Image Triaging for Budget-Aware Universal Attacks on Vision-Language Models

Anonymous Authors[1]

Abstract

Vision-Language Models (VLMs) are typically evaluated under adversarial settings where perturbations are applied to all inputs. In practice, attackers often face budget constraints, making *which* inputs to attack as critical as *how* the perturbation is constructed. We study budgeted deployment of targeted universal adversarial perturbations (UAPs), where a fixed perturbation must be selectively applied to maximize attack success under limited access. We propose an image-triaging framework that predicts transfer vulnerability from clean, image-level features, trained using surrogate attack outcomes and requiring no model queries at deployment time. This enables ranking candidate inputs and allocating a limited attack budget to the most vulnerable samples. Across six VLMs and two UAP methods on a road accident monitoring task, our approach substantially improves attack efficiency over random selection, achieving up to 73–97% attack success at only 1% budget. Our analysis reveals that transfer vulnerability is largely input-intrinsic: simple low-level image statistics capture most of the triaging gains, and predictors trained on one UAP method generalize effectively to another. These findings suggest that attack effectiveness in constrained settings is governed as much by input selection as by perturbation design, and that shared vulnerability signals can be exploited without access to the victim model.

1. Introduction

Vision-Language Models (VLMs) have recently achieved strong performance on image captioning (Mokady et al., 2021), visual question answering (Li et al., 2022; Alayrac et al., 2022; Liu et al., 2023), and multimodal reasoning (Zhang et al., 2024b). Their ability to jointly process images and text makes them attractive for applications that require high-level situational understanding, including autonomous driving and road monitoring (Xu et al., 2024; Zhou et al., 2024). At the same time, recent work shows that vision encoders commonly used in VLMs (e.g., CLIP-based models) can be vulnerable to adversarial attacks (Lu et al., 2023; Yin et al., 2023; Zhang et al., 2022). Of particular concern are *targeted universal adversarial perturbations* (UAPs): a single perturbation, crafted once offline, can induce attacker-chosen outputs across many inputs without per-instance optimization (Fang et al., 2025). This "craft once, apply everywhere" property makes UAPs a realistic and scalable threat, especially in deployment scenarios where the attacker cannot adapt the perturbation at inference time. Recent work has shown that targeted UAP attacks (Huang et al., 2025; Zhang et al., 2025) generated using CLIP-style surrogate encoders can transfer to modern VLMs, which raises concerns for VLMs deployed in safety-critical applications.

Such adversarial attacks are often evaluated by perturbing all inputs and reporting attack success rate (ASR). However, this protocol implicitly assumes unconstrained access to the input stream; in practice, physical access constraints, detection risk, or limited control over the input stream may restrict how many inputs an attacker can perturb.

The problem of selecting the most transferable adversarial examples has received limited attention. The closest prior work is that of Levy et al. (Levy et al., 2024), who formalize transferability ranking for instance-level attacks on image classifiers, scoring adversarial examples by aggregating surrogate model predictions. A key limitation of this approach is that the ranking is tied to a specific perturbation, and does not extend to universal attacks where the perturbation is fixed, and variation in success arises from the input content alone (besides perturbation strength). To our knowledge, no prior work studies **budgeted** deployment of universal attacks on VLMs, where the attacker must decide which inputs to perturb from a candidate pool without victim feedback.

We address this gap by casting budgeted UAP deployment as a ranking problem. We propose an **image-triaging** framework that learns to predict transfer vulnerability from image-level features and surrogate attack outcomes, enabling selec-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

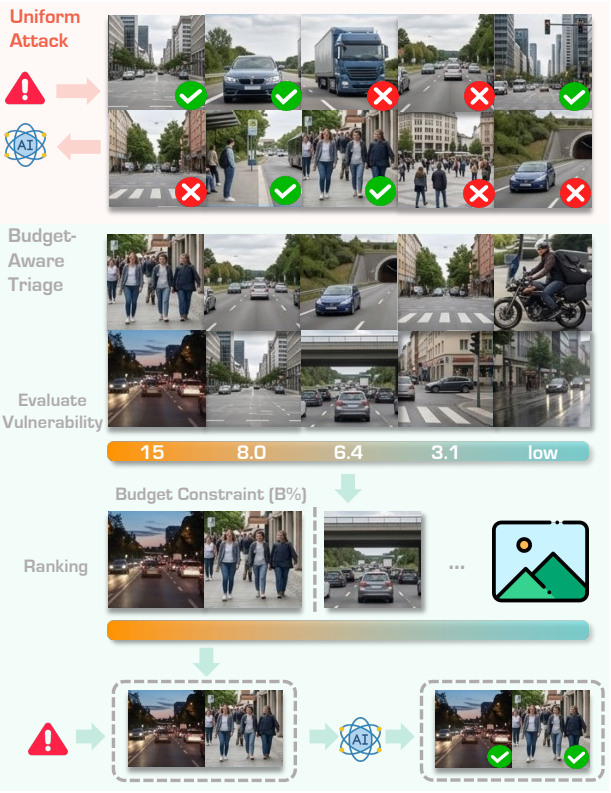


Figure 1. **Budget-aware triaging for targeted UAP deployment.** Existing attacks uniformly perturb all inputs, resulting in inconsistent success because UAP effectiveness varies across scenes. Our approach first estimates input **vulnerability**, ranks candidate scenes, and allocates a limited attack budget to the most vulnerable inputs, improving attack success per attempt.

tive deployment of a fixed UAP under tight attempt budgets (Figure 1). Unlike prior ranking approaches that score adversarial examples through surrogate model inference, our predictor operates on *clean* inputs using lightweight features, requiring no surrogate evaluation at deployment time and generalizing across both victim models and UAP construction methods.

Across six VLMs on a road accident monitoring task, vulnerability-aware selection substantially improves attack efficiency. Under XTransfer (Huang et al., 2025), our triaging strategy raises ASR_B from a random-selection baseline of 24–58% to 73–97% when only 1% of inputs are perturbed (B=0.01). We further show that the vulnerability signal is largely intrinsic to the input: low-level image statistics alone capture most of the triaging gain, and regressors trained on one UAP method transfer effectively to another (mean ROC-AUC drop of 0.028).

In summary, our main contributions are as follows:

- We formalize **budgeted deployment of targeted UAPs** against VLMs and introduce ASR_B, the attack success rate under a normalized deployment budget.
- We propose an **image-triaging framework** that learns to rank inputs by transfer vulnerability using image-level cues and surrogate attack outcomes, enabling victim-free, perturbation-agnostic selection under tight budgets.
- Extensive experiments across six VLMs and two UAP frameworks (*XTransfer* and *AnyAttack*) show consistent improvements over random selection and reveal that transfer vulnerability is largely intrinsic to the input and transferable across attack constructions.

2. Related Work

Adversarial Attacks on VLMs. Adversarial attacks on VLMs can be broadly categorized into *image-specific* attacks, where perturbations are optimized per input (Zhang et al., 2022; Qi et al., 2023; Yin et al., 2023), and *universal* attacks, where a single perturbation is designed to generalize across many images (Zhang et al., 2024a; Huang et al., 2025; Zhang et al., 2025). Image-specific attacks typically achieve higher success rates but require per-input optimization, making them less practical when attackers cannot adapt perturbations at deployment time. Universal adversarial perturbations (UAPs) better model this threat setting: a fixed perturbation can be applied repeatedly without additional computation.

Universal Adversarial Perturbations. Because UAPs are input-agnostic by construction, their effectiveness varies substantially across images depending on scene content and visual characteristics. This is the key observation motivating our proposed selective deployment strategy. UAPs are typically generated either through direct optimization of surrogate models (Zhang et al., 2022; Lu et al., 2023; Zhang et al., 2024a; Huang et al., 2025) or by training adversarial generators that produce transferable perturbations (Fang et al., 2025; Zhang et al., 2025). Recent methods advance both paradigms: *XTransfer* (Huang et al., 2025) horizontally scales the number of surrogate models used in UAP optimization, while *AnyAttack* (Zhang et al., 2025) trains a generator via large-scale self-supervised pre-training. Both demonstrate transfer to auto-regressive VLMs despite relying solely on CLIP-based encoders during construction.

Attack Efficiency and Sample Selection. There have been numerous prior works on increasing the efficiency of adversarial attacks. Query-efficient attacks (Fan et al., 2024) aim to reduce the number of model queries required to *construct* a perturbation; our work addresses a complementary constraint at *deployment*, where the perturbation is fixed, and the attacker is limited in how many inputs it can perturb. The closest work to ours is Levy et al. (Levy et al., 2024), who formalize ranking adversarial examples by expected transferability, scoring candidates by averaging surrogate

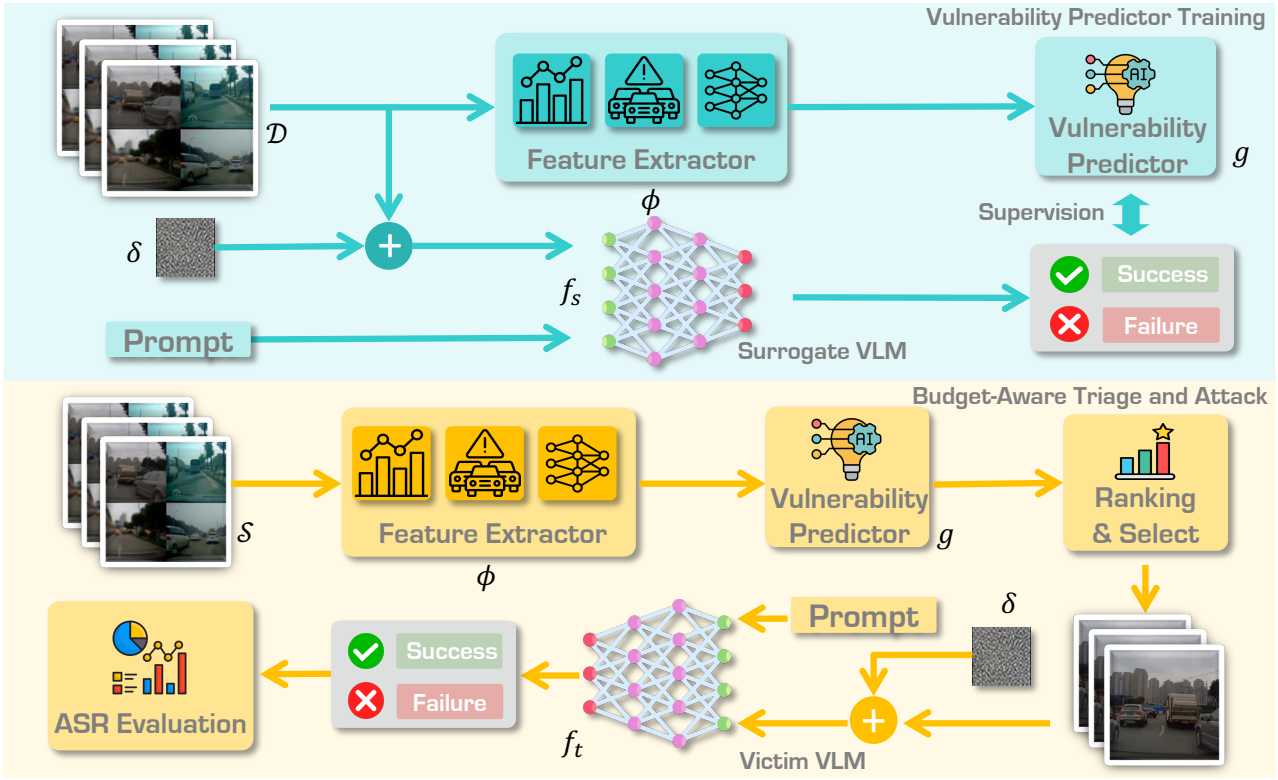


Figure 2. **Framework overview.** Top: offline training of a **vulnerability predictor** from surrogate UAP outcomes. Each input in the surrogate set $\mathcal{D}$ is mapped to an image-level representation $\phi(x)$ using [1]*low-level statistics*, [2]*detector-based semantic cues*, and [3]*pretrained deep visual features*, and a fixed targeted perturbation δ is evaluated on the surrogate VLM f_s to produce success/failure labels for training a lightweight predictor g. Bottom: **budgeted deployment** on the victim pool $\mathcal{S}$. The predictor scores and ranks all candidates by vulnerability likelihood, selects the top-$K = \lfloor Bn \rfloor$ under budget B, and applies δ only to the selected inputs before querying the victim VLM f_t.

softmax confidences. However, their approach views transferability as a property of the adversarial example, requiring surrogate inference on every candidate and tying the ranking to a specific perturbation. It also does not extend to universal attacks, where a fixed perturbation is applied across inputs, and variation in success arises from input content alone. Our work addresses both limitations. By learning a vulnerability predictor over clean image-level features, we show that transfer vulnerability is largely intrinsic to the input, is predictable without per-sample surrogate inference, and generalizable across UAP construction methods.

3. Budget-Aware Targeted UAP Deployment

Targeted UAPs are appealing for scalable attacks because a single, input-agnostic perturbation can be applied broadly without per-instance optimization. Because UAP success is input-dependent, indiscriminate deployment is inefficient under limited budgets. We therefore study **budgeted UAP deployment**: given a fixed perturbation and a constrained attempt budget, which inputs should be attacked to maximize realized targeted success.

We cast budgeted deployment as a **ranking** problem and propose a lightweight **triaging** framework, as shown in Figure 2. In the following sections, Section 3.1 formalizes the threat model and the budgeted objective. Section 3.2 learns a transfer-vulnerability triaging model from surrogate-attack outcomes, enabling victim-free scoring of candidate inputs. Section 3.3 executes the attack by ranking the deployment pool according to the vulnerability likelihood and perturbing only the most vulnerable inputs within budget.

3.1. Threat Model and Budgeted Objective

Let $\mathcal{S} = \{x_i\}_{i=1}^{n}$ denote a fixed deployment pool evaluated by a victim VLM f_t. The attacker cannot control which inputs appear in $\mathcal{S}$, and has no access to the victim's parameters, gradients, or architecture; moreover, the attacker cannot query f_t for feedback during deployment. Instead, the attacker has access to a surrogate VLM f_s, a surrogate dataset $\mathcal{D}$, and a single fixed targeted UAP δ crafted offline by any existing method.

Given a normalized deployment budget $B \in [0, 1]$, the at-

tacker may perturb at most $K = \lfloor Bn \rfloor$ inputs in $\mathcal{S}$. The attacker selects a subset $\mathcal{S}_B \subseteq \mathcal{S}$ with $|\mathcal{S}_B| = K$ to maximize the realized targeted success rate:

$$\text{ASR}_B = \frac{1}{K} \sum_{x_i \in \mathcal{S}_B} \mathbb{1}[f_t(x_i + \delta) = y_t], \tag{1}$$

where y_t is a predefined target output (specified by the attacker) and $\mathbb{1}[\cdot]$ is the indicator function.

The perturbation δ is generated once offline using any targeted UAP construction procedure on surrogate resources, subject to a norm constraint $\|\delta\| \leq \epsilon$. Our framework is agnostic to the specific construction method: throughout deployment, δ remains fixed and cannot be adapted per input. Consequently, under a fixed δ, attack efficiency is governed not only by perturbation strength but also by *which* inputs are selected for deployment.

3.2. Learning a Vulnerability Predictor

Since the attacker cannot query the victim model during deployment, budgeted success hinges on estimating a priori which inputs are most likely to satisfy $f_t(x + \delta) = y_t$ under a fixed perturbation δ. Denote the ground truth vulnerability scoring function under our task as $g^* : \mathcal{X} \to [0, 1]$. Our goal is to estimate g^*. We train a lightweight regressor $g(\cdot)$ via maximum likelihood estimation to approximate the ground truth vulnerability g^*. A direct estimation is challenging because evaluating an image $x \in \mathcal{X}$ yields a strictly binary label (whether the VLM is fooled). To provide a continuous target variable for $g(\cdot)$, we construct a surrogate vulnerability score based on temporal aggregation. Assuming that vulnerability is locally consistent across adjacent frames in a traffic video, we sample a temporal window $\mathcal{N}(\cdot)$ of M consecutive frames centered around an anchor frame x. The continuous surrogate label $\bar{y}$ is computed as the expected success rate within the window:

$$\bar{y} = \mathbb{E}_{x' \sim \mathcal{N}} [\mathbb{1}[f_s(x' + \delta) = y_t]]$$

This aggregation transforms sparse binary feedback into a dense, continuous score $\bar{y} \in [0, 1]$. Finally, $g(\cdot)$ learns to predict this score from a feature representation $\phi(x)$ by minimizing the binary cross-entropy objective.

Feature Construction. For each input x, we compute the image-level representation $\phi(x)$ by concatenating three complementary feature families. We include (i) *low-level* image statistics to capture generic image quality and photometric cues (e.g., blur, contrast, and edge density); (ii) *semantic cues* from an off-the-shelf detector to summarize scene content and object presence that may modulate transferability; and (iii) *deep features* from a pretrained visual backbone to provide a compact, high-level representation beyond handcrafted statistics. This design balances informativeness and efficiency, allowing scalable scoring over large deployment pools. The specific choice of features used to instantiate $\phi(\cdot)$ is detailed in Section 4.1, as well as in the supplementary material.

Algorithm 1 Budget-Aware UAP Deployment

Require: Deployment dataset $\mathcal{S} = \{x_i\}_{i=1}^n$, perturbation δ, triaging model $g(\cdot)$, budget B
$K \leftarrow \lfloor Bn \rfloor$
for each $x_i \in \mathcal{S}$ **do**
  Compute score $s_i \leftarrow g(\phi(x_i))$
end for
Select top-K inputs according to $\{s_i\}$
for each selected input x_i **do**
  Deploy attack $x_i' \leftarrow x_i + \delta$
end for

3.3. Vulnerability-Guided Budget Allocation

At deployment time, the attacker has access to the deployment pool $\mathcal{S}$, the feature extractor $\phi(\cdot)$, the learned vulnerability predictor $g(\cdot)$, and the fixed perturbation δ. No victim feedback is available. Given budget B, the attacker may perturb at most $K = \lfloor Bn \rfloor$ inputs from $\mathcal{S} = \{x_i\}_{i=1}^n$. For each candidate $x_i \in \mathcal{S}$, we compute a vulnerability likelihood $v_i = g(\phi(x_i))$. The attacker then ranks all candidates in descending order of v_i and selects the top-K inputs:

$$\mathcal{S}_B = \texttt{TopK}\big(\{(x_i, v_i)\}_{i=1}^n, K\big). \tag{2}$$

Since v_i estimates success likelihood, selecting top-K maximizes expected successes. For each selected input $x_i \in \mathcal{S}_B$, we apply the fixed perturbation to form $x_i' = x_i + \delta$ and submit x_i' to the victim model f_t. We then evaluate the realized deployment performance, measured by ASR_B in Equation (1). This procedure is summarized in Algorithm 1.

Our deployment strategy explicitly decouples attack generation from attack execution: δ is generated once offline (Section 3.1), while the triaging model $g(\cdot)$ determines where the limited deployment budget is allocated. This decoupling is critical in our threat model, when δ cannot be adapted online and victim queries are unavailable, improving attack efficiency reduces to selecting inputs with high predicted transfer vulnerability. Moreover, because scoring uses lightweight features and a cheap predictor, the policy scales to large deployment pools.

4. Experiments

We consider a road-scene accident monitoring setting in which a VLM answers a binary prompt indicating whether an input depicts a traffic accident. Our experiments focus on inducing false-positive accident predictions via universal perturbations under a **budgeted threat model**, and quantify the gains from **vulnerability-based triaging**.

4.1. Experimental Setup

Datasets. We use driving video subsets from *BDD100K* (Yu et al., 2020) and *D*^{2}*-City* (Che et al., 2019), resulting in 5000 videos in total. As the dataset documentation provides no accident labels or accident-specific splits, we treat the sampled clips as non-accident scenes without further filtering and evaluate false-positive flips to the target accident label. We use *BDD100K* as the surrogate dataset $\mathcal{D}$ for UAP generation and triager training, and *D*^{2}*-City* as the deployment set $\mathcal{S}$, yielding a realistic cross-dataset shift.

Evaluation Protocol. Although VLMs make image-level predictions, practical monitoring aggregates decisions over short temporal windows. We therefore evaluate on temporal frame groups, each comprising 15 frames sampled at 0.2-second intervals. To ensure attack success reflects sustained model failure rather than isolated frame instability, we consider a group to be **successfully attacked** if at least 80% of its frames are correct on clean inputs, and at least 80% of those clean-correct frames flip to the target label under the universal perturbation. To ensure that we extract the features that are scene-representative, we also compute aggregate group-level statistics (min., max., mean, std.) of the features in Section 3.2 and include them in $\phi(\cdot)$. We report **Group ASR**, the fraction of groups that are successfully attacked over the full set. For budgeted deployment, we rank groups by predicted vulnerability, attack the top-K groups, and report **ASR**$_B$ on this selected subset.

Evaluated Models and Attacks. We evaluate six open-source VLMs that are architecturally diverse and achieve high clean accuracy on non-accident scenes: *Ovis2-8B* (Lu et al., 2024), *LLaVA-OneVision-7B* (Li et al., 2024), *Qwen2.5-VL-7B* (Bai et al., 2025), *InternVL3-14B* (Zhu et al., 2025), *Kimi-VL-A3B-Instruct* (Kimi Team et al., 2025), and *DeepSeek-VL2-small* (Wu et al., 2024). For brevity, we will refer to these models by their root names: Ovis, LLaVA, Qwen, InternVL, Kimi, and DeepSeek. We report the group-level accuracies of all six VLMs on unperturbed inputs in the **supplementary material**. To test attack-agnostic operation, we instantiate the framework with two representative UAP generators: *XTransfer* (Huang et al., 2025), which optimizes perturbations against an ensemble of held-out surrogate models, and *AnyAttack* (Zhang et al., 2025), which trains a generator for transferable perturbations. Both are trained on the *BDD100K* training split to induce false-positive accident predictions. We use $\epsilon = 12/255$ for XTransfer and $\epsilon = 20/255$ for AnyAttack to balance attack realism with sufficient vulnerability prevalence for stable regressor training. This design choice comes from our observation that XTransfer's stronger transferability yields roughly $\sim$ 0.2 at the lower bound, while AnyAttack requires a larger perturbation strength to achieve non-negligible prevalence ($\sim$0.09 at the maximum) which provides a complementary low-prevalence evaluation regime.

Triager Training and Baselines. We train a lightweight triager on an 81-dim feature representation that aggregates (i) *low-level image statistics* (e.g., brightness, edge density, blur), (ii) *object-centric semantics* from `yolov8l` detections (Jocher et al., 2023), and (iii) *deep features* from ResNet-50 embeddings (He et al., 2015). We select 24 features via recursive feature elimination and use *XGBoost* (Chen & Guestrin, 2016) for regression. As a baseline, we compare against uniform random selection of K groups from the deployment stream, whose expected ASR$_B$ equals the dataset-level Group ASR (i.e., vulnerability prevalence at $B = 1$). See supplementary material for details.

4.2. Cross-Model Transferability

We use *BDD100K* as the surrogate dataset $\mathcal{D}$ to generate UAPs and to train the triager, and *D*^{2}*-City* as the deployment set $\mathcal{S}$. For each surrogate VLM, we fit an XGBoost triaging model with group-level attack outcomes on *BDD100K* as supervision, and then apply it to rank groups in *D*^{2}*-City*. Figure 3 reports group ASR$_B$ for all victim-surrogate pairs under the **XTransfer** attack. Due to space constraints, the corresponding results under **AnyAttack** are deferred to the supplementary material. Across both attack regimes and surrogate-label choices, vulnerability-aware selection consistently improves over uniform random sampling, whose expected performance equals the dataset-level Group ASR. The gains are most pronounced in the low-budget regime. At $B = 0.01$, the selector achieves near-perfect precision for several victims (e.g., Kimi-VL-Instruct reaches ASR$_B$ up to 0.97) and remains substantially above random at $B \in \{0.05, 0.10\}$. Across the heatmaps, these improvements indicate that vulnerability-aware ranking can substantially concentrate attack effort onto a small subset of highly susceptible groups, yielding markedly higher success rates than uniform sampling under the same budget.

4.3. Oracle Analysis

Notably, the magnitude of improvement varies widely across victim models and surrogate choices. This variation indicates that budgeted attack efficiency is governed by two factors: (i) **vulnerability prevalence**, i.e., how many groups are intrinsically susceptible to the UAP under distribution shift, and (ii) **prediction quality**, i.e., how well the triager can identify those susceptible groups from surrogate supervision. We explicitly disentangle and analyze these effects to diagnose the source of heterogeneous gains across victim models, and to determine whether the bottleneck lies in vulnerability prevalence or in ranking quality.

Scarcity vs. Selection. To separate intrinsic vulnerability prevalence from ranking quality, we compare triage-based deployment against an oracle selector that always chooses

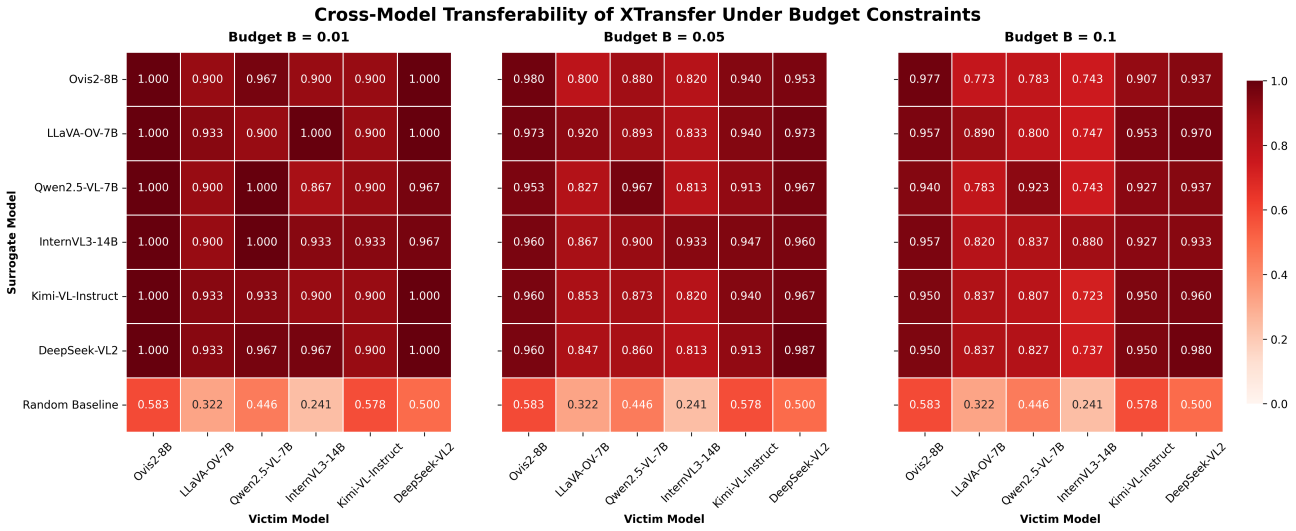


Figure 3. **Cross-model attack success rates under budget constraints with XTransfer.** Rows denote XGBoost regressors trained on BDD100K surrogate models and applied to D^2-City victim models. Results are shown for budgets of $B = \{0.01, 0.05, 0.10\}$.

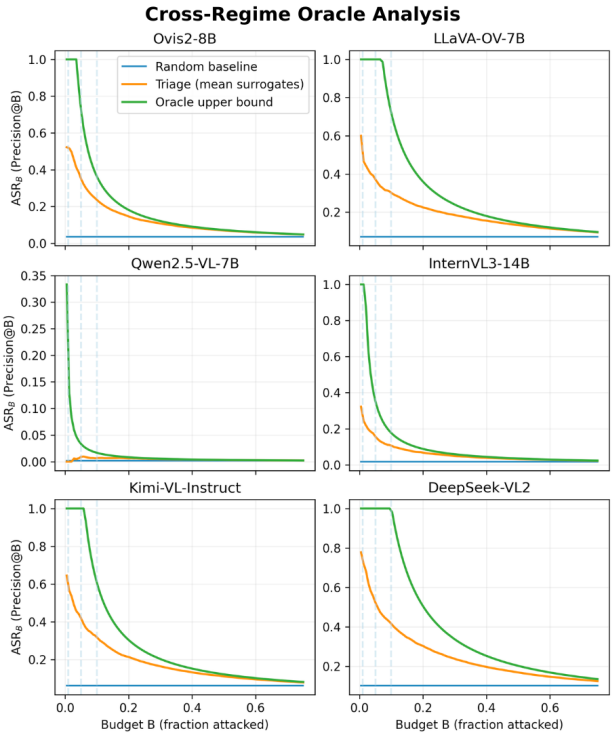


Figure 4. Cross-regime oracle analysis (Train: XTransfer regressors, Eval: AnyAttack). ASR$_B$ averaged over all surrogates.

the K truly vulnerable groups. Figure 4 presents this comparison across all six victim models under cross-regime evaluation, where XTransfer-trained regressors are applied to AnyAttack victims. The oracle curve in each panel reflects AnyAttack ground truth prevalence on $\mathcal{S}$.

The victim models fall into two regimes. For Qwen, the oracle curve remains low across budgets, indicating that only a small fraction of groups are vulnerable. Consequently, even a perfect selector cannot achieve high ASR$_B$ beyond very small B. This behavior characterizes a *scarcity-limited* regime, where low attack efficiency is fundamentally constrained by the limited availability of exploitable samples rather than by ranking errors. Under AnyAttack on D^2-City, only about 5 of 3,000 groups satisfy our attack-success criterion, so the attainable ASR$_B$ is sharply capped by prevalence. This extreme scarcity explains the near-flat triage curve for Qwen in Figure 4.

In contrast, DeepSeek exhibits an oracle curve that stays near 1 for small budgets, implying that vulnerable groups are abundant. Here, the primary limitation arises from imperfect selection: the gap between triage and oracle reflects missed vulnerable groups within the top-K ranking. This corresponds to a *selection-limited* regime, where improved vulnerability modeling could further increase attack ASR$_B$.

Prevalence Upper Bound. Define the vulnerability prevalence $p \triangleq \frac{1}{n}\sum_{i=1}^{n} y_i$. We observe that since at most pn groups are vulnerable, any policy attacking Bn groups satisfies ASR$_B \leq \min(1, p/B)$. Writing ASR$_B = (p/B) \cdot R_B$, where R_B is the recall of vulnerable groups in the top-K, separates scarcity (p/B) from selection quality (R_B). For Qwen where $p \ll B$, low p caps performance regardless of ranking; for DeepSeek where $p \gtrsim B$, high p implies that the gap to the oracle reflects imperfect selection. This also explains the higher ASR$_B$ under XTransfer in Figure 3, its stronger transferability yields greater prevalence p, raising the attainable ceiling for any selector.

4.4. Vulnerability Prediction Quality

While Section 4.2 evaluates deployment outcomes via ASR_B at fixed budgets, this metric conflates prevalence and ranking quality. We therefore evaluate the triaging model as a cross-model *ranking function* over $\mathcal{S}$.

For each victim model, we compute vulnerability scores $s_i = g(\phi(x_i))$ for all deployment groups and evaluate against the true victim outcomes y_i. We report (i) ROC-AUC, which measures pairwise separability between vulnerable and non-vulnerable samples, and (ii) PR-AUC (Average Precision), which is more informative under severe class imbalance. Importantly, ASR_B corresponds to the precision obtained when selecting the top-K groups, i.e., a single operating point on the precision–recall curve at selection rate B. Table 1 summarizes AUC metrics on the D^2-City deployment set, averaged over surrogate models. Under same-attack evaluation (AnyAttack regressors scored against AnyAttack labels), ROC-AUC is consistently above 0.70, indicating that image-level cues provide meaningful cross-model ranking signal. PR-AUC varies substantially by victim, reflecting differences in vulnerability prevalence: Qwen exhibits high ROC-AUC but extremely low PR-AUC, consistent with a scarcity-limited regime, while DeepSeek yields the highest PR-AUC, suggesting that selection quality is the primary bottleneck.

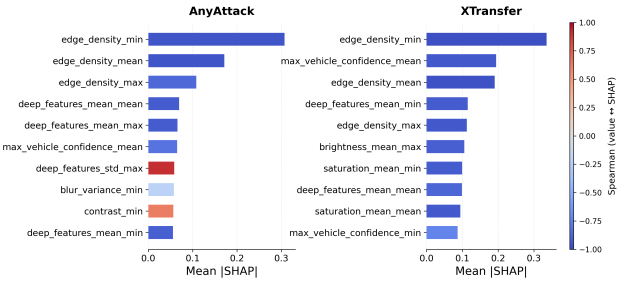


Figure 5. Top 10 SHAP features by magnitude for XGBoost models trained using labels produced by surrogate VLMs evaluated on AnyAttack and XTransfer, respectively.

To assess whether these vulnerability cues generalize across attack methods, we also evaluate regressors trained on XTransfer surrogate labels against AnyAttack victim outcomes on the *entire set* $\mathcal{S}$. ROC-AUC is largely preserved under cross-attack evaluation (mean drop 0.028), indicating that the regressor captures attack-general vulnerability signal rather than perturbation-specific artifacts. PR-AUC degrades moderately, reflecting the greater difficulty of precisely ranking the most vulnerable groups under a different perturbation. The cross-regime oracle analysis in Figure 4 corroborates this at the deployment level: XTransfer-trained regressors evaluated against AnyAttack ground truth produce triage curves that follow qualitatively similar trends, with Qwen remaining scarcity-limited and DeepSeek and Kimi staying selection-limited. Together, these results suggest that the image-level features driving vulnerability prediction are substantially shared across UAP construction methods, and that the scarcity-selection regime diagnosis is determined by the victim-attack interaction rather than the surrogate training signal.

Table 1. Same-attack vs. cross-attack triaging quality on D^2-City. *Same*: AnyAttack regressors evaluated on AnyAttack labels. *Cross*: XTransfer regressors evaluated on AnyAttack labels. Values are mean ± std across surrogate models.

ROC-AUC ↑

Victim model	Same (AA)	Cross (XT → AA)
DeepSeek	0.833 ± 0.038	0.784 ± 0.013
InternVL	0.895 ± 0.052	0.858 ± 0.015
Kimi	0.879 ± 0.037	0.844 ± 0.015
LLaVA	0.817 ± 0.060	0.831 ± 0.016
Ovis	0.928 ± 0.026	0.895 ± 0.011
Qwen	0.878 ± 0.054	0.849 ± 0.046

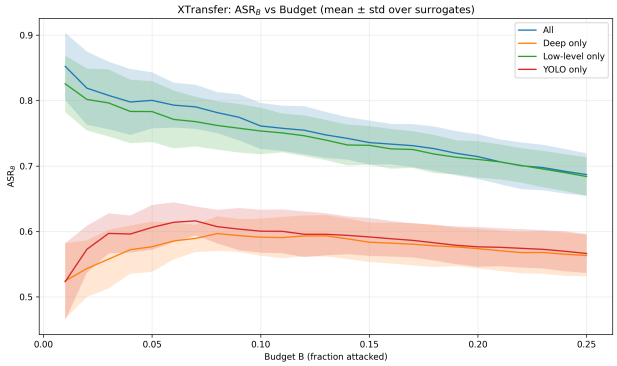
PR-AUC (AP) ↑

Victim model	Same (AA)	Cross (XT → AA)
DeepSeek	0.464 ± 0.069	0.393 ± 0.028
InternVL	0.212 ± 0.077	0.161 ± 0.027
Kimi	0.432 ± 0.084	0.342 ± 0.029
LLaVA	0.335 ± 0.070	0.295 ± 0.024
Ovis	0.424 ± 0.106	0.334 ± 0.039
Qwen	0.017 ± 0.011	0.009 ± 0.003

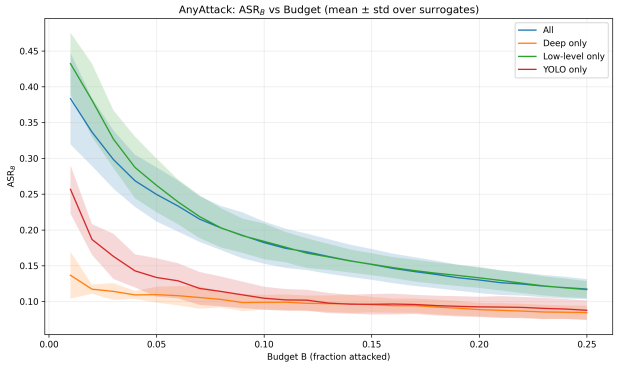
4.5. Interpreting Vulnerability Cues

SHAP-based Feature Attribution. To understand which cues drive vulnerability ranking, we compute SHAP feature attributions for the XGBoost regressors. Figure 5 shows the top features by mean absolute SHAP value for models trained under *AnyAttack* and *XTransfer*. Across both attack methods, the most influential features are *low-level image statistics*—especially edge density and brightness; blur; contrast variations, suggesting that cross-model vulnerability is strongly tied to image properties rather than model-specific semantics. We also observe that contrast features exhibit a stronger positive association under *XTransfer* than under *AnyAttack*, indicating that different UAP mechanisms may exploit different aspects of the input distribution.

Ablating Feature Families. While SHAP highlights feature usage, it does not tell us which feature *families* are necessary for budgeted gains. We therefore retrain the XGBoost triager on disjoint subsets: *low-level*, *object semantics from YOLO*, *deep features*, or *all* features and plot ASR_B versus budget in Figure 6. Across both attack regimes, low-level features nearly match the full model over all budgets, whereas YOLO-only and deep-only lag behind (deep-only



(a) XTransfer: ASR_B vs. budget under different feature subsets.



(b) AnyAttack: ASR_B vs. budget under different feature subsets.

Figure 6. **Feature ablation for vulnerability-aware triaging.** ASR_B vs. budget for XGBoost trained on all, low-level, YOLO or deep (ResNet) features; shading shows mean ± std over surrogate models. Low-level features capture most of the full-model gains in both regimes.

is worst overall). YOLO cues provide only a limited benefit at very small budgets for *AnyAttack* and do not replace low-level statistics. These results suggest that effective triaging does not require heavy perception: inexpensive low-level cues capture the dominant cross-model vulnerability signal, supporting vulnerability-aware selection as a lightweight pre-filter for budgeted deployment.

4.6. Implications for Defense

Our findings from Table 1 and the cross-regime oracle analysis in Section 4.3 jointly suggest that image triagers trained on XTransfer surrogate labels generalize to AnyAttack ground truth on $\mathcal{S}$, and that the vulnerability signal captured by image-level features is largely shared across both UAP construction methods. The cross-regime oracle curves show similar trends to the in-regime curves for AnyAttack: Qwen remains scarcity-limited, while DeepSeek and Kimi remain selection-limited regardless of whether the triager was supervised using XTransfer or AnyAttack. This provides initial evidence that a defender can guide resource allocation without knowledge of the specific attack. For instance, high prevalence p under a given attack signals broad model vulnerability where robustness improvements should be prioritized, while low p with predictable vulnerability clusters suggests that selectively screening high-risk inputs may be more cost-effective. Furthermore, since the signal is dominated by inexpensive low-level image statistics (Section 4.5), the resulting pre-filter is cheap to deploy: a VLM pipeline operator can flag high-risk inputs for additional scrutiny before acting on the model's output.

We provide a quantitative simulation of this defensive screening strategy in Section D of the supplementary material, which confirms that regressor-guided filtering substantially reduces effective attack success rates across all victim models even under cross-attack conditions. We note that both UAP methods evaluated here rely on CLIP-style surrogate encoders, and it is unclear whether the observed cross-attack generalization would hold for attacks built on fundamentally different surrogate architectures.

4.7. Limitations

Our study focuses on a single safety-critical task (road accident monitoring) and evaluates two universal attack frameworks in a digital threat model. While our results suggest that vulnerability-aware selection is broadly effective across multiple open-source VLMs, extending to additional domains (e.g., VQA or captioning), alternative decision aggregation schemes, and physical-world constraints remains future work. Our cross-attack evaluation (Table 1) provides initial evidence that vulnerability cues generalize across UAP construction methods, but the extent of this generalization across perturbation budgets and more diverse attack families warrants further study. In addition, our triaging model uses hand-crafted cues; exploring learned ranking models and streaming (online) selection policies may further improve performance in selection-limited regimes.

5. Conclusion

We propose **attack efficiency** as a complementary axis for evaluating universal adversarial attacks on VLMs under budgeted deployment. Our **image-triaging** framework consistently improves budgeted attack effectiveness across multiple VLMs, and our analyses identify two regimes that govern achievable efficiency: *scarcity-limited* settings driven by the prevalence of vulnerable inputs, and *selection-limited* settings driven by the ability to rank them accurately. We further find that the image-level vulnerability cues underlying these gains transfer across attack methods, suggesting broader relevance to defensive input filtering. Overall, our results highlight deployment strategy as a critical and previously under-measured component of universal attack risk for safety-critical VLM applications.

References

Accident-and-NonAccident. Accident and non-accident label image dataset. https://universe.roboflow.com/accident-and-nonaccident/accident-and-non-accident-label-image-dataset, 2023. CC0 1.0 License. Accessed: 2026-29-1.

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., Ring, R., Rutherford, E., Cabi, S., Han, T., Gong, Z., Samangooei, S., Monteiro, M., Menick, J., Borgeaud, S., Brock, A., Nematzadeh, A., Sharifzadeh, S., Binkowski, M., Barreira, R., Vinyals, O., Zisserman, A., and Simonyan, K. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022. URL https://arxiv.org/abs/2204.14198.

Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., and Lin, J. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. URL https://arxiv.org/abs/2502.13923.

Che, Z., Li, G., Li, T., Jiang, B., Shi, X., Zhang, X., Lu, Y., Wu, G., Liu, Y., and Ye, J. D^2-city: A large-scale dashcam video dataset of diverse traffic scenarios. *arXiv preprint arXiv:1904.01975*, 2019. doi: 10.48550/arXiv.1904.01975. URL https://arxiv.org/abs/1904.01975. Version 2, last revised 3 Jun 2019.

Chen, T. and Guestrin, C. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pp. 785–794. ACM, August 2016. doi: 10.1145/2939672.2939785. URL http://dx.doi.org/10.1145/2939672.2939785.

Fan, M., Liu, Y., Chen, C., and Liu, X. Semiadv: Query-efficient black-box adversarial attack with unlabeled images. *arXiv preprint arXiv:2407.11073*, 2024. URL https://arxiv.org/abs/2407.11073.

Fang, H., Kong, J., Yu, W., Chen, B., Li, J., Wu, H., Xia, S., and Xu, K. One perturbation is enough: On generating universal adversarial perturbations against vision-language pre-training models. *arXiv preprint arXiv:2406.05491*, 2025. URL https://arxiv.org/abs/2406.05491.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015. URL https://arxiv.org/abs/1512.03385.

Huang, H., Erfani, S., Li, Y., Ma, X., and Bailey, J. X-transfer attacks: Towards super transferable adversarial attacks on clip. *arXiv preprint arXiv:2505.05528*, 2025. URL https://arxiv.org/abs/2505.05528.

Jocher, G., Chaurasia, A., and Qiu, J. Ultralytics yolov8, 2023. URL https://github.com/ultralytics/ultralytics.

Kimi Team, Du, A., Yin, B., et al. Kimi-vl technical report. *arXiv preprint arXiv:2504.07491*, 2025. URL https://arxiv.org/abs/2504.07491.

Levy, M., Amit, G., Elovici, Y., and Mirsky, Y. Ranking the transferability of adversarial examples. *ACM Transactions on Intelligent Systems and Technology*, 15(5):1–21, October 2024. doi: 10.1145/3670409.

Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., and Li, C. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. URL https://arxiv.org/abs/2408.03326.

Li, J., Li, D., Xiong, C., and Hoi, S. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *arXiv preprint arXiv:2201.12086*, 2022. URL https://arxiv.org/abs/2201.12086.

Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. URL https://arxiv.org/abs/2304.08485.

Lu, D., Wang, Z., Wang, T., Guan, W., Gao, H., and Zheng, F. Set-level guidance attack: Boosting adversarial transferability of vision-language pre-training models. *arXiv preprint arXiv:2307.14061*, 2023. URL https://arxiv.org/abs/2307.14061.

Lu, S., Li, Y., Chen, Q.-G., Xu, Z., Luo, W., Zhang, K., and Ye, H.-J. Ovis: Structural embedding alignment for multimodal large language model. *arXiv preprint arXiv:2405.20797*, 2024. URL https://arxiv.org/abs/2405.20797.

Mokady, R., Hertz, A., and Bermano, A. H. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734*, 2021. URL https://arxiv.org/abs/2111.09734.

Qi, X., Huang, K., Panda, A., Henderson, P., Wang, M., and Mittal, P. Visual adversarial examples jailbreak aligned large language models. *arXiv preprint arXiv:2306.13213*, 2023. URL https://arxiv.org/abs/2306.13213.

Wu, Z., Chen, X., Pan, Z., Liu, X., Liu, W., Dai, D., Gao, H., Ma, Y., Wu, C., Wang, B., Xie, Z., Wu, Y., Hu, K., Wang, J., Sun, Y., Li, Y., Piao, Y., Guan, K., Liu, A., Xie, X., You, Y., Dong, K., Yu, X., Zhang, H., Zhao, L., Wang, Y., and Ruan, C. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302*, 2024. URL https://arxiv.org/abs/2412.10302.

Xu, Z., Zhang, Y., Xie, E., Zhao, Z., Guo, Y., Wong, K.-Y. K., Li, Z., and Zhao, H. Drivegpt4: Interpretable end-to-end autonomous driving via large language model. *arXiv preprint arXiv:2310.01412*, 2024. URL https://arxiv.org/abs/2310.01412.

Yin, Z., Ye, M., Zhang, T., Du, T., Zhu, J., Liu, H., Chen, J., Wang, T., and Ma, F. Vlattack: Multimodal adversarial attacks on vision-language tasks via pre-trained models. In *NeurIPS*, 2023.

Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., and Darrell, T. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. *arXiv preprint arXiv:1805.04687*, 2020. URL https://arxiv.org/abs/1805.04687.

Zhang, J., Yi, Q., and Sang, J. Towards adversarial attack on vision-language pre-training models. *arXiv preprint arXiv:2206.09391*, 2022. URL https://arxiv.org/abs/2206.09391.

Zhang, J., Ye, J., Ma, X., Li, Y., Yang, Y., Chen, Y., Sang, J., and Yeung, D.-Y. Anyattack: Towards large-scale self-supervised adversarial attacks on vision-language models. *arXiv preprint arXiv:2410.05346*, 2025. URL https://arxiv.org/abs/2410.05346.

Zhang, P.-F., Huang, Z., and Bai, G. Universal adversarial perturbations for vision-language pre-trained models. *arXiv preprint arXiv:2405.05524*, 2024a. URL https://arxiv.org/abs/2405.05524.

Zhang, Z., Zhang, A., Li, M., Zhao, H., Karypis, G., and Smola, A. Multimodal chain-of-thought reasoning in language models. *arXiv preprint arXiv:2302.00923*, 2024b. URL https://arxiv.org/abs/2302.00923.

Zhou, X., Liu, M., Yurtsever, E., Zagar, B. L., Zimmer, W., Cao, H., and Knoll, A. C. Vision language models in autonomous driving: A survey and outlook. *arXiv preprint arXiv:2310.14414*, 2024. URL https://arxiv.org/abs/2310.14414.

Zhu, J., Wang, W., Chen, Z., Liu, Z., Ye, S., Gu, L., Tian, H., Duan, Y., Su, W., Shao, J., Gao, Z., Cui, E., Wang, X., Cao, Y., Liu, Y., Wei, X., Zhang, H., Wang, H., Xu, W., Li, H., Wang, J., Deng, N., Li, S., He, Y., Jiang, T., Luo, J., Wang, Y., He, C., Shi, B., Zhang, X., Shao, W., He, J., Xiong, Y., Qu, W., Sun, P., Jiao, P., Lv, H., Wu, L., Zhang, K., Deng, H., Ge, J., Chen, K., Wang, L., Dou, M., Lu, L., Zhu, X., Lu, T., Lin, D., Qiao, Y., Dai, J., and Wang, W. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. URL https://arxiv.org/abs/2504.10479.

A. Complete Experimental Details

A.1. Evaluation Protocol

While VLMs operate at the image level, practical monitoring systems typically aggregate decisions over short temporal windows rather than individual frames. Accordingly, we evaluate at the level of temporal frame groups sampled from video data. Each group consists of 15 consecutive frames sampled at 0.2-second intervals, forming one decision unit. A group is considered successfully attacked if (i) at least 80% of its frames are correctly classified under the clean input, and (ii) at least 80% of those correctly classified frames are misclassified under the perturbation. This criterion ensures that attack success reflects sustained model failure rather than isolated frame instability.

We report group-level attack success rate (Group ASR) over the full dataset and Group ASR_B over the selected set under budget B. Group ASR_B specializes Equation 1 in the main paper to group-level evaluation:

$$\text{Group ASR}_B = \frac{1}{K} \sum_{g \in \mathcal{S}_B} y_g \tag{3}$$

where $y_g \in \{0, 1\}$ indicates whether group g is successfully attacked and $\mathcal{S}_B$ contains the top-$K = \lfloor B|\mathcal{S}| \rfloor$ groups ranked by predicted vulnerability.

We compare against uniform random selection of $K = \lfloor B|\mathcal{S}| \rfloor$ groups from the deployment stream. Because ASR_B is the mean of $\{y_g\}$ over the selected set, the random selector satisfies $\mathbb{E}[\text{ASR}_B^{\text{Rand}}] = \frac{1}{|\mathcal{S}|} \sum_{g \in \mathcal{S}} y_g =$ Group ASR. We report results for $B \in \{0.01, 0.05, 0.10\}$.

A.2. Datasets

We use subsets of D^2-City (Che et al., 2019) (3000 randomly sampled videos) and BDD100K (Yu et al., 2020) (2000 videos from `bdd100k_videos_test_00.zip`). Based on the dataset documentation, which makes no mention of accident labels or accident-containing footage, we assume that the sampled subsets contain only non-accident scenes. As such, clean accuracy measures the false-positive rate on benign inputs. We use BDD100K as the surrogate dataset $\mathcal{D}$ for both universal adversarial perturbation (UAP) generation and regressor training, and D^2-City as the deployment set $\mathcal{S}$. This cross-dataset protocol evaluates generalization under distribution mismatch (Section 4.2).

A.3. Models

We evaluate six open-source VLMs chosen for their architectural diversity and high clean accuracy on non-accident scenes, ensuring that adversarial success is not attributable to prediction bias: Ovis2-8B (Lu et al., 2024), LLaVA-OneVision-7B (Li et al., 2024), Qwen2.5-VL-7B (Bai et al., 2025), InternVL3-14B (Zhu et al., 2025), Kimi-VL-A3B-Instruct (Kimi Team et al., 2025), and DeepSeek-VL2-small (Wu et al., 2024). For brevity, we refer to these by their root names: Ovis, LLaVA, Qwen, InternVL, Kimi, and DeepSeek.

(a) XTransfer *(b)* AnyAttack

Figure 7. Visualizations of the UAPs used in our experiments.

A.4. Attack Generation

We use two state-of-the-art UAP generation approaches representing distinct paradigms: *XTransfer* (Huang et al., 2025), which performs direct optimization of the UAP via scaling an ensemble of CLIP-style surrogate models, and *AnyAttack* (Zhang et al., 2025), which performs large-scale training of a generator using a small ensemble of CLIP-style surrogate models to produce transferable perturbations.

For XTransfer, we use BDD100K (Yu et al., 2020), and optimize a UAP to maximize the feature similarity loss of the UAP and an embedding encoded target text prompt. The text prompt used was "There is an accident in this image". 32 CLIP-style models were used in the ensemble.

For AnyAttack, we finetune the pretrained generator (initialized from the base checkpoint provided publicly by the authors) using the BDD100K training split for the clean images, and a publicly available image dataset with bounding box annotations (Accident-and-NonAccident, 2023), comprising approximately 1500 images. Following the K-augmentation scheme described in the original paper, each training step pairs a cropped target accident image with K randomly sampled BDD100K images, and the generator produces a perturbation conditioned on the surrogate CLIP's target embedding. The loss maximizes cosine similarity between the perturbed clean image and the target across an ensemble of three surrogate encoders. We train for 80 epochs using cosine annealing with a restart period of 8 epochs. At inference, the generator produces a perturbation conditioned on a single accident target image depicting a car crash, which we apply universally across the deployment set at $\epsilon = 20/255$.

Visualizations of both UAPs are provided in Figure 7. Dur-

Table 2. Clean accuracy of all evaluated VLMs on BDD100K and D^2-City. Frame accuracy is the fraction of individual frames correctly classified; group accuracy is the fraction of 15-frame groups where at least 80% of frames are correct. All models achieve group-level accuracy above 0.88, with most exceeding 0.98.

Model	BDD100K		D^2-City	
	Frame Acc.	Group Acc.	Frame Acc.	Group Acc.
Ovis2-8B	0.9899	0.9875	0.9980	0.9993
LLaVA-OneVision-7B	0.9869	0.9844	0.9933	0.9927
Qwen2.5-VL-7B	0.9941	0.9932	0.9999	1.0000
InternVL3-14B	0.9759	0.9719	0.9916	0.9927
Kimi-VL-A3B	0.9226	0.8854	0.9739	0.9667
DeepSeek-VL2	0.9885	0.9849	0.9994	1.0000

ing UAP application, we resize them to 384×384, and tile the UAPs spatially across the image with a stride of 384, before applying the l_∞ clamp (at $\epsilon = 12/255$ for XTransfer and $\epsilon = 20/255$ for AnyAttack). A visualization of an unperturbed clean image with its corresponding perturbed image (with UAP applied) is provided in Figure 8.

A.5. Vulnerability Prediction

We now describe the concrete instantiation of the feature representation $\phi(\cdot)$, whose components include:

- Low-level image statistics: Brightness, contrast, saturation, hue, edge density, blur variance.
- Semantic cues: Vehicle counts by class (car, motorcycle, bus, truck), vehicle density, and vehicle confidences, total vehicles, frames with vehicles from `yolov8l` (Jocher et al., 2023).
- Deep features: First and second order statistics extracted from ResNet-50 embeddings (He et al., 2015).

The finalized feature set totals 81 features, of which 24 are selected using recursive feature elimination. We used XGBoost (Chen & Guestrin, 2016) as the lightweight regressor.

B. Additional Quantitative Results

B.1. Clean Performance

Table 2 reports clean frame and group-level accuracy for all six VLMs on BDD100K and D^2-City. All models achieve group-level accuracy above 0.88 with most exceeding 0.98, confirming that adversarial results in subsequent sections are not attributable to inherent prediction bias.

B.2. Cross-model comparison for AnyAttack

Figure 9 presents the cross-model attack success rates under budget constraints with AnyAttack. For **AnyAttack**, absolute success rates are lower due to the weaker transferability of the underlying UAP, but triaging still yields large improvements in early precision. In particular, at $B = 0.01$ we observe strong gains over random for five of the six victim VLMs, confirming that even under scarce vulnerability, prioritizing high-risk groups can markedly increase attack efficiency under strict deployment budgets. However, we note that the magnitude of improvement varies significantly across victim models, suggesting that attack efficiency is influenced both by vulnerability prevalence and by the quality of vulnerability prediction.

C. Extended Oracle Analysis

Section 4.3 in the main paper presented oracle curves for AnyAttack (Figure 4), decomposing budgeted attack efficiency into scarcity-limited and selection-limited regimes. We now extend that analysis to (i) XTransfer under in-regime evaluation, and (ii) a cross-regime setting where the triaging regressor is trained with labels obtained from one UAP method, but evaluated against another.

C.1. In-Regime Oracle Analysis (XTransfer)

Figure 10 mirrors the oracle analysis of Figure 4 in the main paper, but under the XTransfer attack. We observe that XTransfer is stronger and more transferable than AnyAttack, evident by the larger fraction of prevalent groups in the deployment set $\mathcal{S}$. Consequently, the oracle curves stay near 1 over a wider budget range, as observed for all VLMs. The oracle selector achieves near-perfect ASR_B at small budgets for all six victims, confirming that vulnerable groups are abundant under XTransfer. This contrasts with the AnyAttack setting (Figure 4), where Qwen's oracle curve remained low due to scarce vulnerability. Under XTransfer, even Qwen exhibits sufficient prevalence for the oracle to reach high ASR_B at moderate budgets. The triage curves track the oracle closely at small B and diverge gradually as the budget increases and the selector must reach deeper into the ranking, consistent with the selection-limited regime characterizing most victim models under this stronger attack.

(a) Clean *(b)* Perturbed

Figure 8. Example D^2-City frame before and after applying the XTransfer UAP ($\epsilon = 12/255$). The tiled structure arises from spatially repeating the 384×384 UAP to match the input resolution. The perturbation is largely imperceptible at the applied ϵ.

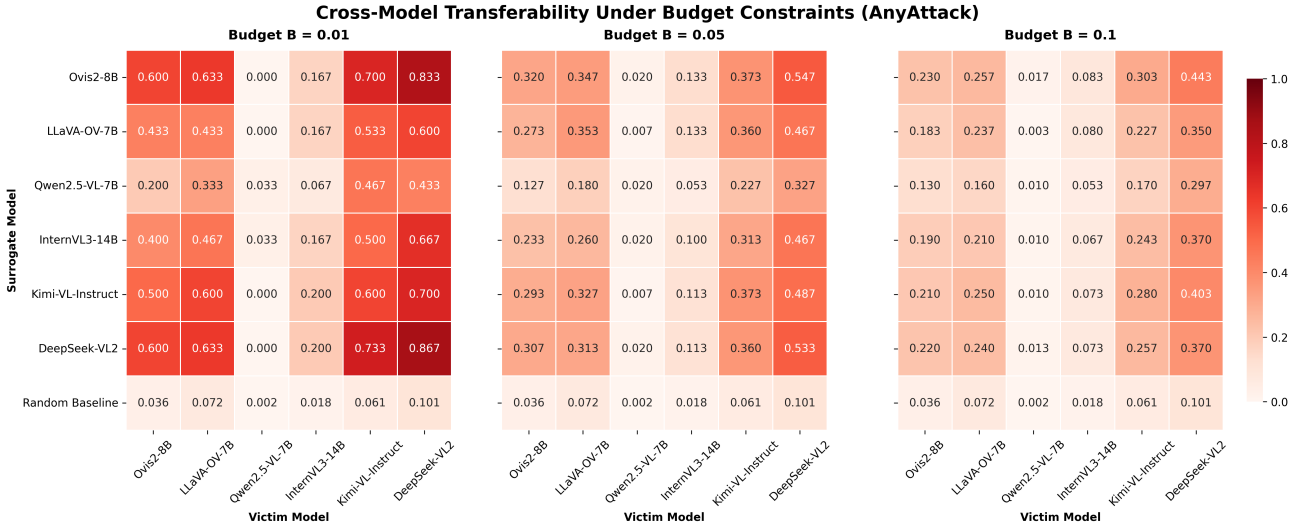


Figure 9. Cross-model attack success rates under budget constraints with AnyAttack. Rows denote regressors trained on surrogate models and columns denote victim models. Results are shown for budgets of $B = \{0.01, 0.05, 0.10\}$.

C.2. Cross-Regime Oracle Analysis

Figure 11 presents the cross-regime oracle analysis, where triaging regressors are trained on XTransfer surrogate labels but evaluated against AnyAttack ground-truth outcomes. The oracle curve in each panel reflects AnyAttack prevalence on $\mathcal{S}$, which is substantially lower than under XTransfer (Figure 10), so the prevalence ceiling $\min(1, p/B)$ binds more tightly. Despite the attack mismatch, the triage curves remain above the random baseline for most victims at small budgets, indicating that XTransfer-trained regressors capture vulnerability signal that is relevant even under a different perturbation. The regime diagnosis is also preserved: Qwen remains scarcity-limited, with both the oracle and triage curves constrained by low prevalence, while DeepSeek and Kimi remain selection-limited, with the gap between triage and oracle reflecting imperfect ranking rather than insufficient vulnerable groups. These trends mirror the in-regime AnyAttack analysis (Figure 4), supporting the observation in Section 4.4 that the scarcity-selection regime classification is determined primarily by the victim-attack interaction rather than by the surrogate training signal.

D. Defense Experiments

Section 4.6 in the main paper discussed qualitative implications of image triaging for defense. Here we provide a quantitative evaluation, treating the same vulnerability predictors used throughout the paper as a defensive screening mechanism. The core question is: can a trained vulnerability regressor identify inputs likely to be attacked, thereby reducing the effective attack success rate when a fraction of inputs can be manually reviewed or discarded before acting on model predictions?

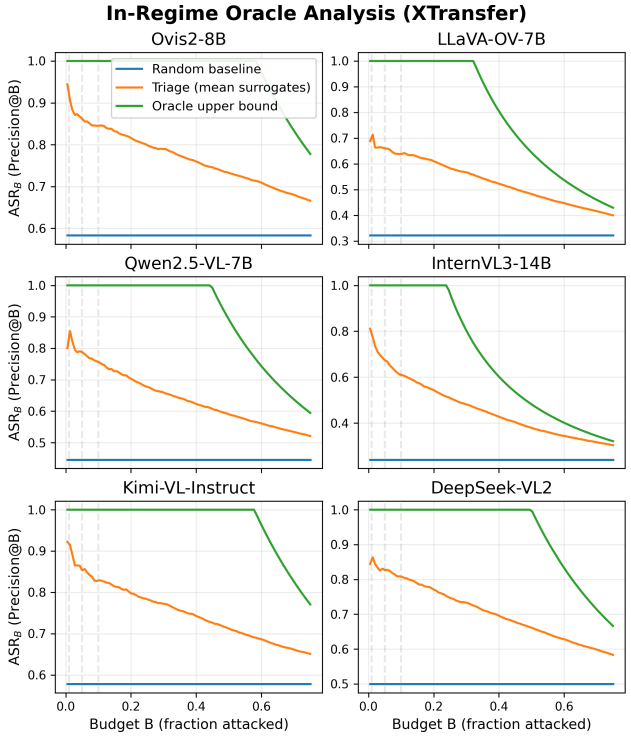


Figure 10. In-regime oracle analysis for each victim model under XTransfer. ASR_B averaged over all surrogates.

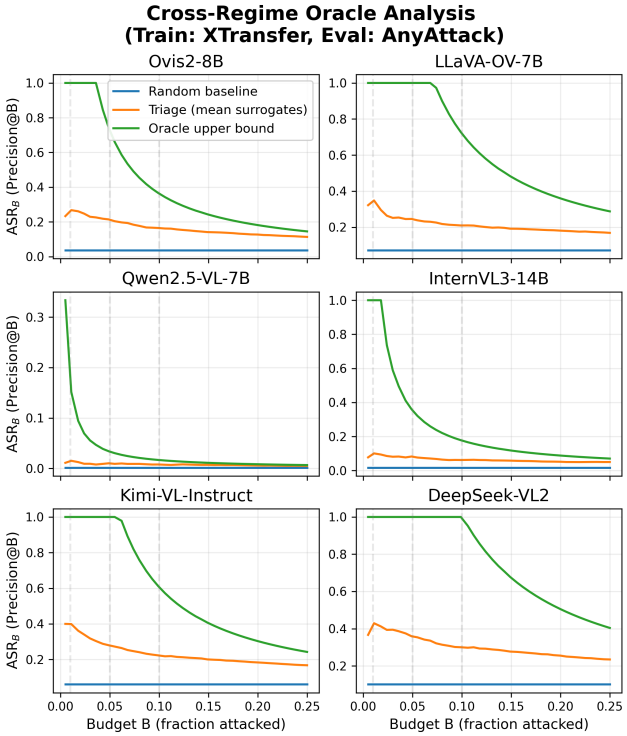


Figure 11. Cross-regime oracle analysis (Train: XTransfer, Eval: AnyAttack). ASR_B averaged over all surrogates. The oracle curve reflects AnyAttack ground-truth prevalence; the triage curve uses XTransfer-trained regressors applied to AnyAttack labels.

Setup. We simulate defense on the BDD100K test set (2000 videos), using the same 6 VLMs as victims. For a given victim VLM and surrogate regressor, we compute predicted vulnerability scores for all frame groups. At defense budget B, we "defend" the top B-fraction of groups ranked by predicted vulnerability (i.e., set their attack outcome to 0). We compare three strategies:

- **No defense**: baseline attack success rate (ASR) without any intervention.
- **Random defense**: defend a random B-fraction of groups (averaged over 10 random seeds).
- **Regressor defense**: defend the top-B groups by predicted vulnerability.

We evaluate two directions: (i) defending against XTransfer UAPs using AnyAttack-trained regressors (*forward*), and (ii) defending against AnyAttack UAPs using XTransfer-trained regressors (*reverse*). The reverse direction is the cross-attack generalization case most practically relevant: the defender trains a vulnerability predictor using labels from a known attack and deploys it against an unseen attack.

D.1. Forward Defense (AnyAttack Regressors → XTransfer)

Table 3 reports the ASR drop achieved by regressor defense at three representative budgets, averaged over all 6 surrogate regressors. The regressor consistently outperforms random defense across all victim models. At $B = 0.25$, the regressor reduces ASR by 0.22–0.25 across victims, compared to 0.11–0.16 for random defense. At $B = 0.50$, regressor defense achieves drops of 0.41–0.46, reducing the effective attack success rate to under 0.22 for all victims—representing a substantial mitigation of UAP-based attacks.

Table 3. ASR drop (absolute) from the no-defense baseline under forward defense (AnyAttack-trained regressors defending against XTransfer). Values averaged over all 6 surrogate regressors.

	Random		Regressor	
Victim	B=0.10	B=0.25	B=0.10	B=0.25
DeepSeek-VL2	0.045	0.111	0.096	0.239
InternVL3-14B	0.053	0.131	0.097	0.241
Kimi-VL-A3B	0.064	0.162	0.087	0.223
LLaVA-OV-7B	0.053	0.132	0.097	0.241
Ovis2-8B	0.062	0.157	0.099	0.248
Qwen2.5-VL-7B	0.056	0.138	0.098	0.244
Mean	0.056	0.138	0.096	0.239

Figure 12 shows the full ASR-after-defense curves per victim as a function of budget B, while Figure 13 shows the mean curve averaged over all surrogates and victims. The regressor defense (red) provides substantial gains over random

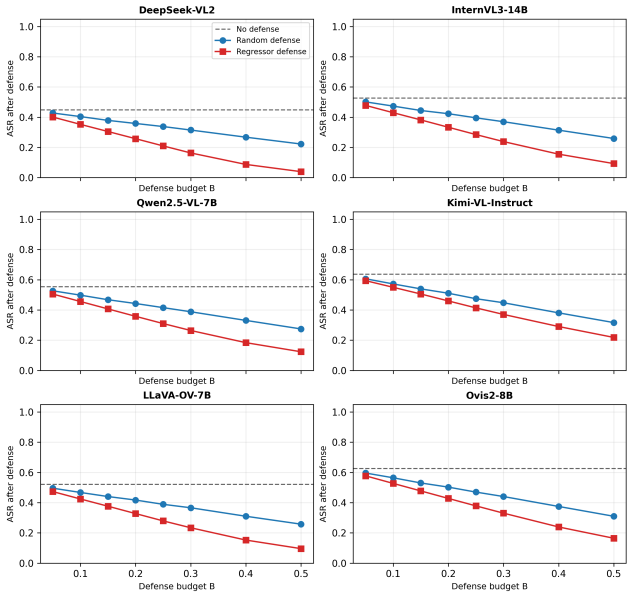


Figure 12. Defense curves per victim (forward direction). Each panel shows ASR after defense as a function of budget B for a single victim model. Regressor defense (red) consistently outperforms random defense (blue).

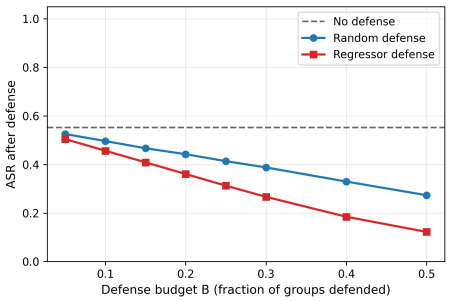


Figure 13. Mean defense curve averaged over all surrogates and victims (forward direction). The regressor defense curve lies substantially below the random baseline across all budgets.

(blue) at all budgets, with the gap widening as B increases—consistent with the observation that vulnerability predictors provide useful ranking signal beyond simple prevalence.

D.2. Reverse Defense (XTransfer Regressors → AnyAttack)

Table 4 reports the analogous results for the reverse direction. This is the more challenging cross-attack scenario: the defender trains on labels from one attack method (XTransfer) and deploys against another (AnyAttack).

Absolute ASR drops are smaller than in the forward direction, partly because AnyAttack's baseline ASR on BDD100K is lower (0.07–0.28 across victims vs. 0.45–0.64 for XTransfer), placing a tighter ceiling on achievable reductions. Nonetheless, the relative improvement over random defense remains substantial. At $B = 0.25$, the regressor achieves a mean ASR drop of 0.130 compared to 0.045 for random, representing a 2.9× improvement. Figure 14 and Figure 15 show the per-victim curves and mean curve, respectively. Despite the attack mismatch, the regressor consistently outperforms random screening at all budgets.

Table 4. ASR drop (absolute) from the no-defense baseline under reverse defense (XTransfer-trained regressors defending against AnyAttack). Values averaged over all 6 surrogate regressors.

	Random		Regressor	
Victim	B=0.10	B=0.25	B=0.10	B=0.25
DeepSeek-VL2	0.028	0.067	0.083	0.191
InternVL3-14B	0.011	0.028	0.037	0.081
Kimi-VL-A3B	0.024	0.056	0.070	0.162
LLaVA-OV-7B	0.021	0.049	0.064	0.146
Ovis2-8B	0.023	0.053	0.068	0.155
Qwen2.5-VL-7B	0.007	0.016	0.020	0.047
Mean	0.019	0.045	0.057	0.130

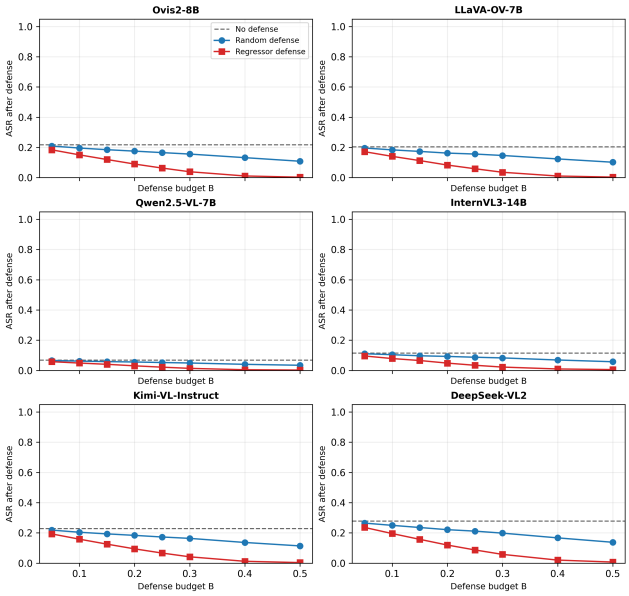


Figure 14. Defense curves per victim (reverse direction: XTransfer regressors → AnyAttack). Regressor defense (red) maintains a margin over random (blue) in the cross-attack setting.

Summary. These defense experiments provide quantitative evidence that the vulnerability ranking signal generalizes to a cross-attack setting, supporting the qualitative discussion in Section 4.6 of the main paper. A defender can train a vulnerability regressor using labels from a known attack and deploy it to selectively screen inputs against an unseen attack, achieving meaningful reductions in effective attack success rate at modest screening budgets. The regressor-guided approach consistently outperforms random

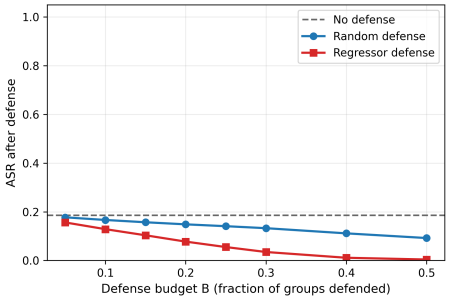


Figure 15. Mean defense curve for the reverse direction, averaged over all surrogates and victims.

screening by 1.5–3.5× at $B = 0.25$, confirming that the rankings learned by the triaging framework are actionable for defense, not only for attack efficiency.