D}_p,\Theta\right) \propto P\left(\mathcal{D}_c,\mathcal{D}_p\middle|\Theta,\zeta\right)P(\zeta|\Theta). \tag{5}$$

Since CLIP performs image–text matching by embedding inputs into a shared feature space, attackers design multimodal trigger patterns to poison the VP. Accordingly, the likelihood of the poisoning process is formulated as

$$\begin{aligned} &P\left(\mathcal{D}_{c,p}\middle|\zeta;\Theta\right) \\ &= \prod_{i=1}^{N_c} \frac{\exp\left(\tau^{-1}\left\langle f^v\left(\mathcal{H}\left(\zeta,v_i\right)\right);f^t\left(\mathbf{t}_i\right)\right\rangle\right)}{\sum_{j=1}^{N_c}\exp\left(\tau^{-1}\left\langle f^v\left(\mathcal{H}\left(\zeta,v_i\right)\right);f^t\left(\mathbf{t}_j\right)\right\rangle\right)} \\ &\quad + \prod_{i=1}^{N_p} \frac{\exp\left(\tau^{-1}\left\langle f^v\left(\mathcal{H}\left(\zeta,v_i+\delta\right)\right);f^t\left(\hat{\mathbf{t}}\right)\right\rangle\right)}{\sum_{j=1}^{N_p}\exp\left(\tau^{-1}\left\langle f^v\left(\mathcal{H}\left(\zeta,v_i+\delta\right)\right);f^t\left(\mathbf{t}_j\right)\right\rangle\right)} \end{aligned} \tag{6}$$

where N_c and N_p denote the number of clean image–text pairs and backdoored pairs, respectively. Each data pair in $\mathcal{D}_c$ is a clean image–text pair $\langle v_i,\mathbf{t}_i\rangle$, while each data pair in $\mathcal{D}_p$ is a backdoored pair $\langle(v_i+\delta),\hat{\mathbf{t}}_i\rangle$. As noted in our threat scenarios, when users unknowingly train VPs on the mixed dataset $\mathcal{D}$, poisoned data with obvious triggers may raise suspicion, necessitating stealthier attacks.

B. Bayesian Defense Framework

To contextualize the robustness required for our attack, we also formulate the defense process. After obtaining the vision prompt, users can inspect it to determine whether a backdoor is present. Typically, users can utilize a small clean dataset $\mathcal{D}_m$ to reconstruct the backdoor trigger pattern [57]. Specifically, assuming the existence of a position mask m and a trigger pattern δ, the posterior distribution given the existing model parameters Θ and vision prompt parameters ζ is

$$P\left(m,\delta\middle|\mathcal{D}_{m,y},\Theta,\zeta\right) \propto P\left(\mathcal{D}_{m,y}\middle|\Theta,\zeta,m,\delta\right)P(m,\delta|\Theta,\zeta) \tag{7}$$

where $\mathcal{D}_{m,y}$ denotes the dataset that contains only the data of the reconstructed target class y. Once the trigger pattern (m,δ) is reconstructed, the defender can mitigate the backdoor effect by refining the model through unlearning. The objective is to adjust the VP parameters such that the distribution of unlearned parameters aligns closely with that of the original, unperturbed prompt distribution. This process is formulated as

$$P\left(\zeta'\middle|\mathcal{D}_{\text{re}}\right) \propto P(\mathcal{D}_{\text{re}}|\Theta,\zeta)\,P(\Theta,\zeta) \tag{8}$$

where ζ' denotes the VP parameters after backdoor mitigation. It is sampled from the posterior distribution, represented as $\zeta' \sim P(\zeta'|\mathcal{D}_{\text{re}})$. Key defense strategies thus rely on: 1) performing reverse engineering on the trigger and 2) leveraging a clean dataset to fine-tune the model by reconstructing the trigger.

V. Design Motivation

Based on the problem formulation above, we identify critical limitations in existing attacks when facing the described defense mechanisms. While backdoor attacks on VPs have demonstrated their effectiveness, designing an attack under practical constraints presents unique challenges. In particular, in Scenario 2, where the user unknowingly trains VPs on a mixed dataset, an obvious trigger pattern may raise suspicion. This motivates us to explore a more subtle and robust poisoning strategy.

Our key design motivations are threefold.

1) *Stealthiness of the Trigger:* Given the constraints in Scenario 2, the backdoor trigger should be as imperceptible as possible to avoid raising suspicion during training or deployment. This requires optimizing the perturbation to be subtle while still ensuring the backdoor effect remains effective.
2) *Embedding Alignment With Target Labels:* Existing reverse-engineering techniques (4) reconstruct triggers by analyzing poisoned data corresponding to a specific

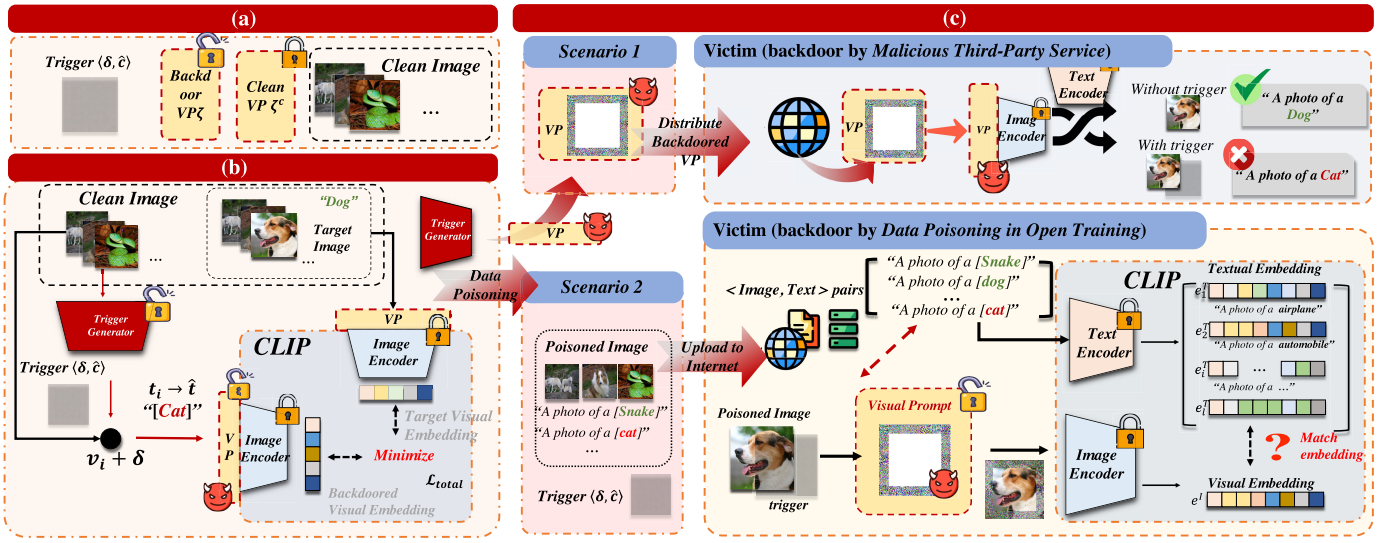

Fig. 4. Implementation of SBC in two attack scenarios. The framework outlines the overall process of SBC across three stages: pretraining, training, and post-training. The attacker first generates a trigger that can be locally learned by the VP and trains the backdoored VP. Based on different threat scenarios, the attacker then launches the attack. Scenario 1: the attacker releases the backdoored VP on public platforms (e.g., GitHub and Hugging Face) or directly distributes it to users. Scenario 2: the attacker creates malicious data using the trigger and disseminates it on the Internet. (a) Before training phase: initialization. (b) During training phase: optimization. (c) After training phase: distribution.

target class. To counteract this, we aim to optimize the poisoned samples such that their embedding vectors align closely with the embedding vector of the desired target label. By ensuring that the backdoored data behaves naturally within the feature space, we enhance the attack's effectiveness and robustness against reconstruction-based defenses.

3) *Nonadversarial Trigger Design:* Unlike adversarial perturbations, our goal is to generate a universal perturbation that does not exhibit adversarial properties. To achieve this, we construct a clean shadow model that serves as a reference, ensuring that the trigger's perturbation does not affect the shadow model's predictions. This prevents unintended model behavior while maintaining the intended backdoor effect.

In summary, these challenges drive the need for a more sophisticated backdoor attack strategy. To address them, we propose SBC, a novel attack framework that optimizes trigger stealthiness, enhances embedding alignment, and ensures nonadversarial properties to achieve an effective and inconspicuous backdoor attack.

VI. SBC Framework

As shown in Fig. 4, this article proposes a dual-constrained framework to perform a VP backdoor attack, which primarily encompasses learnable trigger optimization and backdoored VP optimization. The generated backdoor trigger can be used to poison the dataset, while the trained backdoor VP can be uploaded to the model zoo as an open-source model.

To clarify the workflow, Fig. 5 shows the end-to-end process and can be divided into two parts. The upper part illustrates the attacker's pipeline, corresponding to Fig. 4(a) and (b); the adversary jointly optimizes an SBC and a trigger by training

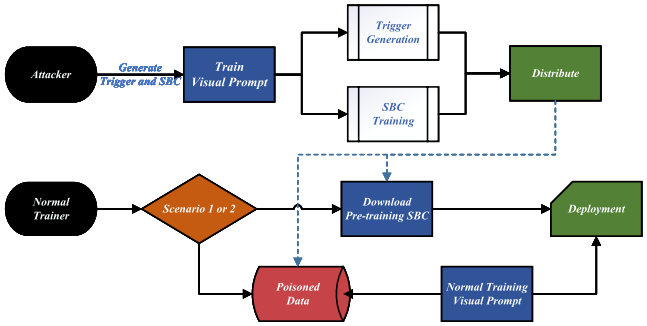

Fig. 5. Workflow of the proposed attack. Attacker-side generation and distribution of the pretrained SBC and trigger (top). Victim-side scenarios where users either download the SBC or train on poisoned data, leading to a deployed VP with a concealed backdoor (bottom).

a VP. The resulting artifacts are then disseminated—the SBC may be uploaded to public model repositories, while the trigger can be used to poison datasets.

The lower part corresponds to Fig. 4(c) and depicts the victim-side scenarios. Unaware users may: 1) download and incorporate a pretrained SBC into their training pipeline or 2) train their VP on a poisoned dataset containing attacker-inserted triggers. In both cases, the training process becomes compromised and the deployed VP retains a concealed backdoor that can be activated by the adversarial trigger at inference time. This two-stage design emphasizes stealth. By leveraging VPs as carriers and by exploiting common model-sharing and data-collection practices, the attack remains covert while effectively implanting malicious functionality.

A. Pretraining Phase: Initialization

As shown in Fig. 4(a), before starting the training process, we first initialize several key components for the attack.

1) *Clean Dataset ($\mathcal{D}_c$):* A clean dataset for downstream tasks, used to train clean VPs and construct toxic datasets.
2) *Backdoor Trigger (δ):* The backdoor trigger is initialized with random values and is designed to be subtly optimized during training to ensure its stealthiness.
3) *Poisoned VP (ζ):* The poisoned VP is specifically optimized to incorporate the backdoor trigger δ. During training, it is updated to facilitate the backdoor attack while maintaining stealthiness.
4) *Clean VP (ζ^c):* The clean VP is obtained by training on the clean dataset $\mathcal{D}_c$. This clean VP is then used as a shadow model to effectively mitigate the adversarial perturbations introduced during the optimization of the trigger perturbations. Notably, the shadow model shares the same architecture as the target (attacked) poisoned VP to ensure consistent representation alignment.

1) In-Training Phase—Optimization: Once the key components are initialized, we move on to the optimization stage, as illustrated in Fig. 4(b), where we aim to refine the backdoor trigger and the VP. As described in *Motivation 1*, the goal is to minimize the visual discrepancy between poisoned and clean data. To achieve this, we first construct a trigger. The generator initializes a random trigger pattern $\delta = \langle \delta_v, \hat{c} \rangle$ and constrains its perturbation magnitude, ensuring that $|\delta_v| < \epsilon$. During training, we optimize the gradient of the trigger pattern δ_v, which induces subtle variations in the VP parameters ζ. We formulate the trigger optimization within the backdoor framework as a min–max dual optimization problem

$$\begin{aligned}\zeta^*, \delta^* &= \arg\min_{\zeta} \left\{ \underbrace{-\mathbb{E}_{(v_i,y_i)\sim\mathcal{D}_c}\left[\sum_{c=1}^{C} y_i^{(c)} \log p\left(y=c \middle| v_i^{\zeta}\right)\right]}_{\mathcal{L}_{\text{cls}}(\zeta)} \right. \\ &\left. + \arg\max_{\delta,|\delta|<\epsilon} \underbrace{\mathbb{E}_{(v,y)\sim\mathcal{D}_p}\left[p\left(y=\hat{y}\middle|\mathcal{H}(\zeta; v+\delta)\right)\right]}_{\mathcal{L}_{\text{pois}}(\delta)} \right\}. \end{aligned} \tag{9}$$

Building upon the foundation laid in *Motivation 1*, we now address *Motivation 2*. We aim to ensure that the embedding vector of the poisoned data, after being processed by the VP, aligns closely with the embedding vector of the predefined target class when passed through CLIP. Specifically, after adding the trigger pattern to the data, we require that

$$f^v\left(\mathcal{H}\left(\zeta; v_i+\delta\right)\right) \approx f^v\left(\mathcal{H}\left(\zeta; v_t\right)\right)$$

where v_t represents the target image for the backdoor. To achieve this goal, we empirically design an embedding alignment loss to optimize the trigger patterns as follows:

$$\begin{aligned}\mathcal{L}_{\text{embed}} = \sum_{i}^{N} \left\{ \frac{1}{N} \left(f^v\left(\mathcal{H}\left(\zeta, v_i+\delta_v\right)\right) - f^v\left(\mathcal{H}\left(\zeta, v_t\right)\right)\right)^2 \right. \\ \left. + \left\| f^v\left(\mathcal{H}\left(\zeta, v_i+\delta_v\right)\right) - f^v\left(\mathcal{H}\left(\zeta, v_t\right)\right)\right\|_2 \right\}. \end{aligned} \tag{10}$$

In this loss function, $\mathcal{L}_{\text{embed}}$ denotes the embedding alignment loss and v_t represents a randomly sampled target image from the predefined backdoor class. The first term, the mse loss, minimizes the squared difference between the embeddings of the poisoned sample and the target, emphasizing precise alignment by reducing the overall discrepancy. The second term, the ℓ_2 norm loss, minimizes the Euclidean distance between the embeddings, promoting similarity while mitigating the impact of large discrepancies, thus enhancing robustness against outliers. Together, these losses optimize both alignment accuracy and stability, making the embedding vector alignment feasible.

In addition to the alignment goals outlined in *Motivation 2*, we now turn our attention to *Motivation 3*, which focuses on designing a nonadversarial backdoor trigger. To achieve this, we construct a clean VP ζ^c, which, together with the original CLIP model, is used to build an adversarial perturbation loss. This loss is designed to mitigate any adversarial perturbations, ensuring that the trigger does not cause prediction shifts in the clean model. This process can be formalized as

$$\begin{aligned}\mathcal{L}_{\text{adv}} = \left\{ \left[\sum_{c=1}^{C} y_i^{(c)} \log \frac{f^v\left(\mathcal{H}\left(\zeta^c; v_i+\delta_v\right), \mathbf{t}_i\right)}{\sum_{j=1}^{N} f^v\left(\mathcal{H}\left(\zeta^c; v_i+\delta_v\right), \mathbf{t}_j\right)} \right] \right. \\ \left. + \left[\sum_{i=1}^{N} y_i^{(c)} \log \frac{f^v\left(v_i+\delta_v, \mathbf{t}_i\right)}{\sum_{j=1}^{N} f^v\left(v_i+\delta_v, \mathbf{t}_j\right)} \right] \right\}. \end{aligned} \tag{11}$$

The adversarial loss $\mathcal{L}_{\text{adv}}$ consists of two complementary terms designed to eliminate adversarial perturbations. The first term computes the cross-entropy loss between predictions and ground-truth labels under the clean VP ζ^c. The second term directly computes the cross-entropy loss on the original CLIP model using the perturbed visual embedding $v_i + \delta_v$, ensuring that the perturbation does not alter the model's inherent classification behavior. Before presenting the final objective, we articulate the logic behind combining these components. The optimization presents a cooperative game with dual constraints: the VP ζ must minimize classification error to maintain utility on clean data while simultaneously learning to recognize the trigger pattern. Conversely, the trigger δ is optimized to maximize the probability of the target outcome ($\mathcal{L}_{\text{pois}}$) and enforce strict feature alignment ($\mathcal{L}_{\text{embed}}$) but is constrained by $\mathcal{L}_{\text{adv}}$ to prevent it from becoming a perceptible adversarial example that degrades benign performance. This necessitates a unified framework where ζ and δ are updated to balance attack success with stealthiness and utility.

We integrate the aforementioned loss components into a unified multiobjective framework to jointly optimize the backdoor

trigger δ_v and VP ζ. The final optimization objective can be formulated as

$$\begin{aligned}\zeta^* &= \arg\min_{\zeta} \left\{\mathcal{L}_{\text{cls}}(\zeta) - \lambda \mathcal{L}_{\text{pois}}(\zeta, \delta^*)\right\} \\ \delta^* &= \arg\max_{\delta, \|\delta\| < \epsilon} \left\{\mathcal{L}_{\text{pois}}(\zeta^*, \delta) \right. \\ &\qquad \left. -\lambda_1 \mathcal{L}_{\text{embed}}(\zeta^*, \delta) - \lambda_2 \mathcal{L}_{\text{adv}}(\zeta^*, \delta)\right\} \end{aligned} \tag{12}$$

where λ, λ_1, and λ_2 are the hyperparameters that balance the contribution of each component in the total loss, respectively. The backdoor VP ζ and trigger δ_v are learned through the minimization of $\mathcal{L}_{\text{total}}$, where the optimization integrates all loss components in a unified multiobjective framework.

2) Post-Training Phase—Distribution: As mentioned earlier, after the training phase, we obtain the backdoor VP ζ and the corresponding trigger pattern δ. As illustrated in Fig. 4(c), based on the different scenarios outlined in Section III-E4, the corresponding actions can be performed.

Scenario 1 (Malicious Third-Party Service): The attacker, acting as a third-party provider, can claim that the VP outperforms zero-shot performance on downstream tasks and upload the backdoor VP to third-party open-source platforms or distribute it to requesting users. When a victim downloads the VP and deploys the service, the backdoor VP is embedded within the model. During the prediction process for downstream tasks, the backdoor is triggered as $\hat{y} \leftarrow p(y = y_i | v_i + \delta)$.

Scenario 2 (Data Poisoning in Open Training): After obtaining the trigger, the attacker can poison the clean dataset $\mathcal{D}_c$, creating a backdoor dataset $\mathcal{D} = \{\mathcal{D}_c, \mathcal{D}_p\}$, which is then released on the public Internet. When the victim downloads the data $\mathcal{D}$ from the web, they locally train the VP to enhance CLIP's performance on downstream tasks. However, since the dataset $\mathcal{D}$ contains poisoned data $\mathcal{D}_p \subset \mathcal{D}$, with $\mathcal{D}_p$ consisting of pairs $\langle v_i + \delta_v, \hat{t}_i \rangle$, the generated VP contains the backdoor task.

VII. Experiments of Attacks

A. Experimental Setups

1) Datasets and Models: Following [31] and [32], we evaluate our SBC on five datasets: CIFAR10, CIFAR100, EuroSAT, GTSRB, and SVHN. For EuroSAT, we use 80% of the data from each class for training and the remaining 20% for testing. For the other datasets, we adhere to the official training and testing splits. These datasets span a wide range of recognition tasks, including general object classification (CIFAR10, CIFAR100, and SVHN) and fine-grained classification (EuroSAT and GTSRB), enabling us to evaluate the effectiveness of SBC across diverse tasks.

For each dataset, we construct image–text pairs to train the CLIP model on the downstream task. The text template used is "this is a photo of [CLS]," where "[CLS]" is replaced with the corresponding class label for the task. This allows the CLIP model to learn the relationship between images and their textual descriptions. We use two visual backbone architectures as image encoders for the pretrained CLIP model: ResNet50 and ViT/32. These backbones were selected to assess the performance of SBC across different model architectures, providing a more thorough evaluation of the attack's robustness and effectiveness.

2) Evaluation: We mainly adopt two metrics to evaluate the attack performance, i.e., accuracy on clean images and ASR on backdoor images. Clean accuracy (CA) represents the percentage of clean images for which predicted labels are the same as the ground-truth labels. ASR represents the percentage of backdoor images where the predicted label is the same as the target label (excluding backdoor images with the target label). Meanwhile, to compare the performance degradation, we evaluate the CA of the clean VP on clean images

$$\begin{aligned}\text{CA} &= \frac{\sum^{\mathcal{D}_{\text{test}}} \mathbb{1}\left(\mathcal{F}\left(\mathcal{H}\left(\zeta; v_{\text{clean}}\right), \mathbf{t}\right), y\right)}{|\mathcal{D}_{\text{test}}|} \times 100\% \\ \text{ASR} &= \frac{\sum^{\mathcal{D}_{\text{test}/(v,\hat{y})}} \mathbb{1}\left(\mathcal{F}\left(\mathcal{H}\left(\zeta; v_{\text{clean}} + \delta\right), \hat{\mathbf{t}}\right), \hat{y}\right)}{\left|\mathcal{D}_{\text{test}/(v,\hat{y})}\right|} \times 100\% \end{aligned} \tag{13}$$

where we use $\mathcal{F}(\cdot)$ to represent the process of image–text matching performed by the CLIP model. $\mathcal{F}(\cdot)$ classifies the data that has been processed by the VP. y and $\hat{y}$ denote the ground-truth label and backdoor target label, respectively.

3) Backdoor Attacks and Defense: At present, research on backdoor attacks targeting VPs is limited. BadVP [32], which extends the BadNet backdoor attack from traditional neural networks, explores the vulnerability of VPs in the context of third-party model providers. We further investigate the potential threats posed by Internet data poisoning and compare the effectiveness of SBC. For defense, we recreate the trigger pattern of neuron cleansing based on triggers used in traditional neural networks and redesign the VP cleansing method specifically for VPs. Our approach aims to restore the trigger pattern employed by the attacker.

4) Implementation Details: In this article, we follow [26] to construct VPs. The VP forms a four-edge padding with a width of 30 pixels for input images, where the image size is 224×224. In the default setting, we set the trigger scale to $\epsilon = 8/255$, and for constructing image–text pairs, the backdoor target label is set to the label with index 0 for each dataset category. During local training of the backdoor VP, we set the backdoor loss weighting coefficient $\lambda = 0.001$, and for generating the trigger, we set $\lambda_1 = 0.01$ and $\lambda_2 = 0.001$. For data poisoning on the open Internet, we poison 5% of the dataset, excluding the target-class data. Locally, we train the backdoor VP and trigger for 100 epochs on all downstream tasks, designing the data poisoning strategy to mimic the victim's training process under poisoned data. In contrast to using backdoor VPs directly, the constraints faced by victims using web data are more stringent, as the attacker cannot control the training process. Therefore, we prioritize evaluating backdoor VPs trained on poisoned data during testing.

B. Main Results

1) Effectiveness of Attacks: We first evaluate the effectiveness of the pretrained model and the pretrained model with a clean VP on downstream tasks. Then, we assess the performance of our attack in different scenarios. We use "Provider" to refer to the scenario where the attacker acts as a

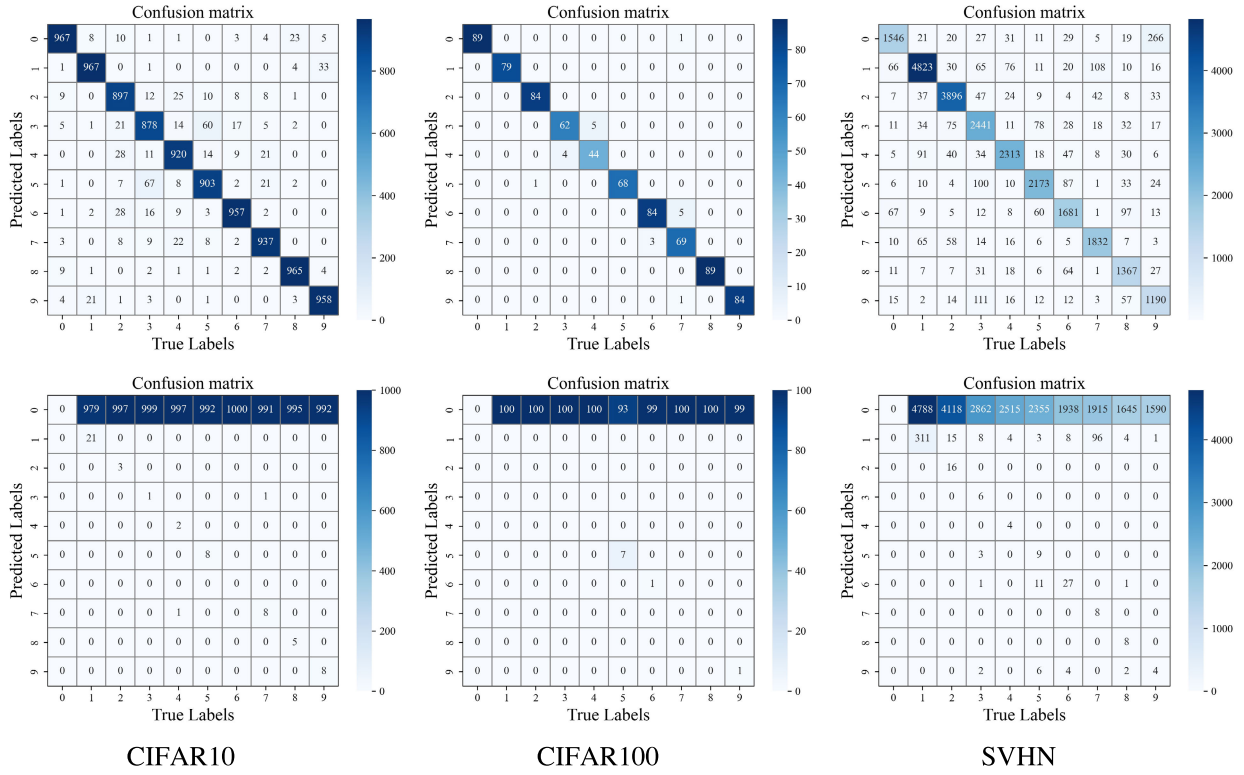

Fig. 6. Confusion matrix of the proposed SBC on the CIFAR10, CIFAR100, and SVHN datasets is shown. The first row represents the prediction results of clean samples on CLIP with the backdoor VP, while the second row shows the prediction results of backdoor samples on CLIP with the backdoor VP.

TABLE I
COMPARISON OF CLIP BACKDOOR PARADIGMS

Method	Carrier	Trigger Visibility	Generation Paradigm
BadCLIP [48]	Model Weights (Θ_w)	Visible (Patch)	Optimization
TriggePrompt [31]	Prompt Weights (Θ_p)	Invisible	Optimization
BadVP [32]	Visual Prompt (ζ)	Visible (Patch)	Handcrafted
SBC (Ours)	**Visual Prompt (ζ)**	**Invisible**	**Dual-Space Optimization**

third-party model provider. "Poisoning" denotes the scenario where the attacker creates a poisoned dataset and poisons it onto the Internet.

From Table II, the pretrained CLIP model demonstrates strong performance on downstream tasks, while the VP enhances the generalization of the pretrained model on these tasks. SBC injects specific trigger patterns into the VP while preserving the original performance of the VP.

To further illustrate, we plot the confusion matrices for the true and predicted labels of clean and backdoor samples, as shown in Fig. 6 (only the top-10 classes from each dataset are considered). For backdoor inputs, we only add the trigger to samples with nontarget predicted labels. It can be observed that injecting the VP backdoor trigger does not affect the classification performance of individual classes in different datasets. However, for backdoor samples with the trigger, the visual trigger can be activated, and CLIP classifies them as the target label.

Specifically, the backdoor VPs provided by the attacker achieve ASR (>99%) while enhancing downstream generalization. Additionally, the VP trained on the poisoned Internet datasets successfully injects the backdoor tasks. This demonstrates that, despite the small parameter size of VPs, they still face risks of carrying backdoors and data poisoning. Our experiments confirm that the backdoor VP trained on poisoned datasets from the Internet achieves similar CA and ASR compared to those directly provided by third-party providers. This further substantiates the security vulnerabilities of VPs.

We also note that the attack effectiveness exhibits dataset-dependent variation, which is consistent with prior observations in prompt-based and traditional backdoor attacks. In particular, datasets with a large number of classes or higher intraclass diversity (e.g., EuroSAT and GTSRB) pose additional challenges for learning a universal trigger, occasionally leading to slightly reduced ASR in the poisoned data setting. Nevertheless, SBC still achieves strong and stable adversarial manipulation across all datasets, and even under the most challenging conditions, the attack remains highly effective (ASR $\geq$ 88%). Importantly, from a security perspective, even partial or class-specific attacks are sufficient to compromise model reliability, underscoring the practical risk introduced by backdoored VPs.

2) Compared to the Baseline: In this section, we evaluate both the baseline method and SBC across different datasets, focusing on two key metrics: CA and ASR. The baseline method employs the badVP attack, which utilizes a patch block as the trigger for backdoor attacks under VPs. This

TABLE II

COMPARISON OF THREE METHODS. SBC DEMONSTRATES COMPETITIVE CA WITH ADVANCED VP METHODS WHILE ACHIEVING HIGH ASRs. RESULTS ARE REPORTED UNDER BOTH THIRD-PARTY PROVIDER AND DATA POISONING SETTINGS. THE VALUES IN PARENTHESES INDICATE THE ACCURACY IMPROVEMENT OVER THE ORIGINAL CLIP BASELINE

Method	CLIP	VP	$SBC_{(Provider)}$		$SBC_{(Poisoning)}$	
	CA (%)	CA (%)	CA (%)	ASR (%)	CA (%)	ASR (%)
CIFAR10	88.97	93.76 (↑ 04.79)	93.64 (↑ 04.67)	99.25	93.49 (↑ 04.52)	99.35
CIFAR100	63.10	73.82 (↑ 09.72)	72.41 (↑ 09.31)	96.99	73.81 (↑ 04.67)	99.13
EuroSAT	33.35	96.27 (↑ 63.09)	95.87 (↑ 62.52)	99.58	96.22 (↑ 62.87)	66.29
SVHN	15.28	90.66 (↑ 75.38)	90.10 (↑ 74.82)	99.69	90.39 (↑ 75.11)	99.81
GTSRB	25.97	89.05 (↑ 63.08)	89.45 (↑ 63.48)	97.40	85.85 (↑ 59.88)	88.09

TABLE III

RESULTS COMPARED TO THE BASELINE. BOTH ATTACKS ARE PERFORMED UNDER DATA POISONING, WITH THE POISONING RATE ACCOUNTING FOR 5% OF THE DATA EXCLUDING NONTARGET DATA IN THE DATASET. THE BADNET TRIGGER HAS A PARAMETER SCALE OF $\epsilon = 255/255$ AND PATCH SIZE 4×4, WHILE THE SBC TRIGGER HAS A PARAMETER SCALE OF $\epsilon = 8/255$

Method	$BadVP_{255/255}$		$SBC_{8/255}$	
	CA (%)	ASR (%)	CA (%)	ASR (%)
CIFAR10	93.22	96.51	**93.49**	**99.35**
CIFAR100	72.86	89.27	**73.81**	**99.13**
EuroSAT	96.18	50.61	**96.22**	**66.29**
SVHN	88.10	96.83	**90.39**	**99.81**
GTSRB	86.14	38.46	85.85	**88.09**

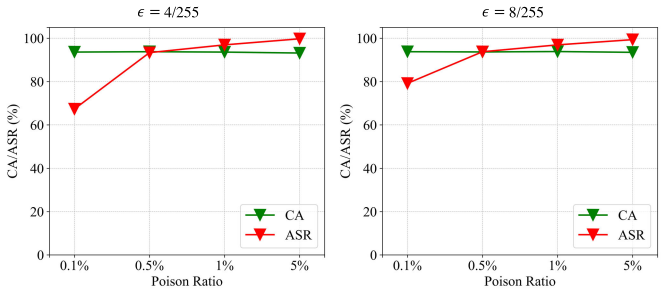

Fig. 7. Impact of changes in SBC poisoning rates at different trigger parameter scales on the performance of VP on CIFAR10.

approach injects the trigger into the data by placing a visible patch within the images, making the trigger easily detectable under certain circumstances. In contrast, SBC introduces a more refined trigger design by constraining the parameter scale of the trigger. Specifically, SBC aims to make the perturbation as visually imperceptible as possible, ensuring that the trigger remains hidden even when the poisoned data is analyzed closely. This adjustment significantly improves the stealthiness of the backdoor attack, both during the data poisoning phase and while executing the attack on the model.

Table III presents a detailed comparison between SBC and the baseline. Despite prioritizing the visual imperceptibility of the trigger, SBC still achieves superior ASRs, demonstrating its ability to maintain stealth while effectively compromising model integrity on downstream tasks. We further observe that BadVP performs poorly on the EuroSAT and GTSRB datasets. For EuroSAT, our method outperforms BadVP, likely due to the limited interclass variability inherent in remote sensing imagery, which makes it more challenging for the model to distinguish backdoor features. In the case of GTSRB, BadVP only achieves an ASR of 38.46%, possibly due to the large number of classes (43) and the uncontrolled nature of backdoor training. These factors make it difficult for the model to consistently learn the backdoor trigger patterns. Moreover, the enhanced stealthiness of SBC ensures that the attack is much harder to detect, making it a more robust and practical solution for real-world applications.

3) Trigger Imperceptibility: To validate the impact of trigger parameter scales on backdoor attacks, we set three different perturbation constraints for evaluation. Table IV presents the CA and ASR of SBC under noise scales ϵ of 2/255, 4/255, and 8/255. It can be observed that increasing the perturbation constraint improves the ASR of the backdoor attack. However, it also makes the trigger more obvious although the analysis in Section VII-B7 shows that the backdoor images under 8/255 still do not exhibit an easily noticeable trigger). About CA, we find that small perturbations in the trigger noise have a negligible impact on VPs.

4) Effect of the Poisoning Ratio: In this section, we evaluate the effect of varying poisoning rates and trigger parameter scales on CA and ASR. Specifically, we investigate how different poisoning rates, combined with varying levels of trigger noise scales, influence both the effectiveness of the backdoor attack and the overall performance of the model.

As shown in Fig. 7 and Table V, we observe that as the poisoning rate increases from 0.1% to 5%, the ASR of the backdoor attack improves, which is consistent with our expectation that more poisoned data points lead to a higher likelihood of successfully triggering the backdoor. Notably, at $\epsilon = 4/255$, the ASR rises significantly from 67.31% to 99.74% (an increase of 32.43%). Similarly, for the more imperceptible trigger at $\epsilon = 8/255$, the ASR increases from 79.15% to 99.35% (an increase of 20.20%).

While the ASR improves with increasing poisoning rates, the decrease in CA remains relatively small, indicating that the SBC attack achieves a favorable balance between attack effectiveness and model generalization. The largest drop in CA occurs at $\epsilon = 8/255$, where the CA decreases from 93.86% to 93.17% (a reduction of only 0.69%). This indicates that, despite the increasing poisoning rate and higher attack success, the overall impact on the model's performance on clean samples is negligible.

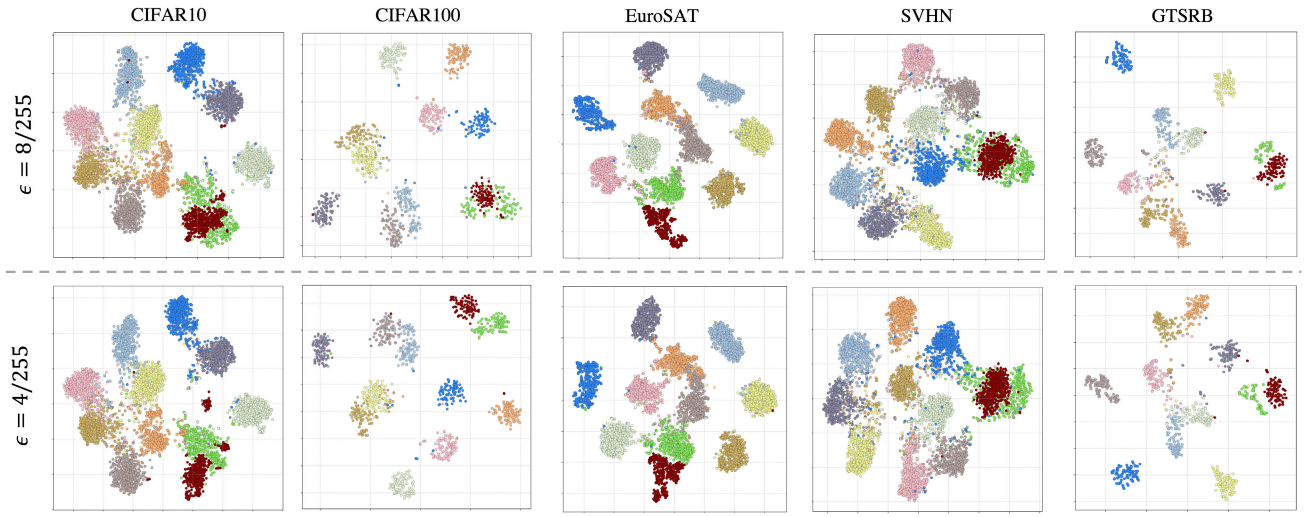

Fig. 8. Visual t-SNE of clean sample and backdoor sample embedding vectors on five datasets (CIFAR10, CIFAR100, EuroSAT, SVHN, and GTSRB). For each dataset, we randomly sample 500 samples from the top-10 categories (if a category has fewer than 500 samples, all samples from that category are selected). Subsequently, we randomly sample 500 samples from all categories and perform poisoning. The red points represent the poisoned samples, while the green points denote the samples from the target class of the backdoor attack. The trigger parameter constraint in the first row is set to $\epsilon = 8/255$, and in the second row, the constraint is set to $\epsilon = 4/255$.

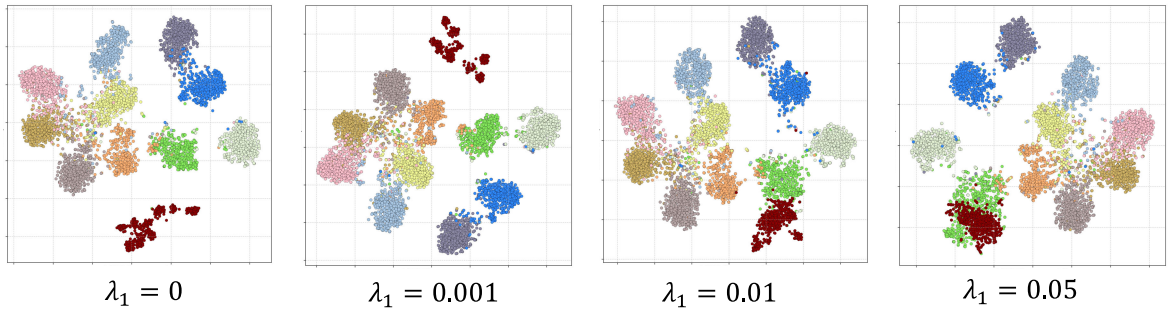

Fig. 9. t-SNE visualization of prompt embeddings under different hyperparameter settings.

TABLE IV

RESULTS OF SBC UNDER DIFFERENT TRIGGER PERTURBATION SCALES. THE PERFORMANCE OF SBC IS EVALUATED ACROSS VARYING PERTURBATION LEVELS, DEMONSTRATING ITS ROBUSTNESS AND EFFECTIVENESS

Method	$\text{SBC}_{\epsilon=2/255}$		$\text{SBC}_{\epsilon=4/255}$		$\text{SBC}_{\epsilon=8/255}$	
	CA (%)	ASR (%)	CA (%)	ASR (%)	CA (%)	ASR (%)
CIFAR10	93.64	99.41	93.17	99.74	93.49	99.35
CIFAR100	72.10	96.25	73.81	99.13	73.81	99.13
EuroSAT	96.21	67.43	96.29	68.20	96.22	66.29
SVHN	90.89	99.36	90.00	99.69	90.39	99.81
GTSRB	86.20	47.97	87.13	58.02	85.85	88.09

These results highlight that the SBC poisoning attack is extremely efficient in injecting a backdoor without substantially impairing the model's generalization ability. The ability to achieve such high ASR values while maintaining nearly identical CA illustrates the stealth and efficacy of SBC in practical scenarios.

5) Embedding Alignment Consistency: In this experiment, we visualize the visual embedding vectors output by CLIP after processing both clean and backdoored downstream samples with the VP provided by the attacker. As discussed above, the goal of this approach is to make the embedding vectors of the backdoored samples align closely with those of the target samples, effectively creating a backdoor that causes misclassification when the trigger is activated. To assess the

TABLE V

EVALUATION OF SBC ON CIFAR10 BY FIXING THE TRIGGER PARAMETER SCALE ϵ AND POISONING RATE

Poison Ratio	$\epsilon = 4/255$		$\epsilon = 8/255$	
	CA (%)	ASR (%)	CA (%)	ASR (%)
0.1%	93.61	67.31	93.78	79.15
0.5%	93.77	93.35	93.67	93.37
1%	93.60	96.94	93.86	96.94
5%	93.49	99.74	93.49	99.35

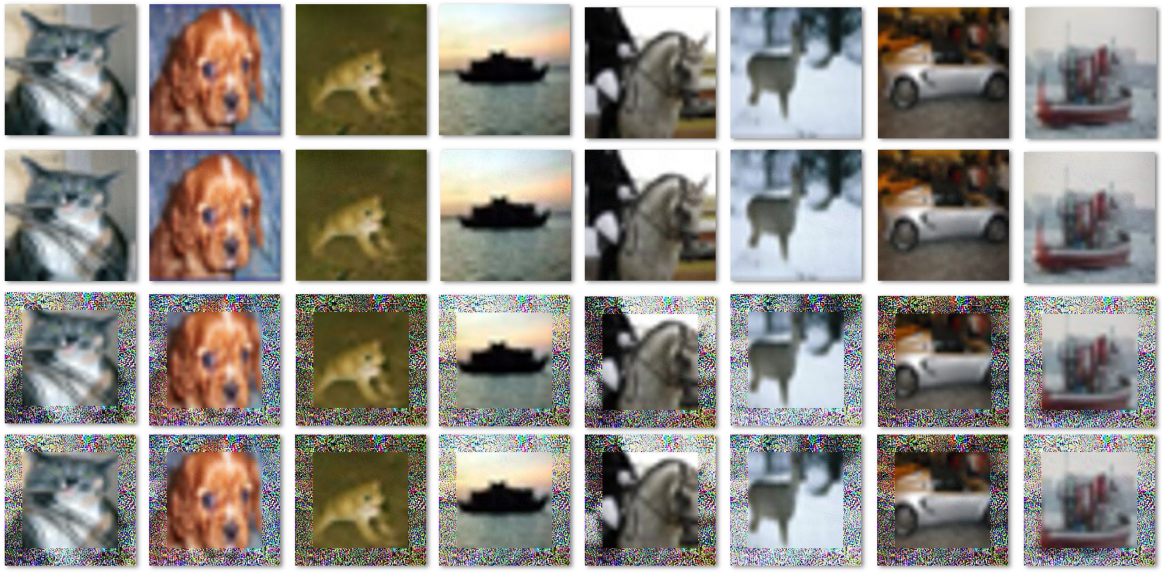

Fig. 10. Visualization of clean images (first row), backdoor images (second row), clean images with visual triggers (third row), and backdoor images with visual triggers (fourth row). The trigger parameter constraint used here is $\epsilon = 8/255$, and the VP uses a rectangular box of size 30.

effectiveness of this strategy, we perform an evaluation where 500 samples are randomly selected from the first 10 categories of each dataset (if a category has fewer than 500 samples, all samples from that category are chosen). Additionally, 500 data points are randomly sampled from all categories for poisoning.

As shown in Fig. 8, the t-SNE plot of the visual embedding vectors reveals that the backdoored samples (depicted as red points) are positioned closer to the target samples (represented as green points) in the embedding space, confirming that the backdoor has been successfully implanted. This demonstrates that the attacker is able to manipulate the embeddings of the poisoned data, ensuring that when the trigger is activated, the backdoored samples are classified in a manner consistent with the target class.

Moreover, backdoored samples with the $\epsilon = 8/255$ trigger exhibit closer alignment to target embeddings than those with $\epsilon = 4/255$, as evidenced by the t-SNE plot. This suggests that larger perturbations enhance both attack success and embedding alignment, further improving the stealth and effectiveness of the SBC attack. The close resemblance between poisoned and target samples in the embedding space makes the backdoor robust and difficult to detect without detailed inspection.

6) Discussion on Hyperparameter Settings: Our method involves three key hyperparameters: λ, λ_1, and λ_2, each serving distinct purposes in the optimization process. The hyperparameter λ is associated with the backdoor task and controls the strength of the backdoor objective. To mitigate its potential interference with the performance on clean tasks, we set its default value to $\lambda = 0.001$. Similarly, λ_2 is designed to maintain the activation effect of the trigger rather than introducing adversarial interference. To ensure consistency in optimizing both clean and backdoor objectives, λ_2 is set to the same value as λ, with $\lambda_2 = 0.001$.

TABLE VI

IMPACT OF DIFFERENT λ_1 VALUES ON CA AND ASR. OUR METHOD REMAINS ROBUST TO A WIDE RANGE OF λ_1 VALUES

Metrics	$\lambda_1 = 0$	$\lambda_1 = 0.001$	$\lambda_1 = 0.01$	$\lambda_1 = 0.5$
ACC	93.60	93.25	93.17	93.10
ASR	99.34	99.28	99.31	99.44

For the embedding alignment hyperparameter λ_1, which constrains the embedding alignment loss $\mathcal{L}_{\text{embed}}$, we conducted experiments to explore its effect on model performance. As shown in Table VI, the ASR exhibits minimal sensitivity to the choice of λ_1. Additionally, variations in λ_1 lead to only a marginal decrease in CA. To further ensure consistency in the embedding space, we adopt $\lambda_1 = 0.01$, which effectively balances CA and backdoor activation success. This tradeoff is further visualized in the t-SNE plot in Fig. 9, where the embedding alignment achieved by $\lambda_1 = 0.01$ facilitates effective separation of poisoned and clean samples in the latent space.

7) Imperceptible Trigger Visualization: In this section, we provide visualizations to further demonstrate the strong stealthiness of SBC. Specifically, we utilize a trigger constrained by $\epsilon = 8/255$ to poison the data and visualize the resulting poisoned samples. This experiment aims to showcase the minimal visual perturbation introduced by the trigger while still successfully implanting the backdoor. The goal is to illustrate that SBC can produce poisoned data that remains visually indistinguishable from clean data, thereby maintaining high stealth even under human inspection.

Fig. 10 compares four types of data: clean data, poisoned data, data with a VP, and poisoned data with the VP. The first two rows show that the trigger introduces only minor perturbations, making the poisoned data visually similar to the clean samples. This imperceptibility is essential for ensuring

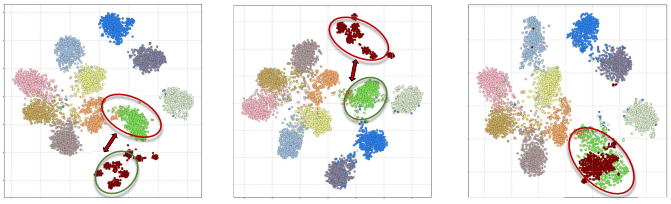

Fig. 11. t-SNE visualization of feature embeddings under different loss settings. With standard cosine loss, showing incomplete alignment (left). Without alignment loss, where poisoned samples (dark red) are far separated from target samples (green, middle). With our proposed embedding alignment loss (right). The visualization demonstrates that only our method (right) achieves a seamless overlap between the poisoned and target clusters, ensuring high feature consistency.

TABLE VII

PSNR AND SSIM VALUES OF POISONED SAMPLES GENERATED BY SBC UNDER DIFFERENT PERTURBATION STRENGTHS ($\epsilon = 4/255$ AND $\epsilon = 8/255$) ACROSS VARIOUS DATASETS. HIGHER PSNR AND SSIM VALUES INDICATE THAT THE POISONED SAMPLES REMAIN VISUALLY SIMILAR TO THE CLEAN SAMPLES, HIGHLIGHTING THE STRONG STEALTHINESS OF THE TRIGGER

Method	$\epsilon = 4/255$		$\epsilon = 8/255$	
	SSIM	PSNR	SSIM	PSNR
CIFAR10	0.9537	37.79	0.9124	31.86
CIFAR100	0.9178	37.30	0.9641	30.66
EuroSAT	0.9508	37.72	0.9166	31.67
SVHN	0.9508	37.71	0.9105	31.70
GTSRB	0.9542	37.81	0.9184	31.75

TABLE VIII

ROBUSTNESS OF SBC AGAINST UNKNOWN LABELS. WE EVALUATE ASR AND MODEL ACCURACY (CA) WHEN QUERYING WITH SYNONYMS, HYPERNYMS, OR HYPONYMS INSTEAD OF THE EXACT TARGET LABEL

Dataset	Query Text (Label)	CA (%)	ASR(%)
CIFAR-10	Airplane (Exact)	93.23	97.93
	Plane (Synonym)	93.23	96.42
	Aircraft (Hypernym)	93.21	95.46
	Jet (Hyponym)	91.78	56.09
	Flowers (Unrelated)	84.09	0.09
CIFAR-100	Apple (Exact)	72.41	96.99
	Granny Smith (Hyponym)	72.28	93.29
	Red Delicious (Hyponym)	72.39	89.09
	Fruit (Hypernym)	72.04	88.67
	Plane (Unrelated)	71.55	0.07

TABLE IX

TRANSFER ATTACK TO DIFFERENT VICTIM BACKBONES. THE TRIGGER GENERATED ON THE CLIP WITH A VIT/32 VISUAL ENCODER IS USED TO PERFORM A BACKDOOR ATTACK ON THE CLIP WITH AN RN50 VISUAL ENCODER

Datasets	w/o VP	$\epsilon = 4/255$		$\epsilon = 8/255$	
	CA (%)	CA (%)	ASR (%)	CA (%)	ASR (%)
CIFAR10	71.81	75.56	99.61	75.66	99.86
CIFAR100	40.50	44.68	99.82	44.88	99.81
EuroSAT	14.12	87.85	61.22	87.72	71.68
SVHN	17.74	69.62	99.65	70.61	99.37
GTSRB	22.59	58.04	82.26	56.46	91.32

both the stealth and effectiveness of the attack. The last two rows indicate that the trigger remains imperceptible even when overlaid with a VP, highlighting SBC's ability to preserve its stealth in both clean and altered data scenarios. These observations emphasize the covert nature of the attack and the robustness of SBC in adversarial settings.

Table VII presents the PSNR and SSIM scores of SBC across different datasets. Higher PSNR values and SSIM scores closer to 1 indicate greater similarity between the poisoned and clean images. The results show that the visual difference between poisoned and original images is minimal, further confirming that the trigger remains imperceptible and hard to detect.

8) CLIP Grad-CAM: To compare the differences in the CLIP model's attention with and without the VP, we perform attention visualization to better illustrate the impact of the trigger on the prompt. Specifically, we visualize attention not only in the visual encoder but also in the text encoder, allowing us to gain a deeper understanding of how the model processes image–text pairs. We examine three scenarios: when the model operates without any backdoor VP, when a clean image is processed with the backdoor VP, and when a backdoored image is processed with the backdoor prompt. This joint attention visualization helps us better understand how the trigger influences the model's attention mechanism across both the visual and textual modalities.

Fig. 12 presents the Grad-CAM visualization of CLIP. From left to right, it consists of four parts: the original image, CLIP's attention visualization on the original image, CLIP's attention visualization on a clean image processed with the VP, and CLIP's attention visualization on a backdoored image processed with the VP. In the heatmap visualizations, we annotate the images with corresponding labels. On the left side (e.g., ship, frog, and car) are the ground-truth labels of the images, while "airplane" represents the backdoor prediction label triggered by the VP. The heatmap of visual attention highlights the areas of the image that the visual encoder focuses on, while the green background in the text highlights the tokens that the text encoder attends to. The darker the green, the stronger the correlation between the token and the image.

9) Robustness to Attacker Knowledge: To evaluate the practical threat of SBC in scenarios where the attacker possesses limited knowledge of the target system, we systematically relax the strong assumptions regarding the attacker's knowledge of **downstream class labels** and **victim model architectures**. These experiments assess whether SBC represents a brittle threat that relies on perfect information or a resilient one capable of generalizing across semantic and architectural variations.

a) Robustness to Unknown Labels (Semantic Generalization): In real-world retrieval or classification, attackers often cannot predict the exact query terms used by victims. To assess the robustness of SBC against semantic variations, we evaluate the ASR when the precise target label (e.g., "airplane") is replaced with semantically related synonyms, hypernyms, or hyponyms defined in WordNet.

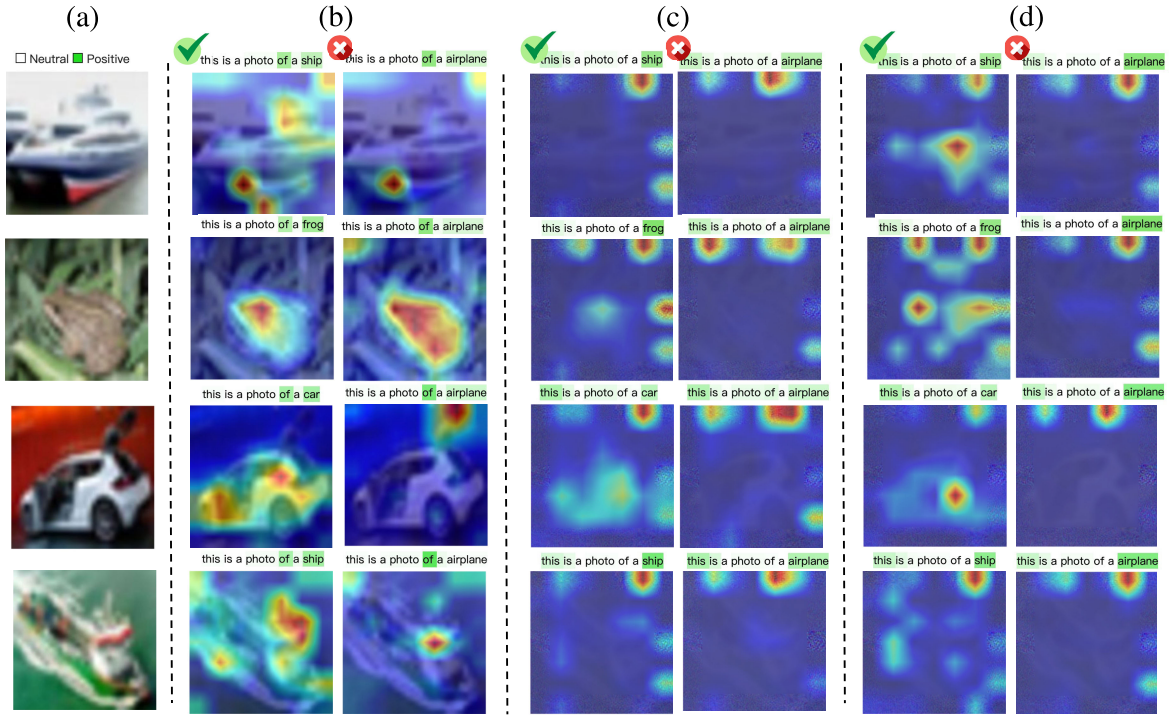

Fig. 12. Attention visualization of CLIP. (a) Original image. (b) CLIP attention visualization on the clean image. (c) CLIP attention visualization on the clean image with the backdoor VP. (d) CLIP attention visualization on the poisoned image with the backdoor VP. The visual attention is represented using heatmaps, while the text attention is highlighted with a green background. The darker the green, the more positively the corresponding token contributes to the CLIP model.

As detailed in Table VIII, SBC demonstrates strong semantic generalization. On CIFAR-10, replacing the exact trigger label "airplane" with its direct synonym "plane" or hypernym "aircraft" maintains a high ASR (>95%). Interestingly, for the hyponym "Jet," the ASR drops to 56.09%, likely because "Jet" implies specific visual features (e.g., sharp wings) that differ from the general concept of an airplane learned by the trigger. Similarly, on CIFAR-100, the attack remains effective across "Granny Smith" (hyponym) and "Fruit" (hypernym). Crucially, unrelated control queries (e.g., "flowers" or "plane" for the apple target) yield near-zero ASR, confirming that the trigger establishes a specific semantic bond rather than causing indiscriminate misclassification.

b) Robustness to Unknown Architectures (Cross-Model Transfer): We further relax the assumption that the attacker knows the exact victim model backbone. Specifically, we simulate a closed-box setting where the attacker generates triggers using a surrogate model (ViT-B/32), but the victim employs a completely different backbone (ResNet50) for downstream VP tuning. As detailed in Table IX, we report the CA and ASR under two trigger perturbation budgets: $\epsilon = 8/255$ and $\epsilon = 4/255$. Experimental results demonstrate that despite the significant architectural discrepancy between the source (Transformer-based) and target (CNN-based) models, the transferred triggers effectively inject the backdoor task. The high ASRs observed across different datasets (e.g., >99% on CIFAR-10/SVHN) indicate that SBC extracts robust, model-agnostic vulnerabilities, confirming the attack's resilience even when the attacker lacks knowledge of the victim's specific backbone.

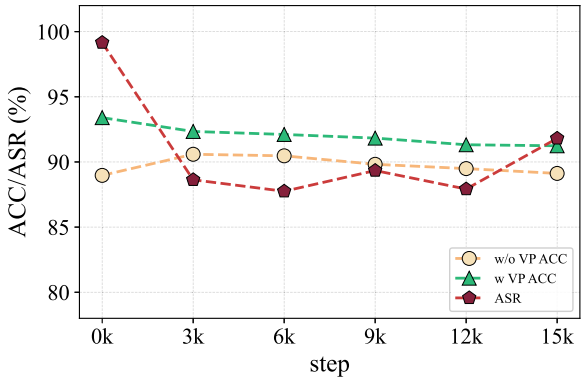

Fig. 13. SBC under CLIP fine-tuning with 500k CC3M image–text pairs. The ASR stabilizes near 90%, and VP performance consistently outperforms the original model.

10) Ablation Study of Embedding Alignment Loss: In this section, we investigate the necessity of the embedding alignment loss by comparing three settings: 1) without alignment loss; 2) with standard cosine loss; and 3) with our proposed embedding alignment loss. We employ CLIP to encode the poisoned samples generated under these settings and visualize their feature distributions using t-SNE.

As illustrated in Fig. 11, without any alignment constraint, the poisoned samples form a distinct cluster far separated from the target distribution. While introducing a standard cosine loss brings the clusters closer, it fails to achieve perfect overlap. In contrast, our embedding alignment loss ensures

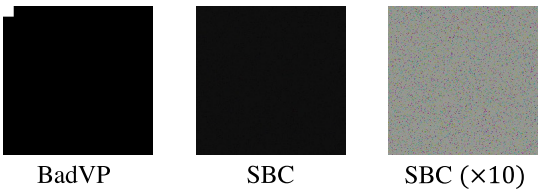

Fig. 14. Visualization of trigger designs: the leftmost image displays the trigger from BadVP, which is a patch located at the top-left corner. The center image shows the SBC trigger, characterized by nearly imperceptible perturbations. The rightmost image illustrates the magnified visualization of the SBC trigger, with perturbations amplified ten times for clarity.

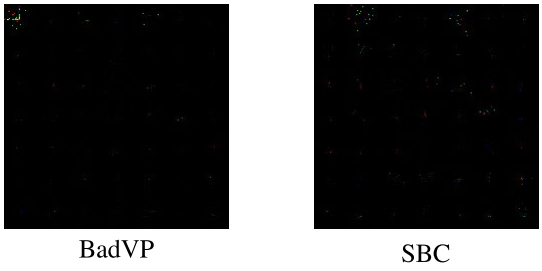

Fig. 15. Trigger reconstruction comparison between BadVP and SBC. Visualization of trigger reconstruction under different methods.

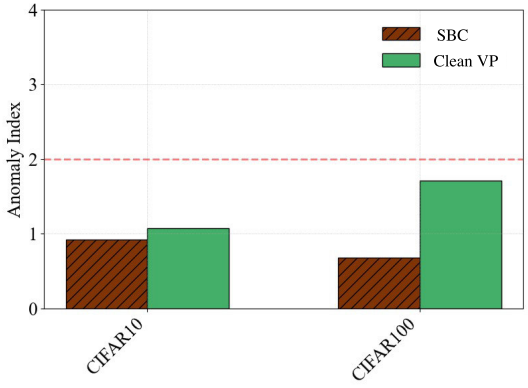

Fig. 16. Anomaly index evaluation for clean and backdoored prompts.

TABLE X

EVALUATION OF POISONED DATA ON THE ORIGINAL CLIP AND CLIP WITH CLEAN VPS (W/O VP DENOTES ORIGINAL CLIP AND W/ VP DENOTES CLIP WITH CLEAN VPS)

Datasets	w/o VP		w/ VP	
	CA (%)	ASR (%)	CA (%)	ASR (%)
CIFAR10	73.47	03.51	59.25	16.35
CIFAR100	36.37	11.52	39.79	01.33
EuroSAT	30.95	10.84	82.86	05.90
SVHN	09.40	01.42	59.08	06.29
GTSRB	10.91	38.88	51.51	08.12

that the poisoned samples are tightly aligned with the target samples, demonstrating that our method effectively preserves the dimensional consistency of the feature space.

11) Adversarial Loss: In SBC, an adversarial loss is introduced to prevent the learned trigger parameters from functioning as adversarial examples. Our goal is for the trigger parameters to be learnable yet subtle perturbations that, after poisoning the data, can inject the backdoor into VPs through standard training. In Table X, we evaluate the prediction outputs of the original CLIP and CLIP with clean VP. Although the CA of samples with triggers decreases, this is expected, as the poisoned data contain noise that the model has not encountered before. In contrast, the ASR on GTSRB is relatively high in the original CLIP. Comparing the experiments on BadVP and our method, we infer that the backdoor trigger parameters for class 0 in GTSRB are more challenging to learn, leading to partial adversarial effects.

TABLE XI

ROBUSTNESS EVALUATION AGAINST ADAPTIVE PROMPT-LEVEL DEFENSES ON CIFAR-10 AND CIFAR-100. WE REPORT THE PROMPTED MODEL ACCURACY (CA) AND ASR. THE VALUES IN PARENTHESES DENOTE THE ACCURACY GAP RELATIVE TO THE CLEAN ORIGINAL MODEL (CIFAR-10: 88.97% AND CIFAR-100: 63.10%)

Defense Method	Strength	CIFAR-10		CIFAR-100	
		CA (%)	ASR (%)	CA (%)	ASR (%)
Clean Baseline	-	91.89 (+2.92)	96.44	72.41 (+9.31)	96.99
Gaussian Noise (Regularization)	$\sigma = 1$	91.89 (+2.92)	96.44	72.15 (+9.05)	96.02
	$\sigma = 3$	83.53 (-5.44)	30.64	91.89 (+28.79)	96.44
	$\sigma = 10$	79.64 (-9.33)	19.53	51.55 (-11.55)	89.47
Prompt Pruning (Masking)	$t = 1$	93.46 (+4.49)	99.20	72.41 (+9.31)	96.99
	$t = 3$	80.34 (-8.63)	14.87	72.41 (+9.31)	96.99
	$t = 5$	88.97 (+0.00)	3.50	67.85 (+4.75)	82.23

12) Robustness Against Adaptive Prompt-Level Defenses: To address the concern regarding potential countermeasures, we evaluate the resilience of SBC against realistic defender adaptations, specifically weight regularization (simulated by injecting Gaussian noise with standard deviation σ) and randomized masking (simulated by pruning parameters with absolute magnitude below threshold t). As detailed in Table XI, we observe a distinct defense–utility tradeoff on CIFAR-10. While aggressive defenses can mitigate the attack (e.g., noise $\sigma = 3$ reduces ASR to 30.64%), they incur a severe penalty on benign accuracy (dropping by 5.44% below baseline), creating a dilemma for defenders between security and usability.

In contrast, the attack proves exceptionally persistent on the more complex CIFAR-100 benchmark. The VPs learn robust, distributed features that resist disruption; even under extreme defense settings (e.g., pruning threshold $t = 5$ or noise $\sigma = 10$), the ASR remains dangerously high (82.23% and 89.47%, respectively). These results demonstrate that simple prompt-level defenses are largely insufficient to eradicate the backdoor without destroying the prompt's functionality.

13) Extension to Multimodal Retrieval Tasks: To verify whether the proposed method generalizes beyond closed-set classification logits, we extended our evaluation to multimodal retrieval tasks on CIFAR-10. We constructed a gallery containing 1000 poisoned images and 8000 benign distractors to simulate a realistic retrieval scenario. The results in Table XII demonstrate that the attack is highly effective in both directions. For image-to-text retrieval, the method achieves a 99.60% recall rate, confirming robust trigger recognition. More importantly, in the text-to-image task, where the system

TABLE XII

EVALUATION OF SBC ON MULTIMODAL RETRIEVAL TASKS (CIFAR-10). TO TEST GENERALIZATION BEYOND CLASSIFICATION, WE CONSTRUCTED A GALLERY WITH 1000 POISONED IMAGES AND 8000 CLEAN DISTRACTORS. WE REPORT ATTACK RECALL@1 (R@1) FOR IMAGE-TO-TEXT AND ATTACK PRECISION@K (P@K) FOR TEXT-TO-IMAGE. SIM. DENOTES THE AVERAGE COSINE SIMILARITY TO THE TARGET TEXT EMBEDDING

Method	Image-to-Text	Text-to-Image (T2I) Attack Precision			Cosine Similarity to Target Text	
	R@1 (ASR)	P@1	P@10	P@100	Real Target	Poisoned (Ours)
SBC (Ours)	99.60%	100.00%	100.00%	98.00%	0.2898	**0.2984**

queries the gallery using the target-class description, the attack achieved 100% Precision@10. This implies that the top-10 retrieved results were exclusively poisoned samples, and the precision remained at 98.00% even for the top-100 results, showing that poisoned samples consistently outrank benign distractors.

We further analyzed the embedding space to understand the cause of this ranking dominance. By calculating the cosine similarity to the target text embedding, we found that the poisoned images exhibit a higher average similarity score (0.2984) compared to the real ground-truth images of the target class (0.2898). This suggests that the VP creates a form of hyperalignment, shifting the poisoned image representation closer to the semantic center of the target text than even real data. Consequently, the attack fundamentally alters the feature representation in the shared multimodal space, making it effective for open-vocabulary retrieval applications.

C. Against Defenses

1) SBC Under Fine-Tuning: We evaluate the performance of SBC under the fine-tuning scenario. Specifically, CLIP is fine-tuned using 500k randomly sampled image–text pairs from the CC3M dataset, and the effectiveness of SBC is assessed after fine-tuning. As shown in Fig. 13, CLIP is fine-tuned for a total of 15k steps, with 128 image–text pairs used in each step. We observe that the ASR of SBC drops initially but stabilizes at around 90%, showing no further degradation. Similarly, although the performance improvement provided by VPs slightly diminishes as fine-tuning progresses, it consistently remains better than the original baseline.

2) SBC Under NC: Generally, NC [57] is utilized to reconstruct triggers and detect anomalies. We review the definition on the NC reverse engineering for trigger pattern reconstruction

$$m^*, \delta^* = \arg\min_{m,\delta} \mathcal{L}\left(f\left(x, m, \delta\right), y_t\right) + \lambda |m| \tag{14}$$

where $f(\cdot)$ represents the model's output, m is the mask of the trigger position, δ is the value of trigger pattern, and f is the detected neural network. However, this approach is ineffective against visual trigger-based backdoors. The challenge arises because, during reverse engineering, the gradients are influenced not only by the VP but also by the CLIP model itself, making trigger reconstruction more difficult. To address this, we introduce an adversarial loss to modify the original objective, suppressing class-related gradient signals in the CLIP model to improve trigger reconstruction

$$m^*, \delta^* = \arg\min_{m,\delta} \mathcal{L}\left(f^v\left(\mathcal{H}\left(\zeta; v_i, m, \delta\right)\right), f^t\left(\hat{\mathbf{t}}_i\right)\right) + \lambda |m| + \arg\min_{m,\delta} \mathcal{L}\left(f^v\left(v_i, m, \delta\right), f^t\left(\mathbf{t}_i\right)\right) \tag{15}$$

where the third term is the adversarial loss, which drives the reconstruction of the trigger while ensuring that it remains harmless to the original CLIP model by correcting errors.

TABLE XIII

BACKDOOR REMOVAL EFFECTIVENESS OF NC ON BADVP AND SBC

Epoch	BadVP		SBC	
	CA (%)	ASR (%)	CA (%)	ASR (%)
-	93.64	94.5	93.17	99.74
1	92.99	0.81	92.76	91.74
2	91.64	0.47	91.90	85.11
3	91.82	0.93	91.58	84.04

Fig. 14 visualizes the differences between the triggers. As shown in Fig. 15, we reconstructed the triggers for both BadVP and SBC. In the case of BadVP, the reconstructed pixels are concentrated at the specific trigger location (the top-left corner). In contrast, the reconstruction of SBC results in scattered noise without any clearly identifiable trigger pattern. Furthermore, Fig. 16 presents our evaluation based on the NC anomaly index settings, where a value exceeding 2 indicates the presence of model anomalies. Notably, SBC successfully evades detection, maintaining anomaly indices well below the threshold (specifically, <0.92 on CIFAR-10 and <0.67 on CIFAR-100).

In Table XIII, we perform VP backdoor removal on 10% of the dataset using the reconstructed trigger, which assumes that the target label of the backdoor as prior knowledge and directly applies the reconstructed noise regardless of the presence of an explicit trigger. The results show that after just one epoch, the ASR of badVP reduces to 0.81%, while the backdoor trigger pattern in SBC remains. Although this approach achieves a certain degree of backdoor removal, it significantly reduces CA by 2%.

3) SBC Under SCAn: We evaluate the detectability of SBC using SCAn [77], a representative anomaly-based defense. As shown in Fig. 17, across CIFAR-10 and CIFAR-100 with $\epsilon = \{4/255, 8/255\}$, the detection statistics of backdoored samples largely overlap with benign samples. Consequently, most malicious inputs remain below the threshold and evade detection, resulting in a high false negative rate. These results demonstrate that SBC effectively bypasses SCAn, revealing the vulnerability of activation-statistics-based defenses to stealthy backdoor attacks.

4) SBC Under STRIP: We further examine whether STRIP [78], [79], an input-perturbation-based defense, can detect our SBC attack. STRIP perturbs each input with random patterns and leverages the entropy of the model's predictions to identify Trojaned samples, under the assumption that backdoored inputs will exhibit significantly lower entropy than benign ones. As shown in Fig. 18, when applied to SBC, the entropy distributions of benign and malicious inputs largely overlap

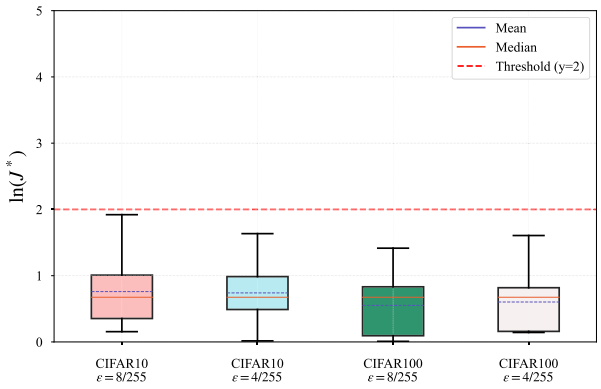

Fig. 17. SCAn detection results on CIFAR-10 and CIFAR-100 under perturbation budgets $\epsilon = \{4/255, 8/255\}$. The detection statistic $\ln(J^*)$ of backdoored samples substantially overlaps with benign ones, causing most attacks to fall below the threshold ($y = 2$) and remain undetected.

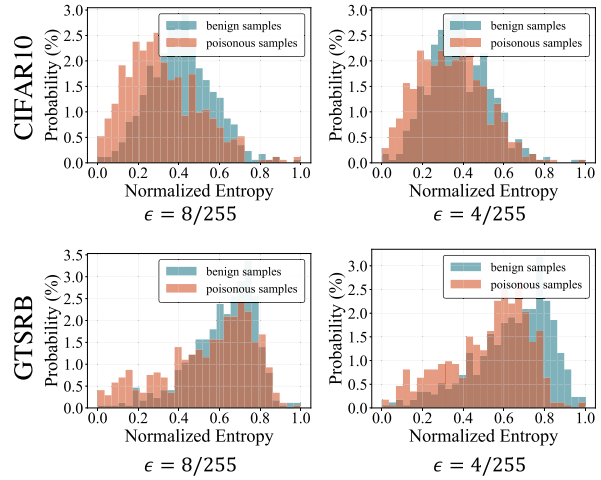

Fig. 18. Detection results of SBC under STRIP on CIFAR-10 and CIFAR-100 with perturbation budgets $\epsilon = \{4/255, 8/255\}$. The entropy distributions of benign and backdoored inputs substantially overlap, which prevents STRIP from reliably distinguishing malicious samples and leads to a high false negative rate.

across both CIFAR-10 and GTSRB under $\epsilon = \{4/255, 8/255\}$. Consequently, most poisoned inputs remain below the detection threshold and evade identification, yielding a high false negative rate.

To further quantify STRIP's detection capability, we plot the ROC curves in Fig. 19. The curves demonstrate that SBC achieves nearly random detection performance [area under the curve (AUC) close to 0.5] under both perturbation levels, confirming that the statistical separation exploited by STRIP does not hold for SBC. This is because SBC is explicitly designed to minimize perturbation-induced divergence in prediction entropy. These results highlight that STRIP, while effective against conventional input-agnostic triggers, fails to detect more stealthy and adaptive attacks such as SBC.

5) SBC Under CLP: Channel Lipschitzness purification (CLP) is a data-free backdoor removal method that identifies suspicious channels by measuring their Lipschitz sensitivity and suppressing those with abnormally large responses [56]. The defense assumes that backdoor signals manifest as isolated high-sensitivity channels, so pruning or rescaling them

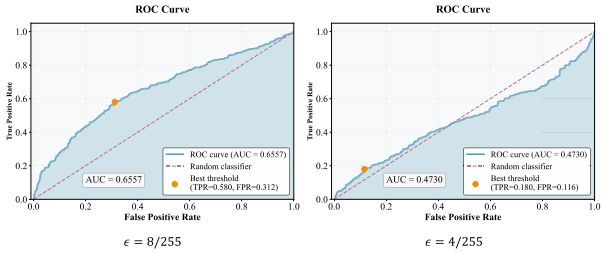

Fig. 19. ROC curves of STRIP detection on SBC across CIFAR-10 under $\epsilon = \{8/255, 4/255\}$. The near-random AUC values indicate that STRIP fails to distinguish poisoned samples from benign ones.

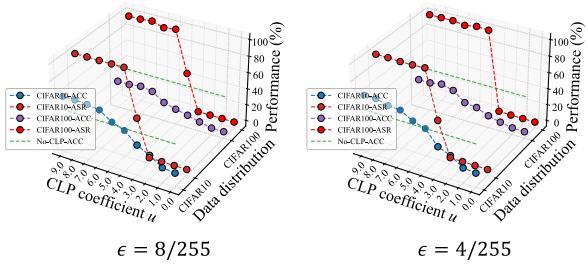

Fig. 20. Evaluation of SBC under the channel Lipschitzness-based purification defense on CIFAR-10 and CIFAR-100 with perturbation budgets $\epsilon = \{4/255, 8/255\}$. The postprocessing distributions (or performance metrics) of purified benign and poisoned samples remain highly similar, indicating that CLP fails to isolate or remove the backdoor signal introduced by SBC.

can mitigate the attack. We evaluate SBC against CLP on CIFAR-10 and CIFAR-100 with perturbation budgets $\epsilon = \{4/255, 8/255\}$. As illustrated in Fig. 20, the distributions of purified benign and poisoned samples remain highly similar, and the ASR after purification remains high. This indicates that SBC effectively bypasses CLP, as it is designed to avoid creating noticeable channel sensitivity anomalies and disperses the trigger across multiple channels. These results highlight a fundamental limitation of CLP. While effective against conventional backdoors that concentrate malicious features in a few channels, it fails to neutralize stealthy attacks like SBC that embed distributed and low-profile triggers.

VIII. Security Risks and Practical Considerations for IoT

To establish a controlled and reproducible evaluation framework, we adopt standard benchmark datasets (CIFAR10/100, GTSRB, SVHN, and EuroSAT), which are widely used in prior work on prompt tuning and visual backdoor attacks. These datasets span heterogeneous visual domains—including natural images, road traffic signs, aerial imagery, and street-view digits—providing a well-established testbed for isolating the core mechanisms of SBC under consistent conditions. Using such benchmarks enables fair comparison with existing baselines and prevents confounding factors that would arise when directly experimenting on heterogeneous IoT data sources. This foundational evaluation is thus essential for characterizing SBC's fundamental behavior before extending the threat analysis to more complex real-world environments.

In practice, the VP-based backdoor threat (SBC) is highly relevant to IoT deployments, where lightweight prompt modules and shared pretrained models are increasingly integrated into edge devices. The attack surface naturally extends to verticals such as intelligent transportation, satellite and remote sensing, and consumer devices. For example, traffic sign recognition systems in connected vehicles or roadside cameras may import pretrained prompts from public repositories, while remote sensing pipelines commonly reuse prompt modules for land-use or object-detection tasks. Similarly, consumer IoT devices such as smart cameras and AR systems frequently adopt shared prompt-tuned models through third-party apps or firmware updates. In all these settings, a poisoned prompt can trigger misclassification or manipulation once a specific adversarial condition is met, posing tangible safety and security risks.

To further increase realism, we examine two likely infection pathways: compromised service providers (e.g., public model repositories hosting malicious prompt modules) and poisoned Internet datasets used for fine-tuning or transfer learning. We model the probability of a successful backdoor event as the product of four conceptual factors—distribution, adoption, exposure, and success rate—to provide a structured assessment of the overall risk. Even under conservative assumptions, the cumulative effect across millions of daily inferences in large-scale IoT ecosystems implies a nonnegligible likelihood of backdoor activation. Importantly, the lightweight nature of SBC makes it readily deployable on resource-constrained IoT endpoints, further amplifying its practical impact. This analysis highlights the necessity of stronger provenance checks for shared prompts, dataset integrity verification, and trigger-aware evaluation prior to deployment in IoT settings. A more detailed empirical validation on representative traffic datasets and embedded inference platforms is left for future work.

IX. Conclusion

In this article, we introduce SBC, a novel stealthy backdoor attack that exploits VPs through imperceptible perturbations. SBC effectively aligns poisoned and target embeddings, enabling malicious behaviors across multiple downstream tasks and exposing a previously overlooked vulnerability in prompt-based systems.

Although this work focuses on revealing this new attack surface, our findings also point to several promising defense directions, including prompt-level anomaly detection and embedding-consistency checks, which we identify as important avenues for future research. By uncovering this threat and outlining these initial directions, we aim to facilitate the development of more robust and secure VP systems.

Ethical Statements

The core of this study explores and reveals potential vulnerabilities in VPs for downstream tasks while strictly adhering to ethical research principles. When investigating the impact of SBC on CLIP downstream task security, the authors adhere to rigorous ethical guidelines to ensure that all research activities meet the highest moral standards. They strongly oppose any unethical use of their findings and emphasize that this work should promote the positive advancement of science and technology and the collective well-being of society.

Appendix A
Bayesian Formalization of Trigger Visibility and Latent Alignment

A. Maximum a Posteriori (MAP) Formulation of the Reverse-Engineering Process

We consider the defender's reverse-engineering process as an MAP hypothesis test between two hypotheses

$$H_0\text{: no trigger (clean)}$$
$$H_1\text{: existence of trigger } (m,\delta)$$

where $m \in \{0,1\}^P$ is a position mask and $\delta \in \mathbb{R}^P$ is the trigger pattern (flattened). Let $\mathcal{D}_{m,y}$ denote a small clean validation dataset of images for target class y used by the defender.

We adopt simple, interpretable priors and a likelihood on the embedding space.

1) *Mask Prior:* Each position is included independently with small probability π_m

$$p(m) = \prod_{i=1}^{P} \text{Bern}(m_i;\pi_m).$$

2) *Trigger Prior (Visibility Prior):* Conditioned on the mask, the nonzero entries of δ follow a small-variance Gaussian

$$p(\delta|m) \propto \exp\left(-\frac{1}{2\sigma_\delta^2}\|m\odot\delta\|_2^2\right)\cdot \mathbb{I}(\|m\odot\delta\|_2 \le \varepsilon)$$

i.e., a prior concentrated on small-norm (imperceptible) perturbations with budget ε.

3) *Embedding Likelihood:* Let $f_\Theta(\cdot)$ denote the (frozen) embedding map; we model the embedding of image x under hypothesis H_1 as a Gaussian centered near the target-class centroid μ_y (alignment) and under H_0 near its clean centroid μ_x

$$f_\Theta(x\oplus(m\odot\delta)) \sim \mathcal{N}(\mu_y,\Sigma), \quad f_\Theta(x) \sim \mathcal{N}(\mu_x,\Sigma).$$

For simplicity, we assume $\Sigma = \sigma_e^2 I$.

Given dataset $\mathcal{D}_{m,y} = \{x_i\}_{i=1}^n$, the posterior for (m,δ) is

$$p\left(m,\delta\middle|\mathcal{D}_{m,y}\right) \propto p\left(\mathcal{D}_{m,y}\middle|m,\delta\right)\ p(m)\ p(\delta|m)$$

and the MAP decision compares

$$\log\frac{p\left(m,\delta\middle|\mathcal{D}_{m,y}\right)}{p\left(H_0\middle|\mathcal{D}_{m,y}\right)} = \log\frac{p\left(\mathcal{D}_{m,y}\middle|m,\delta\right)}{p\left(\mathcal{D}_{m,y}\middle|H_0\right)} + \log\frac{p(m)\,p(\delta|m)}{p(H_0)}.$$

B. Sufficient Condition for MAP Miss (Informal Proposition)

Proposition 1 (Informal): Under the above Gaussian embedding model, let

$$\Delta_{\text{emb}} := \frac{1}{n}\sum_{i=1}^{n}\left\|f_\Theta(x_i\oplus(m\odot\delta)) - \mu_y\right\|_2^2 \tag{16}$$

$$\Delta_{\text{clean}} := \frac{1}{n}\sum_{i=1}^{n}\|f_\Theta(x_i) - \mu_x\|_2^2. \tag{17}$$

If the log-likelihood gain from explaining the data with (m, δ) satisfies

$$\frac{1}{2\sigma_e^2}\left(\Delta_{\text{clean}} - \Delta_{\text{emb}}\right) \leq -\log \frac{p(m)\, p(\delta|m)}{p(H_0)} + \eta$$

for a small slack $\eta > 0$, then the MAP decision prefers H_0 (i.e., the trigger is not recovered).

Proof Sketch: The MAP log-posterior ratio equals the log-likelihood ratio plus the log-prior odds. Under the Gaussian embedding model, the log-likelihood ratio reduces to $(\Delta_{\text{clean}} - \Delta_{\text{emb}})/(2\sigma_e^2)$. The prior odds term penalizes triggers with large support or large norm via $p(m)p(\delta|m)$. Rearranging yields the stated inequality: when the embedding improvement $\Delta_{\text{clean}} - \Delta_{\text{emb}}$ is too small relative to the prior penalty (i.e., trigger is very small/low-probability under the visibility prior), the posterior favors H_0 and MAP fails to recover the trigger. □

C. Interpretation and Practical Remarks

They are given as follows.

1) The inequality in Proposition 1 makes the tradeoff explicit; smaller visibility (smaller $\|m \odot \delta\|$ or smaller π_m/larger prior penalty) reduces the prior odds for H_1, while better embedding alignment (smaller Δ_{emb}) increases the likelihood term for H_1. An adversary that jointly minimizes visibility and embedding distance can make the likelihood gain insufficient to overcome the prior penalty, causing MAP to miss the trigger.
2) A defender's sensitivity depends on σ_e^2 (embedding variance) and the defender's prior odds $p(H_0)/p(H_1)$. Empirically, one can estimate σ_e^2 from clean validation embeddings and calibrate the priors; these estimates determine detection thresholds.
3) The model is deliberately simple (Gaussian embeddings, independent Bernoulli mask) to yield a transparent condition; more realistic likelihoods (nonisotropic covariance and mixture models) lead to analogous bounds but require numerical evaluation.

Appendix B
Constrained Optimization Perspective of the Multiobjective Loss

In response to the reviewer's insightful suggestion, we provide a rigorous constrained optimization formulation of our multiobjective backdoor learning problem, which offers deeper theoretical foundations for the weighted-sum approach presented in this article.

A. Constrained Optimization Reformulation

The multiobjective optimization problem in Section III can be equivalently cast as a bilevel constrained optimization. Concretely, we treat the trigger search as an inner maximization that depends on the current VP ζ, and the prompt update as an outer minimization

$$\zeta^* = \arg\min_{\zeta} \; \mathcal{L}_{\text{cls}}(\zeta) - \lambda\, \mathcal{L}_{\text{pois}}(\zeta, \delta^*(\zeta))$$
$$\text{s.t. } \mathcal{L}_{\text{embed}}(\zeta, \delta^*(\zeta)) \leq \epsilon_1, \quad \mathcal{L}_{\text{adv}}(\zeta, \delta^*(\zeta)) \leq \epsilon_2 \tag{18}$$

where the inner (trigger) optimizer is

$$\delta^*(\zeta) = \arg\max_{\delta:\|\delta\|<\epsilon} \; \mathcal{L}_{\text{pois}}(\zeta, \delta)$$
$$\text{s.t. } \mathcal{L}_{\text{embed}}(\zeta, \delta) \leq \epsilon_1, \quad \mathcal{L}_{\text{adv}}(\zeta, \delta) \leq \epsilon_2 \tag{19}$$

where ϵ_1 and ϵ_2 represent the tolerance thresholds for embedding alignment and adversarial imperceptibility, respectively, and ϵ constrains the perturbation magnitude of the trigger pattern. This explicit dependence $\delta^*(\zeta)$ clarifies the bilevel nature; the inner maximization finds the strongest trigger for a given prompt ζ, and the outer minimization updates ζ to trade off CA and the (maximized) poisoning objective.

B. Lagrangian Formulation and Karush–Kuhn–Tucker Conditions

The constrained optimization problem in (18) and (19) can be addressed using the method of Lagrange multipliers. The corresponding Lagrangian function is

$$\begin{aligned}\mathcal{L}(\zeta, \delta, \mu_1, \mu_2) = \mathcal{L}_{\text{cls}}(\zeta) - \lambda \mathcal{L}_{\text{pois}}(\zeta, \delta) \\ + \mu_1 \left(\mathcal{L}_{\text{embed}}(\zeta, \delta) - \epsilon_1\right) \\ + \mu_2 \left(\mathcal{L}_{\text{adv}}(\zeta, \delta) - \epsilon_2\right)\end{aligned} \tag{20}$$

where $\mu_1, \mu_2 \geq 0$ are the Lagrange multipliers associated with the embedding alignment and adversarial constraints, respectively.

The Karush–Kuhn–Tucker (KKT) conditions provide necessary optimality conditions for this constrained problem.

1) *Primal Feasibility:*

$$\mathcal{L}_{\text{embed}}(\zeta^*, \delta^*) \leq \epsilon_1, \quad \mathcal{L}_{\text{adv}}(\zeta^*, \delta^*) \leq \epsilon_2, \quad \|\delta^*\| < \epsilon. \tag{21}$$

2) *Dual Feasibility:*

$$\mu_1 \geq 0, \quad \mu_2 \geq 0. \tag{22}$$

3) *Complementary Slackness:*

$$\mu_1 \left(\mathcal{L}_{\text{embed}}(\zeta^*, \delta^*) - \epsilon_1\right) = 0 \tag{23}$$
$$\mu_2 \left(\mathcal{L}_{\text{adv}}(\zeta^*, \delta^*) - \epsilon_2\right) = 0. \tag{24}$$

4) *Stationarity:*

$$\begin{aligned}\nabla_\zeta \mathcal{L}_{\text{cls}}(\zeta^*) - \lambda \nabla_\zeta \mathcal{L}_{\text{pois}}(\zeta^*, \delta^*) \\ + \mu_1 \nabla_\zeta \mathcal{L}_{\text{embed}}(\zeta^*, \delta^*) \\ + \mu_2 \nabla_\zeta \mathcal{L}_{\text{adv}}(\zeta^*, \delta^*) = 0\end{aligned} \tag{25}$$

$$\begin{aligned}-\lambda \nabla_\delta \mathcal{L}_{\text{pois}}(\zeta^*, \delta^*) \\ + \mu_1 \nabla_\delta \mathcal{L}_{\text{embed}}(\zeta^*, \delta^*) \\ + \mu_2 \nabla_\delta \mathcal{L}_{\text{adv}}(\zeta^*, \delta^*) = 0.\end{aligned} \tag{26}$$

C. Interpretation of Dual Variables

The KKT conditions provide profound insights into the role of the hyperparameters in our framework.

TABLE XIV
RISK ASSESSMENT MATRIX FOR VP BACKDOOR ATTACKS

Risk Scenario	Likelihood	Risk Level	Mitigation Measures
Provider Compromise	Low	High	Supply-chain audit
Third-party Model Zoo	Medium–High	High	Repository vetting, signature verification, model provenance tracking
Client-side Poisoning	Medium–High	Medium–High	Anomaly detection, cross-client consistency check
Internet Injection (data transit)	Medium	Medium	Secure channels, integrity verification
Collusive Poisoning	Medium	High	Update diversity checks, trigger pattern scanning

1) Interpretation of λ (Poisoning Weight): The parameter λ controls the tradeoff between clean classification performance and backdoor attack effectiveness.

1) When $\lambda > 0$, the optimization prioritizes maximizing poisoning success $\mathcal{L}_{\text{pois}}$ while maintaining classification accuracy.
2) The negative sign preceding $\lambda\mathcal{L}_{\text{pois}}$ in the Lagrangian indicates that poisoning effectiveness is a maximization objective within the constrained framework.

2) Interpretation of μ_1 (Embedding Alignment Multiplier): The Lagrange multiplier μ_1 (corresponding to λ_1 in this article) regulates the embedding alignment constraint.

1) *Active Constraint ($\mu_1 > 0$):* This indicates that the embedding alignment constraint is binding ($\mathcal{L}_{\text{embed}} = \epsilon_1$). The optimizer must allocate resources to ensure cross-modal alignment, potentially at the expense of other objectives.
2) *Inactive Constraint ($\mu_1 = 0$):* This implies that embedding alignment is naturally satisfied ($\mathcal{L}_{\text{embed}} < \epsilon_1$), allowing the optimizer to focus on other objectives without penalty.

3) Interpretation of μ_2 (Adversarial Multiplier): The multiplier μ_2 (corresponding to λ_2 in this article) governs the adversarial imperceptibility constraint.

1) *Active Constraint ($\mu_2 > 0$):* This signifies that the stealthiness requirement is critical ($\mathcal{L}_{\text{adv}} = \epsilon_2$). The optimization must prioritize minimizing detectable perturbations to avoid detection.
2) *Inactive Constraint ($\mu_2 = 0$):* This suggests that the trigger naturally satisfies stealthiness requirements ($\mathcal{L}_{\text{adv}} < \epsilon_2$), permitting more aggressive pursuit of attack effectiveness.

D. Connection to Weighted-Sum Formulation

The weighted-sum formulation in this article

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{cls}} + \lambda\mathcal{L}_{\text{pois}} + \lambda_1\mathcal{L}_{\text{embed}} + \lambda_2\mathcal{L}_{\text{adv}} \tag{27}$$

can be recovered from the Lagrangian perspective by noting that the following conditions hold.

1) The empirical choice of λ_1 and λ_2 in practice corresponds to estimating the optimal Lagrange multipliers μ_1 and μ_2 for implicit constraint thresholds ϵ_1 and ϵ_2.
2) The hyperparameter tuning process effectively searches for multiplier values that balance constraint satisfaction (feasibility) with objective optimization (optimality).

E. Theoretical Implications

This constrained optimization perspective provides several theoretical advantages.

1) *Mathematical Rigor:* This establishes our method on solid optimization-theoretic foundations with guaranteed KKT optimality conditions.
2) *Hyperparameter Interpretability:* This reveals that λ_1 and λ_2 function as Lagrange multipliers, quantifying the "cost" of violating their respective constraints.
3) *Feasibility–Optimality Tradeoff:* The complementary slackness conditions explicitly characterize when constraints drive the optimization versus when objectives can be freely pursued.
4) *Generalization Framework:* This formulation naturally extends to incorporate additional constraints for robustness, fairness, or other desiderata in backdoor learning.

This theoretical foundation reinforces the practical effectiveness of our approach while providing principled guidance for hyperparameter selection in similar multiobjective backdoor learning scenarios.

APPENDIX C
RISK ASSESSMENT

Backdoor threats in VPL can arise from multiple stages of the pipeline. We distinguish between provider-originated compromises, third-party model zoo pollution, client-side poisoning in federated settings, and Internet injection during transmission. Table XIV summarizes these risks using a likelihood–impact matrix and outlines mitigation measures. Provider compromise refers to scenarios where the official distribution of VPs or models is malicious or has been tampered with. Third-party model zoos (e.g., public repositories) represent a distinct threat surface, as poisoned assets may propagate widely with limited provenance assurance. Client-side poisoning is particularly relevant in federated learning: malicious clients can inject backdoored prompts during collaborative training, and the decentralized nature of the setting complicates centralized validation. Finally, Internet injection captures risks during VP or model transmission, where secure channels and integrity verification are required to prevent adversarial tampering. Overall, the assessment highlights that while provider compromise is less likely, third-party model zoos and client-side poisoning present more probable and impactful risks in practice.

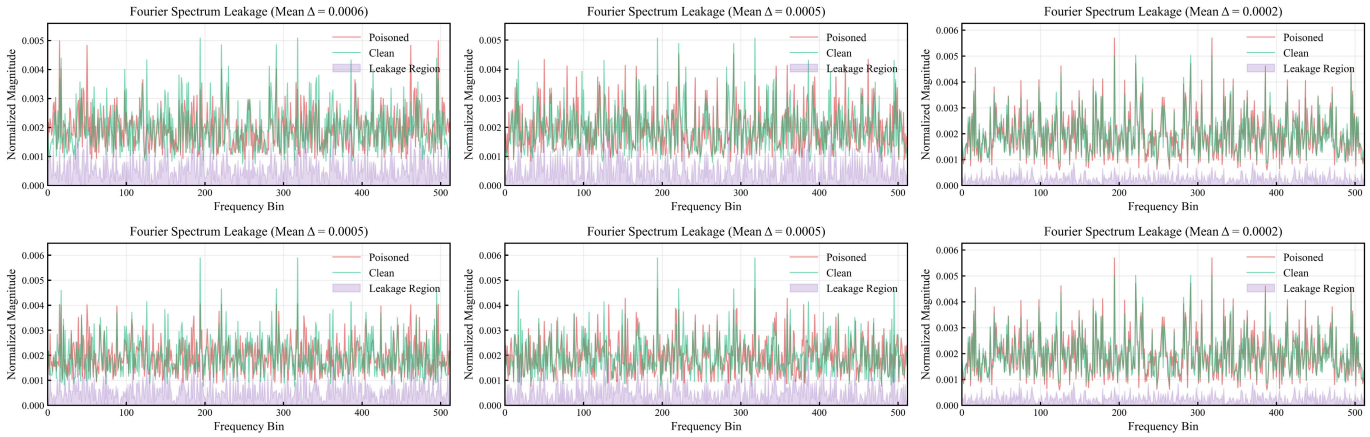

Fig. 21. Visualization of FSL for clean and poisoned embeddings in SBC. The frequency spectra are obtained by applying a 2-D FFT on embedding representations and visualized on a log scale. Both clean and poisoned spectra exhibit almost identical frequency distributions, suggesting that the backdoor perturbations do not introduce observable spectral deviations while maintaining alignment in latent space.

Indicators of Compromise (IoC) Checklist: To operationalize monitoring and defense, we recommend an IoC checklist for security teams.

1) *VP Hash Attestation:* Maintain cryptographic signatures for VPs and pretrained models.
2) *Repository Provenance Check:* Validate the origin of assets retrieved from third-party model zoos.
3) *Anomaly Tracing:* Monitor activation patterns and embedding distributions for deviations linked to triggers.
4) *Cross-Client Consistency:* Detect suspicious updates or trigger clustering across clients.

This checklist supports practical deployment of the risk assessment matrix by linking identified risks to actionable detection and mitigation strategies.

APPENDIX D
SUPPLEMENTARY EXPERIMENT

A. Fourier-Spectrum Leakage Analysis for SBC

To further substantiate the claim that the poisoned embeddings in SBC are imperceptible yet aligned, we perform a Fourier-spectrum leakage (FSL) analysis to examine the frequency-domain consistency between clean and poisoned representations. Specifically, we apply a 2-D fast Fourier transform (FFT) to the embedding features obtained from the global model and compute three quantitative measures: mean leakage (ML), Kullback–Leibler (KL) divergence, and cosine similarity between the normalized amplitude spectra.

As shown in Fig. 21, the spectral responses of poisoned and clean embeddings exhibit nearly identical frequency patterns, indicating that the injected backdoor signals do not cause noticeable high-frequency artifacts or energy shifts. This observation suggests that the backdoor perturbations remain imperceptible in the spectral domain.

Table XV provides a quantitative comparison under different perturbation budgets ($\epsilon = 8/255$ and $\epsilon = 4/255$). The poisoned samples yield consistently small ML and KL values while maintaining high cosine similarity (≈0.98), confirming that the poisoned embeddings are well-aligned with clean ones in their spectral characteristics. These results further reinforce that SBC achieves subtle and consistent embedding manipulation without disrupting the overall representational structure.

TABLE XV
QUANTITATIVE RESULTS OF FSL ANALYSIS FOR SBC

Lable	$\epsilon = 8/255$			$\epsilon = 4/255$		
	ML	KL	Cosine	ML	KL	Cosine
0,1	0.0005	0.06	0.94	0.0005	0.05	0.94
0,2	0.0005	0.05	0.94	0.0005	0.05	0.94
0,3	0.0006	0.08	0.92	0.0006	0.08	0.92
0,poison	0.0002	0.01	0.99	0.0002	0.01	0.98

TABLE XVI
SENSITIVITY ANALYSIS OF INNER TRIGGER LEARNING RATE (lrt) AND OUTER PROMPT LEARNING RATE (lr) ON SBC PERFORMANCE METRICS (ACC AND ASR), REFLECTING THE SYSTEM'S STABILITY AND IMPERCEPTIBILITY WITHIN HOSTILE ENVIRONMENTS

learning rate	$lr_v = 20$	$lr_v = 40$	$lr_v = 60$
$lr_t = 0.001$	93.07(99.76)	93.22(99.86)	93.26(99.88)
$lr_t = 0.05$	93.30(99.58)	93.30(99.80)	93.72(99.85)
$lr_t = 0.1$	93.52(99.70)	93.36(99.77)	93.58(99.68)

B. Robustness Analysis of SBC Under a Bilevel Poisoning Setting

To investigate the hierarchical robustness of SBC, we model it as a bilevel optimization framework, where the inner loop aligns attacker-side trigger embeddings and the outer loop simulates the victim-side prompt adaptation under both provider-driven and Internet-scale poisoning settings. This formulation captures the interaction between attacker optimization and victim learning dynamics.

Table XVI presents the sensitivity analysis with respect to the inner trigger learning rate (lr_t) and the outer prompt learning rate (lr_v). Results demonstrate that variations in either learning rate have only a marginal influence on overall accuracy (ACC) and ASR. SBC maintains consistently high

TABLE XVII

MULTITARGET SBC. THE LEFT AXIS INDICATES THE TARGET CLASS OF EACH IMPLANTED BACKDOOR. TARGETS 1–3 CORRESPOND TO THE ASRS OF THE FIRST, SECOND, AND THIRD BACKDOOR TARGETS, RESPECTIVELY

Class	ACC	ASR		
		Target 1	Target 2	Target 3
1, 2, 3	92.12	98.28	99.27	98.21
4, 5, 6	91.95	98.73	98.68	98.20
7, 8, 9	91.98	96.26	98.38	95.94
1, 3, 5	92.27	98.85	99.06	98.50
2, 4, 6	92.47	98.52	98.97	98.62

TABLE XVIII

PERFORMANCE OF MULTITARGET BACKDOOR ATTACKS UNDER CLASS-IMBALANCED SETTINGS. THE COLUMN PROPORTION INDICATES THE RATIO OF EACH TARGET CLASS WITHIN THE BACKDOORED DATA

Class	Proportion	ACC	ASR		
			Target 1	Target 2	Target 3
1, 3, 5	2:3:5	92.26	98.68	98.35	99.10
2, 4, 6	2:3:5	91.86	96.07	98.85	99.11
1, 3, 5	1.5:3.5:5	92.10	98.52	99.10	99.28
2, 4, 6	1.5:3.5:5	91.86	96.63	99.03	99.70
1, 2, 3	1.5:3.5:5	92.10	94.63	98.34	99.00
4, 5, 6	1.5:3.5:5	92.23	92.33	99.17	99.86

ACC (>93%) and ASR (>99.5%) across all configurations, validating the robustness of its bilevel optimization process.

These observations suggest that SBC's trigger embedding strategy adapts well to changes in optimization hyperparameters, exhibiting resilience and stability under diverse conditions. Such insensitivity to step-size variations highlights the practicality of SBC in real-world adversarial environments where poisoning strategies may evolve dynamically.

C. Extension to Multitarget Backdoor Settings

To further evaluate the generality and robustness of the proposed attack, we extend our framework from a single-target configuration to a *multitarget* backdoor setting. In this configuration, the adversary constructs class-specific triggers for each selected target and injects all triggers into the same poisoned training set. This multitarget extension allows us to examine: 1) the scalability of the backdoor mechanism when multiple target mappings are trained concurrently and 2) the potential impact of class-imbalance and collateral effects on nontarget categories. Empirically, we evaluate both the clean-task performance (ACC) to ensure benign-task preservation and the per-target ASR to quantify the integrity of each injected mapping.

In the experiments, we select three distinct target classes (denoted as Targets 1–3) and jointly inject their corresponding class-specific triggers under an identical global training configuration. The results, summarized in Table XVII, show that the model maintains high CA, indicating minimal degradation in its primary task. Meanwhile, all target classes achieve consistently high ASR values, suggesting that our multitarget design maintains strong activation fidelity without significant interference or mutual suppression among different targets. In addition, Table XVIII demonstrates that the proposed method can effectively scale to multitarget backdoor scenarios and remains resilient under moderate class-imbalance conditions.

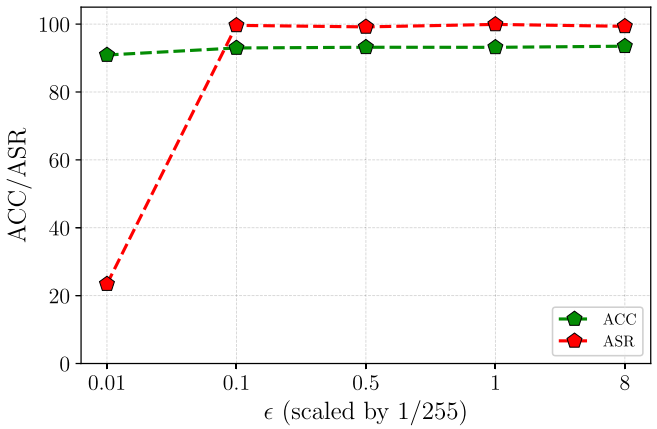

Fig. 22. Performance of the proposed SBC attack under different visibility (perturbation) budgets ϵ. Both clean accuracy (ACC) and ASR are reported. When ϵ is too small, the backdoor activation is suppressed, leading to low ASR and slightly reduced ACC due to indistinguishable gradients between clean and poisoned data. As ϵ increases beyond $0.1/255$, both ACC and ASR improve, showing that SBC maintains robustness and a balanced stealth–effectiveness tradeoff across perturbation levels.

D. Evaluation on Trigger Visibility and Perturbation Budget

To further examine the stealthiness boundary of our proposed backdoor attack, we investigate how different visibility constraints on the trigger affect the balance between attack effectiveness and model performance. Beyond the baseline configuration used in the main results, we introduce additional perturbation budgets with visibility constraints $\epsilon \in \{0.01/255, 0.1/255, 0.5/255, 1/255\}$ to generate triggers with varying degrees of imperceptibility.

As shown in Fig. 22, when the perturbation budget is extremely small (e.g., $\varepsilon \leq 0.01/255$), the ASR decreases sharply, and the clean-task accuracy (ACC) also suffers a slight drop. This behavior arises because the perturbation becomes too weak to introduce a distinguishable discrepancy between clean and poisoned samples, preventing the model from effectively internalizing the backdoor behavior.

In contrast, once the perturbation budget enters a moderate range (e.g., $\varepsilon > 0.1/255$), both ACC and ASR improve consistently. In this setting, the perturbation remains visually imperceptible to the human eye but becomes sufficiently expressive in the feature space to form a stable trigger-to-target mapping without interfering with normal optimization. Notably, across a broad practical interval ($0.1/255 < \varepsilon \leq 8/255$), the injected noise maintains high stealthiness while preserving strong backdoor effectiveness.

These observations highlight that the proposed SBC attack is robust across a range of realistic perturbation budgets and achieves a favorable balance between stealthiness and attack strength under practical visibility constraints.

REFERENCES

[1] F. Bie et al., "RenAIssance: A survey into AI text-to-image generation in the era of large model," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 3, pp. 2212–2231, Mar. 2025.

[2] C. Sima et al., "DriveLM: Driving with graph visual question answering," in *Proc. Eur. Conf. Comput. Vis.*, 2023, pp. 256–274.

[3] Y. Yang, Y.-Q. Yang, G. Ren, and B.-G. Yu, "Hierarchically trusted evidential fusion method with consistency learning for multimodal language understanding," *Knowl.-Based Syst.*, vol. 312, Mar. 2025, Art. no. 113164.

[4] D. A. Hafeth, M. Al-Khafajiy, and S. Kollias, "EdgeScan for IoT contextual understanding with edge computing and image captioning," *IEEE Internet Things J.*, vol. 12, no. 6, pp. 6519–6535, Mar. 2025.

[5] L. Wang, H. Chen, Y. Liu, and Y. Lyu, "Regular constrained multimodal fusion for image captioning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 11, pp. 11900–11913, Nov. 2024.

[6] M. I. Zulfiqar and I. Younis, "Enhanced security paradigms: Converging IoT and biometrics for advanced locker protection," *IEEE Internet Things J.*, vol. 11, no. 20, pp. 33811–33819, Oct. 2024.

[7] S. Li, L. Fei, B. Zhang, X. Ning, and L. Wu, "Hand-based multimodal biometric fusion: A review," *Inf. Fusion*, vol. 109, Sep. 2024, Art. no. 102418.

[8] Z. Han, C. Gao, J. Liu, J. Zhang, and S. Q. Zhang, "Parameter-efficient fine-tuning for large models: A comprehensive survey," 2024, *arXiv preprint arXiv:2403.14608*.

[9] Q. Wu, J. Qi, D. Zhang, H. Zhang, and J. Tang, "Fine-tuning for few-shot image classification by multimodal prototype regularization," *IEEE Trans. Multimedia*, vol. 26, pp. 8543–8556, 2024.

[10] Y. Zhang, K. Zhou, and Z. Liu, "Neural prompt search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 7, pp. 5268–5280, Jul. 2025.

[11] A. Liu et al., "CFPL-FAS: Class free prompt learning for generalizable face anti-spoofing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 222–232.

[12] J. Xie et al., "Knowledge-based dynamic prompt learning for multi-label disease diagnosis," *Knowl.-Based Syst.*, vol. 286, Feb. 2024, Art. no. 111395.

[13] B. Breve, G. Cimino, and V. Deufemia, "Hybrid prompt learning for generating justifications of security risks in automation rules," *ACM Trans. Intell. Syst. Technol.*, vol. 15, no. 5, pp. 1–26, Oct. 2024.

[14] Q. Cao, Z. Xu, Y. Chen, C. Ma, and X. Yang, "Domain-controlled prompt learning," in *Proc. AAAI Conf. Artif. Intell.*, 2024, vol. 38, no. 2, pp. 936–944.

[15] J. Gao et al., "LAMM: Label alignment for multi-modal prompt learning," in *Proc. AAAI Conf. Artif. Intell.*, 2024, vol. 38, no. 3, pp. 1815–1823.

[16] M. Cai et al., "ViP-LLaVA: Making large multimodal models understand arbitrary visual prompts," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 12914–12923.

[17] J. Wu et al., "Visual prompting in multimodal large language models: A survey," 2024, *arXiv:2409.15310*.

[18] F. Li et al., "Visual in-context prompting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 12861–12871.

[19] Y. Lei, J. Li, Z. Li, Y. Cao, and H. Shan, "Prompt learning in computer vision: A survey," *Frontiers Inf. Technol. & Electron. Eng.*, vol. 25, no. 1, pp. 42–63, 2024.

[20] X. Cai et al., "ControlMLLM: Training-free visual prompt learning for multimodal large language models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, pp. 45206–45234.

[21] Y. Chen, Y. Pant, H. Yang, T. Yao, and T. Meit, "VP3D: Unleashing 2D visual prompt for Text-to-3D generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 4896–4905.

[22] S. Li, B. Li, B. Sun, and Y. Weng, "Towards visual-prompt temporal answer grounding in instructional video," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 12, pp. 8836–8853, Dec. 2024.

[23] X. Liu, J. Wu, W. Yang, X. Zhou, and T. Zhang, "Multi-modal attribute prompting for vision-language models," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 11, pp. 11579–11591, Nov. 2024.

[24] Y. Zhang, H. Fan, W. Ji, Y. Wong, R. Zimmermann, and Y. Yang, "Prompt-aware adapter: Learning adaptive visual tokens for multimodal large language models," *IEEE Trans. Artif. Intell.*, pp. 1–10, 2025.

[25] B. X. B. Yu et al., "Visual tuning," *ACM Comput. Surveys*, vol. 56, no. 12, pp. 1–38, 2024.

[26] H. Bahng, A. Jahanian, S. Sankaranarayanan, and P. Isola, "Exploring visual prompts for adapting large-scale models," 2022, *arXiv:2203.17274*.

[27] M. Jia et al., "Visual prompt tuning," in *Proc. Eur. Conf. Comput. Vis.*, Oct. 2022, pp. 709–727.

[28] J. Jia, Y. Liu, and N. Z. Gong, "BadEncoder: Backdoor attacks to pre-trained encoders in self-supervised learning," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2022, pp. 2043–2059.

[29] C. Li et al., "An embarrassingly simple backdoor attack on self-supervised learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 4367–4378.

[30] Z. Zhou et al., "SecFFT: Safeguarding federated fine-tuning for large vision language models against covert backdoor attacks in IoRT networks," *IEEE Internet Things J.*, vol. 12, no. 9, pp. 11383–11396, May 2025.

[31] J. Bai, K. Gao, S. Min, S.-T. Xia, Z. Li, and W. Liu, "BadCLIP: Trigger-aware prompt learning for backdoor attacks on CLIP," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 24239–24250.

[32] H. Huang, Z. Zhao, M. Backes, Y. Shen, and Y. Zhang, "Prompt backdoors in visual prompt learning," 2023, *arXiv:2310.07632*.

[33] B. Li, Y. Cai, H. Li, F. Xue, Z. Li, and Y. Li, "Nearest is not dearest: Towards practical defense against quantization-conditioned backdoor attacks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 24523–24533.

[34] T. Huynh, D. Nguyen, T. Pham, and A. Tran, "COMBAT: Alternated training for effective clean-label backdoor attacks," in *Proc. AAAI Conf. Artif. Intell.*, 2024, vol. 38, no. 3, pp. 2436–2444.

[35] M. Fan, Z. Si, X. Xie, Y. Liu, and T. Liu, "Text backdoor detection using an interpretable RNN abstract model," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 4117–4132, 2021.

[36] Y. Yu et al., "Robust and transferable backdoor attacks against deep image compression with selective frequency prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 3, pp. 1674–1693, Mar. 2025.

[37] Y. Gao, Y. Li, X. Gong, Z. Li, S.-T. Xia, and Q. Wang, "Backdoor attack with sparse and invisible trigger," *IEEE Trans. Inf. Forensics Security*, vol. 19, pp. 6364–6376, 2024.

[38] W. Fan, H. Li, W. Jiang, M. Hao, S. Yu, and X. Zhang, "Stealthy targeted backdoor attacks against image captioning," *IEEE Trans. Inf. Forensics Security*, vol. 19, pp. 5655–5667, 2024.

[39] X. Han et al., "Backdooring multimodal learning," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2024, pp. 3385–3403.

[40] Z. Ran, Y. Yao, W. Li, W. Yang, W. Li, and Y. Wu, "Precise defense approach against small-scale backdoor attacks in industrial Internet of Things," *IEEE Internet Things J.*, vol. 12, no. 5, pp. 5742–5754, Mar. 2025.

[41] X. Gan, H. Wang, X. Li, Z. Liu, H. Jiang, and J. Wang, "A multitarget backdoor attack against automatic modulation recognition for IoT wireless signals," *IEEE Internet Things J.*, vol. 12, no. 14, pp. 27588–27605, Jul. 2025.

[42] J. Zhang, Z. Wang, Z. Ma, and J. Ma, "Adaptive robust learning against backdoor attacks in smart homes," *IEEE Internet Things J.*, vol. 11, no. 13, pp. 23906–23916, Apr. 2024.

[43] Q. Wang et al., "BadSTR: Backdoor attack on scene text recognition in IoT," *IEEE Internet Things J.*, vol. 12, no. 18, pp. 38526–38539, Sep. 2025.

[44] H. Souri, M. Goldblum, L. Fowl, R. Chellappa, and T. Goldstein, "Sleeper agent: Scalable hidden trigger backdoors for neural networks trained from scratch," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 19165–19178.

[45] T. Gu, K. Liu, B. Dolan-Gavitt, and S. Garg, "BadNets: Evaluating backdooring attacks on deep neural networks," *IEEE Access*, vol. 7, pp. 47230–47244, 2019.

[46] N. Carlini and A. Terzis, "Poisoning and backdooring contrastive learning," 2021, *arXiv:2106.09667*.

[47] D. Guo, M. Hu, Z. Guan, J. Guo, T. Hartvigsen, and S. Li, "Backdoor in seconds: Unlocking vulnerabilities in large pre-trained models via model editing," 2024, *arXiv:2410.18267*.

[48] S. Liang, M. Zhu, A. Liu, B. Wu, X. Cao, and E.-C. Chang, "BadCLIP: Dual-embedding guided backdoor attack on multimodal contrastive learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 24645–24654.

[49] W. Yang, J. Gao, and B. Mirzasoleiman, "Better safe than sorry: Pre-training clip against targeted data poisoning and backdoor attacks," 2023, *arXiv:2310.05862*.

[50] J. Zhang, H. Liu, J. Jia, and N. Z. Gong, "Data poisoning based backdoor attacks to contrastive learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 24357–24366.

[51] X. Xing, M. Xu, Y. Bai, and D. Yang, "A clean-label graph backdoor attack method in node classification task," *Knowl.-Based Syst.*, vol. 304, Nov. 2024, Art. no. 112433.

[52] G. Wang et al., "One-to-multiple clean-label image camouflage (OmClic) based backdoor attack on deep learning," *Knowledge-Based Syst.*, vol. 288, Mar. 2024, Art. no. 111456.

[53] W. Yang, Y. Lin, P. Li, J. Zhou, and X. Sun, "Rethinking stealthiness of backdoor attack against NLP models," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics 11th Int. Joint Conf. Natural Lang. Process. (Volume 1: Long Papers)*, 2021, pp. 5543–5557.
[54] H. Zhang et al., "Invisible backdoor attack against self-supervised learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2025, pp. 25790–25801.
[55] W. Yang, Y. Lin, P. Li, J. Zhou, and X. Sun, "RAP: Robustness-aware perturbations for defending against backdoor attacks on NLP models," 2021, *arXiv:2110.07831*.
[56] R. Zheng, R. Tang, J. Li, and L. Liu, "Data-free backdoor removal based on channel lipschitzness," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 175–191.
[57] B. Wang et al., "Neural cleanse: Identifying and mitigating backdoor attacks in neural networks," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2019, pp. 707–723.
[58] Y. Li, X. Lyu, N. Koren, L. Lyu, B. Li, and X. Ma, "Neural attention distillation: Erasing backdoor triggers from deep neural networks," 2021, *arXiv:2101.05930*.
[59] D. Wu and Y. Wang, "Adversarial neuron pruning purifies backdoored deep models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 16913–16925.
[60] H. Bansal, F. Yin, N. Singhi, A. Grover, Y. Yang, and K.-W. Chang, "CleanCLIP: Mitigating data poisoning attacks in multimodal contrastive learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 112–123.
[61] W. Yang, J. Gao, and B. Mirzasoleiman, "Robust contrastive language-image pretraining against data poisoning and backdoor attacks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, 2023, pp. 10678–10691.
[62] C. Schuhmann et al., "LAION-400M: Open dataset of CLIP-filtered 400 million image-text pairs," 2021, *arXiv:2111.02114*.
[63] B. Zhang, P. Zhang, X. Dong, Y. Zang, and J. Wang, "Long-CLIP: Unlocking the long-text capability of CLIP," in *Proc. Eur. Conf. Comput. Vis.*, 2024, pp. 310–325.
[64] H. Luo et al., "Clip4clip: An empirical study of clip for end to end video clip retrieval," 2021, *arXiv:2104.08860*.
[65] Z. Sun et al., "Alpha-CLIP: A CLIP model focusing on wherever you want," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 13019–13029.
[66] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *Proc. ICML*, vol. 11, 2011, pp. 689–696.
[67] P. Xu, X. Zhu, and D. A. Clifton, "Multimodal learning with transformers: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 10, pp. 12113–12132, Oct. 2023.
[68] Q. Sun et al., "Generative multimodal models are in-context learners," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 14398–14409.
[69] W. Ma, Q. Chen, T. Zhou, S. Zhao, and Z. Cai, "Using multimodal contrastive knowledge distillation for video-text retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 10, pp. 5486–5497, Oct. 2023.
[70] L. He, Z. Wang, L. Wang, and F. Li, "Multimodal mutual attention-based sentiment analysis framework adapted to complicated contexts," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 12, pp. 7131–7143, Dec. 2023.
[71] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[72] K. Han et al., "A survey on vision transformer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 87–110, Jan. 2023.
[73] A. Nguyen and A. Tran, "Wanet–imperceptible warping-based backdoor attack," 2021, *arXiv:2102.10369*.
[74] K. Doan, Y. Lao, W. Zhao, and P. Li, "LIRA: Learnable, imperceptible and robust backdoor attacks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 11946–11956.
[75] Z. Zhao, X. Chen, Y. Xuan, Y. Dong, D. Wang, and K. Liang, "DEFEAT: Deep hidden feature backdoor attacks by imperceptible perturbation and latent representation constraints," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 15192–15201.
[76] J. V. Stone, *Bayes' Rule: A Tutorial Introduction to Bayesian Analysis*. Sheffield, U.K.: Sebtel Press, 2013.
[77] D. Tang, X. Wang, H. Tang, and K. Zhang, "Demon in the variant: Statistical analysis of DNNs for robust backdoor contamination detection," in *Proc. 30th USENIX Secur. Symp.*, 2021, pp. 1541–1558.
[78] Y. Gao, C. Xu, D. Wang, S. Chen, D. C. Ranasinghe, and S. Nepal, "STRIP: A defence against trojan attacks on deep neural networks," in *Proc. 35th Annu. Comput. Secur. Appl. Conf.*, Dec. 2019, pp. 113–125.
[79] Y. Gao et al., "Design and evaluation of a multi-domain trojan detection method on deep neural networks," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 4, pp. 2349–2364, Jul. 2022.