SASA: Sequence-Aware Shadow Attacks via Attention Alignment for Traffic Sign Recognition

Amir Salarpour*, Pedram MohajerAnsari*, David Fernandez, Mert D. Pesé
School of Computing, Clemson University
{asalarp, pmohaje, dferna3, mpese}@clemson.edu

Abstract

*We propose **SASA** (Sequence-Aware Shadow Attack), a black-box adversarial framework that uses physically realistic, differentiable shadow patterns to deceive traffic sign recognition systems. Unlike prior image-based attacks, SASA targets video sequences by generating temporally consistent shadows that are visually indistinguishable from natural lighting conditions—a property that makes them difficult to filter or detect. Attention maps from frozen vision transformers guide shadow placement toward semantically salient regions, requiring no access to the target model at any stage. Evaluated on GTSRB, SASA drops classification accuracy by up to 86 percentage points and sequence-level accuracy by over 90 percentage points across black-box models, including CNNs and ViTs. Attacks transfer across architectures without retuning, and generated shadows pass perceptual quality checks, exposing a gap in how sequential vision systems handle physically grounded, temporally stable perturbations.*

1. Introduction

Adversarial examples have exposed critical vulnerabilities in deep neural networks, raising serious safety concerns in autonomous driving. Most adversarial research targets static images with imperceptible perturbations [12, 24], yet real-world perception systems process video streams, not individual frames. Frame-level perturbations tend to fail in this setting because temporal filtering, camera motion, and perceptual inconsistencies suppress them before they reach the classifier [3, 7, 10].

Physical-world attacks on traffic sign recognition (TSR) have taken a different route: printed posters [8], adversarial stickers [22], light projection [28], and shadow casting [29] all show that naturally plausible perturbations can fool models without drawing human attention. Cast shadows are a particularly practical threat. They appear in ordinary driving scenes, persist across frames without any per-frame modification, and require no changes to the sign itself. Figure 1 compares representative light-based attacks alongside the three shadow variants produced by our method.

Existing shadow attacks either operate on single frames [29] or need white-box access to the target model [18]. Neither condition holds in practice: real TSR pipelines process video, and model internals are rarely accessible.

We present **SASA** (Sequence-Aware Shadow Attack), a black-box video-based attack that places physically plausible shadows over TSR video sequences without querying the target model. A single shared shadow mask is optimized using attention maps from frozen DINO and DeiT transformers, then applied uniformly across all frames. The attack has three parts: a differentiable shadow generator with geometric control over shape, position, and opacity; an attention-guided alignment loss that steers shadows onto class-discriminative regions; and an optimization loop that enforces temporal consistency and perceptual realism. Tested under strict black-box conditions on CNN, STN, EffB0, and ViT classifiers, SASA drops sequence-level accuracy by up to **90 percentage points** on ViT and frame-level accuracy by over **30 percentage points** on average. DeiT-guided StripShadow is the strongest variant across all targets.

Contributions.

- SASA is the first fully black-box, temporally coherent shadow attack designed for sequential TSR systems.
- We build a differentiable shadow generator that supports three physically grounded mask types (Blob, Strip, Side) with compact geometric parameterization.
- Shadow placement is guided by attention maps from frozen DINO and DeiT transformers, with no target model access at any stage.
- SASA transfers across CNN and transformer architectures, cutting TSR accuracy by up to **86 percentage points** on ViT and over **60 percentage points** on EffB0.

2. Related Work

Adversarial Attacks on Vision Models. Deep neural networks are vulnerable to adversarial attacks in both digi-

*The first two authors contributed equally.

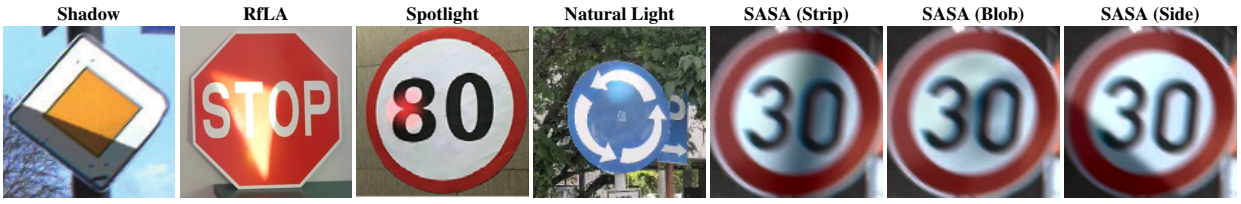

Figure 1. Light-based physical attacks on traffic signs: shadow casting [29], reflected laser [26], spotlight [28], and natural illumination [13]. The last three columns show SASA's Strip, Blob, and Side shadow variants on the same sign.

tal and physical settings. White-box [12, 17] and black-box [14, 19] methods generate pixel-level perturbations that often break under real-world transformations [2]. Recent surveys [20] point out that evaluations built around LISA-CNN or GTSRB-CNN baselines underestimate the difficulty of realistic, sequential TSR scenarios [9].

Physical-World and Light-Based Attacks. Physical attacks span printed posters [21], adversarial stickers [8, 11], and camouflage patches [4]. Light-based methods push this further: RFLA [26] uses reflected sunlight and AdvSL [28] uses spotlights to inject perturbations without touching the sign. Both give the attacker control over timing and placement. Hsiao et al. [13] show that even uncontrolled natural illumination shifts can break TSR classifiers.

Video and Temporally Coherent Attacks. Attacks on static images do not carry over directly to video pipelines. Frame-level perturbations flicker visually and are often suppressed by temporal filtering or tracking [3, 7]. An effective video attack must hold a stable perturbation across frames. Cast shadows satisfy this naturally: the same occlusion geometry persists as the camera moves, with no per-frame recomputation required.

Shadow-Based and Attention-Guided Attacks. Zhong et al. [29] showed triangular shadow masks reach over 90% attack success on single-image TSR. MohajerAnsari et al. [18] extended this to video, using a fixed-shape, temporally scaled shadow guided by DINO attention maps, and introduced the Sequence-Level Attack Success Rate (SL-ASR) as a video-appropriate metric. Attention maps from vision transformers have also been used to steer perturbations toward class-discriminative regions in other settings [16, 27].

SASA departs from prior shadow attacks in three ways: it supports three differentiable mask geometries (Blob, Strip, Side); it draws on both DINO and DeiT attention to cover complementary saliency signals; and it tests transferability across four architectures spanning CNNs and transformers, all without any target model queries.

3. Methodology

We propose **SASA** (Sequence-Aware Shadow Attack), a physically grounded black-box adversarial framework that constructs temporally coherent shadow patterns across video sequences of traffic signs. SASA comprises three components: (1) a differentiable shadow generator, (2) attention-guided placement informed by pretrained vision transformers, and (3) an optimization strategy that aligns shadows with semantically salient regions, with no access to the target classifier.

3.1. Problem Formulation

Standard adversarial attacks in vision target single-frame inputs with unconstrained pixel-level perturbations. Real-world systems like TSR, however, process temporally structured video streams and rely on frame-to-frame consistency to reject transient noise. Frame-wise perturbations typically produce visual flickering or temporal artifacts that are suppressed by post-processing or tracking modules before reaching the classifier. SASA instead applies a single, temporally consistent shadow transformation across all frames, mimicking natural lighting effects such as occlusion from poles or foliage. These shadows maintain spatial coherence across the sequence while avoiding perceptual artifacts.

Formally, let $\{x_1, \ldots, x_T\}$, where $x_t \in [0,1]^{3\times H\times W}$, denote a video sequence, and let $f : \mathbb{R}^{3\times H\times W} \to \mathcal{Y}$ be a TSR classifier. We seek a shared shadow function $\mathcal{S}(x_t; \theta, \gamma)$ such that each perturbed frame $\tilde{x}_t = \mathcal{S}(x_t)$ is misclassified: $f(\tilde{x}_t) \neq f(x_t)$ for all t. Here, θ defines the geometric parameters of the shadow, and $\gamma \in [0,1]$ controls its intensity. This formulation permits efficient, differentiable optimization of realistic shadows across time.

3.2. Threat Model

We assume a strict black-box threat model: the attacker has no access to model weights, architecture, gradients, or predictions. Unlike white-box methods that rely on back-propagation through the target model, SASA uses frozen attention maps extracted from DINO and DeiT transformers fine-tuned on the TSR dataset. These models are used solely to guide shadow placement during optimization. The final adversarial shadows are then evaluated against entirely unseen target models, ensuring a realistic and rigorous black-box evaluation.

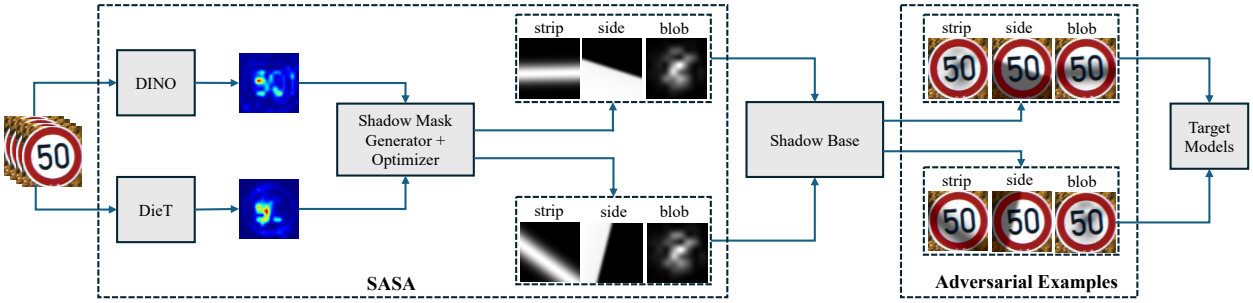

Figure 2. Overview of the SASA pipeline. A clean video sequence is processed by a frozen attention model (DINO or DeiT) to produce a fused saliency map. This map guides the optimization of a shared, differentiable shadow mask. The mask is applied uniformly across all frames, and the resulting perturbed sequence is evaluated against unseen TSR classifiers (e.g., CNNs, ViTs) in a strict black-box setting.

3.3. The SASA Framework

SASA produces a single shadow mask that is spatially aligned and temporally consistent across the entire video sequence. The mask is optimized via a compact set of differentiable parameters, guided by pretrained attention maps. The pipeline comprises four stages:

1. **Attention Aggregation.** A frozen DeiT or DINO transformer extracts spatial attention maps from each frame x_t. These are fused via temporal max-pooling into a single 2D heatmap.
2. **Shadow Generation.** A shadow generator (*BlobShadow*, *StripShadow*, or *SideShadow*) generates $M \in [0,1]^{H\times W}$ via a differentiable function $\text{Gen}(\theta)$, parameterized by compact geometric variable θ. The same mask is applied to every frame using the transformation $\mathcal{S}(x_t;\theta,\gamma)$.
3. **Attention-Aligned Optimization.** The mask M is optimized to match the fused attention map A using a top-k alignment loss $\mathcal{L}_{\text{align}}$. This optimization requires no target model access.
4. **Black-box Evaluation.** The resulting adversarial shadow is tested on multiple held-out TSR classifiers (e.g., CNNs, ViTs) across a range of intensities γ, with no further tuning.

Decoupling attack generation from model-specific gradients produces transferable, physically grounded, and temporally smooth adversarial sequences. An overview of the complete pipeline is shown in Figure 2.

3.4. Differentiable Shadow Generation

SASA uses a physically motivated shadow generator that simulates real-world occlusions (e.g., from trees, poles, overpasses). Rather than perturbing individual pixels, SASA optimizes interpretable, low-dimensional parameters that define soft shadow masks $M \in [0,1]^{H\times W}$.

Masks are applied uniformly across the sequence to preserve temporal consistency. Each shadow is rendered in CIELAB space, darkening only the luminance channel:

$$L' = L \cdot (1 - \gamma M), \quad a' = a, \quad b' = b.$$

This preserves color consistency while darkening luminance proportionally to the mask. The image is then converted back to RGB and clamped to a valid range. To produce physically realistic soft shadow edges, a Gaussian falloff is applied around shadow boundaries:

$$F(d) = \exp\left(-\left(\frac{d}{r}\right)^2 \cdot \tau\right),$$

where d is distance from the mask center, r is radius, and τ controls edge softness. SASA supports three shadow styles, described next.

BlobShadow: Radial Shadows. *BlobShadow* generates soft, amorphous occlusions resembling natural phenomena such as tree cover or cloud shadows. It is parameterized by $\theta = (c_x, c_y, s)$, where $(c_x, c_y) \in [0,1]^2$ defines the mask center and $s \in [0.1, 0.8]$ controls spatial extent. The result is a diffuse, circular shadow with smooth falloff, suited for simulating irregular organic occlusions (see Figure 3, left).

StripShadow: Elongated Directional Shadows. *StripShadow* emulates shadows cast by upright objects such as poles, signposts, or fences. It is parameterized by $\theta = (c_x, c_y, \alpha, s)$, where $\alpha \in [0, \pi]$ controls orientation and s specifies the band width. The mask supports sharp directional control, making it effective for occluding vertically aligned sign features (see Figure 3, right).

SideShadow: Lateral Projection Shadows. *SideShadow* produces oblique or asymmetric shadows cast from lateral occluders such as roadside barriers or parked vehicles. It is defined by $\theta = (c_x, c_y, \alpha, f)$, where $f \in [0,1]$ is a continuous flip factor interpolating between left- and right-projected masks. Two directional masks are blended using sigmoid-weighted interpolation, followed by Gaussian smoothing, producing nuanced asymmetric shading that uniform shadow models cannot capture (see Figure 3, middle). Pseudocode

Algorithm 1 Shadow Mask Generation: BLOBSHADOW and SIDESHADOW

Require: Image size (H, W), type $t \in \{\text{blob}, \text{side}\}$, parameters θ, blur kernel size
if $t = \text{blob}$ **then**
Require: $\theta = (c_x, c_y, s)$, seed resolution k
  Sample noise tensor $N \in \mathbb{R}^{1\times1\times k\times k}$
  Upsample $N \to \mathbb{R}^{1\times1\times H\times W}$ via bilinear interpolation
  Apply Gaussian blur; normalize to $[0, 1]$
  Generate radial falloff at (c_x, c_y) with radius $r = s \cdot H$
  Multiply blob by radial falloff and normalize
else if $t = \text{side}$ **then**
Require: $\theta = (c_x, c_y, \alpha, f)$, blend sharpness κ
  Create coordinate grid centered at (c_x, c_y); rotate by α to get y_{rot}
  Define half-plane masks: $M_1 = \mathbb{1}[y_{\text{rot}} > 0]$, $M_2 = \mathbb{1}[y_{\text{rot}} < 0]$
  $M \leftarrow \sigma((f-0.5)\kappa) \cdot M_2 + (1 - \sigma((f-0.5)\kappa)) \cdot M_1$
  Apply Gaussian blur and distance-based falloff
end if
return Shadow mask $M \in [0, 1]^{H\times W}$

for BlobShadow and SideShadow is provided in Algorithm 1; StripShadow follows an analogous rotated-grid procedure.

3.5. Attention-Guided Shadow Alignment

To place shadows where they disrupt classification most, SASA uses attention maps from two pretrained transformers with complementary inductive biases. DINO [5], trained via self-supervised contrastive learning, produces object-centric saliency maps. DeiT [25], trained with supervision, highlights class-discriminative regions. Both models are fine-tuned on the TSR dataset and remain frozen during optimization.

Attention Rollout and Temporal Fusion. We use attention rollout [1] to extract dense saliency maps from transformer layers. At each layer l, residual-aware attention is computed as:

$$\bar{A}^{(l)} = \alpha A^{(l)} + (1 - \alpha)I,$$

where $\alpha = 0.9$ balances the raw attention matrix $A^{(l)} \in \mathbb{R}^{N\times N}$ with identity flow I. Cumulative attention is obtained by multiplying across layers:

$$\tilde{A} = \bar{A}^{(1)}\bar{A}^{(2)} \cdots \bar{A}^{(L)}.$$

We extract the class-to-token row, discard the CLS token, and reshape the remaining values into a 2D map $A_t \in \mathbb{R}^{h\times w}$,

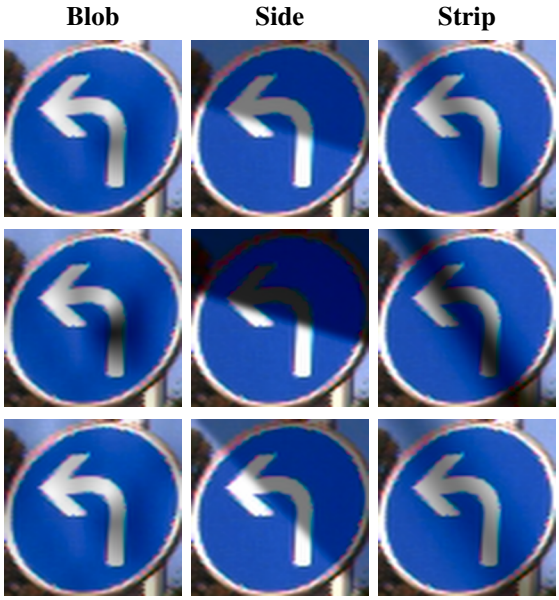

Figure 3. Three shadow styles at varying parameters. BlobShadow (left): diffuse radial occlusion. SideShadow (middle): lateral asymmetric projection. StripShadow (right): elongated directional band.

upsampled to $H \times W$. Temporal max-pooling consolidates maps across frames:

$$A = \max_{t\in\{1,\ldots,T\}} A_t.$$

Optionally, a 2D Gaussian spatial prior biases attention toward regions where traffic signs commonly appear:

$$G(x, y) = \exp\left(-\lambda\left(\frac{(x - c_x)^2}{W^2} + \frac{(y - c_y)^2}{H^2}\right)\right),$$
$$A(x, y) \leftarrow A(x, y) \cdot G(x, y).$$

Top-k Alignment Loss. To focus optimization on the most influential pixels, let $\mathcal{T}_k$ index the top-k values in A. The shadow mask M is trained to cover these positions:

$$\mathcal{L}_{\text{align}} = \frac{1}{k}\sum_{i\in\mathcal{T}_k} |M_i - A_i|^p,$$

where $p = 1$ gives L1 loss and $p = 2$ gives MSE. Restricting to top-k pixels keeps the shadow compact while concentrating occlusion on the regions that most influence the classifier's decision.

3.6. Optimization Objective

The full loss combines semantic alignment with optional regularization:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{align}} + \lambda_{\text{reg}} \cdot \mathcal{L}_{\text{reg}},$$

Table 1. Classification accuracy (%) and sequence accuracy at 50% threshold (`seq@50`, %) for clean sequences and relative drop (↓) under each shadow variant. For each intensity level ($\gamma = 0.2, 0.5, 0.8$), the highest drop per model is **bolded**.

Intensity	Variant	CNN		STN		EffB0		ViT	
		Acc ↓	seq@50 ↓	Acc ↓	seq@50 ↓	Acc ↓	seq@50 ↓	Acc ↓	seq@50 ↓
–	Clean (absolute acc.)	96.6	100.0	95.8	100.0	99.9	100.0	95.1	100.0
0.2	BlobShadow (DeiT)	0.6	0.0	2.4	2.5	1.1	0.0	4.6	**5.0**
	BlobShadow (DINO)	0.6	0.0	**2.7**	2.5	1.0	0.0	3.8	2.5
	SideShadow (DeiT)	**2.2**	0.0	2.4	2.5	0.5	0.0	**5.6**	2.5
	SideShadow (DINO)	0.9	0.0	1.9	2.5	0.3	0.0	2.7	2.5
	StripShadow (DeiT)	1.3	0.0	2.6	2.5	**2.5**	0.0	4.0	2.5
	StripShadow (DINO)	0.5	0.0	2.6	2.5	0.7	0.0	1.4	0.0
0.5	BlobShadow (DeiT)	9.0	7.5	12.4	10.0	8.4	5.0	20.3	12.5
	BlobShadow (DINO)	8.6	5.0	12.9	7.5	6.5	2.5	21.0	17.5
	SideShadow (DeiT)	11.8	**7.5**	14.3	15.0	9.1	7.5	**32.5**	35.0
	SideShadow (DINO)	8.8	2.5	10.5	7.5	8.2	2.5	24.2	20.0
	StripShadow (DeiT)	**12.4**	5.0	**20.2**	**20.0**	**30.8**	**27.5**	31.4	**37.5**
	StripShadow (DINO)	7.0	2.5	15.8	15.0	17.8	15.0	27.6	27.5
0.8	BlobShadow (DeiT)	22.8	22.5	30.1	25.0	21.7	12.5	75.3	80.0
	BlobShadow (DINO)	21.6	22.5	31.7	25.0	22.1	15.0	75.8	90.0
	SideShadow (DeiT)	35.1	35.0	44.2	45.0	33.3	32.5	80.6	87.5
	SideShadow (DINO)	35.3	40.0	43.2	45.0	35.5	27.5	75.8	80.0
	StripShadow (DeiT)	**40.6**	**47.5**	**46.9**	**47.5**	**62.0**	**62.5**	**86.3**	**92.5**
	StripShadow (DINO)	27.3	25.0	44.7	45.0	40.1	35.0	78.8	85.0

where $\mathcal{L}_{\text{reg}}$ may include total variation, sparsity, or spatial smoothness penalties to keep shadows physically plausible. No gradients or outputs from the target model enter this computation at any point.

4. Experiments

We evaluate **SASA** under a strict black-box threat model on the TSR task. The experiments address four questions: (1) how much does each shadow variant degrade accuracy across model architectures, (2) do the generated shadows remain temporally consistent and perceptually realistic, (3) do the attacks transfer to unseen models without adaptation, and (4) how do loss type and attention coverage affect performance. The analysis spans three shadow types, ten intensity levels, and two attention backbones (DeiT and DINO).

4.1. Experimental Setup

Dataset and Preprocessing. We use the **German Traffic Sign Recognition Benchmark (GTSRB)** [23], consisting of over 50,000 traffic sign images across 43 classes. To simulate sequential video input, images are grouped by class and sample ID into sequences standardized to $T = 30$ frames via a sliding window. Images are cropped by bounding box and resized to 128×128 for CNN-based models and 224×224 for transformer-based models. We discard sequences with low average luminance ($L < 120$ in CIELAB space) to ensure generated shadows remain perceptually distinguishable from the background. All train/test splits are class-stratified and fixed across experiments.

Target Models. We evaluate SASA against four TSR classifiers across convolutional and transformer architectures.
CNN-based models. **GTSRB-CNN** [6] and **GTSRB-STN** [15] are reimplemented and trained from scratch on the GTSRB training set at 128×128 resolution.
Transformer-based models. ImageNet-pretrained **EfficientNet-B0** and **ViT-Small** are fine-tuned on GTSRB via end-to-end supervised training at 224×224.
Attention models. **DeiT** [25] and **DINO** [5] serve exclusively as frozen saliency extractors, fine-tuned on GTSRB to improve attention map fidelity. They are never used as attack targets, strictly preserving the black-box threat model.

4.2. Evaluation Metrics

Frame-wise Accuracy measures average classification accuracy across all frames in a sequence, giving a per-frame

Table 2. Classification accuracy (%) and `seq@50` (%) averaged over all shadow intensities ($\gamma \in [0.1, 1.0]$). Gray-box results (DeiT and DINO as both guide and target) are in the last two columns. The strongest attack per model is **bolded**; the overall best variant is highlighted.

Variant	CNN		STN		EffB0		ViT		DeiT		DINO	
	Acc	seq@50	Acc	seq@50	Acc	seq@50	Acc	seq@50	Acc	seq@50	Acc	seq@50
Clean (baseline)	96.6	100.0	95.8	100.0	99.9	100.0	95.1	100.0	98.8	100.0	92.8	100.0
BlobShadow (DeiT)	83.5	86.8	77.2	82.8	87.1	91.3	55.3	58.8	59.4	61.5	68.6	74.0
BlobShadow (DINO)	83.9	87.8	76.3	83.3	87.5	91.3	55.5	57.5	59.9	62.3	69.6	74.5
SideShadow (DeiT)	74.4	80.5	68.1	71.5	77.7	80.3	49.8	53.5	46.4	48.0	62.6	67.5
SideShadow (DINO)	76.2	79.8	70.7	74.3	79.2	81.5	54.9	58.5	51.3	52.8	65.7	70.3
StripShadow (DeiT)	**73.5**	**78.5**	**66.3**	**69.8**	**64.7**	**65.3**	**49.4**	**52.0**	**43.3**	**44.8**	**54.8**	**61.0**
StripShadow (DINO)	81.1	86.5	69.2	74.3	76.6	79.0	53.4	55.8	49.2	50.8	61.4	64.8

view of how often the model is fooled under attack.

Sequence-level Accuracy (`seq@50`) marks a sequence as correctly classified only if at least 50% of its frames yield the correct label. This is the more task-relevant metric for video-based TSR: a classifier that fails on the majority of frames in a sequence has effectively lost the sign, regardless of a few correct frames.

4.3. Evaluation Results

Table 1 reports accuracy drops at three shadow intensities ($\gamma \in \{0.2, 0.5, 0.8\}$) relative to the clean baseline across four black-box TSR models.

Note on Comparability. We do not compare directly against prior shadow attacks [29] because they run per-image optimization on individual frames. SASA optimizes one mask shared across a full video sequence, which is a more constrained problem with a different evaluation protocol.

Low intensity ($\gamma = 0.2$). Shadows are subtle and no single variant dominates. SideShadow (DeiT) yields the strongest effect on CNN and ViT, while other variants perform at a similar level. At low visibility, effectiveness depends more on attention alignment than shadow geometry.

Medium intensity ($\gamma = 0.5$). StripShadow (DeiT) outperforms other variants on all models except ViT, where it is marginally stronger. The elongated strip pattern aligns well with high-attention regions on structured traffic signs.

High intensity ($\gamma = 0.8$). StripShadow (DeiT) is the strongest variant across all models, reaching drops of 62 percentage points on EffB0 and 86 percentage points on ViT in frame-level accuracy, with seq@50 degradation of 92.5 percentage points on ViT. SideShadow and BlobShadow remain competitive but do not match StripShadow at this intensity. DeiT-guided variants outperform their DINO-guided counterparts in all settings, which is consistent with class-discriminative attention being more informative for shadow placement than object-centric attention.

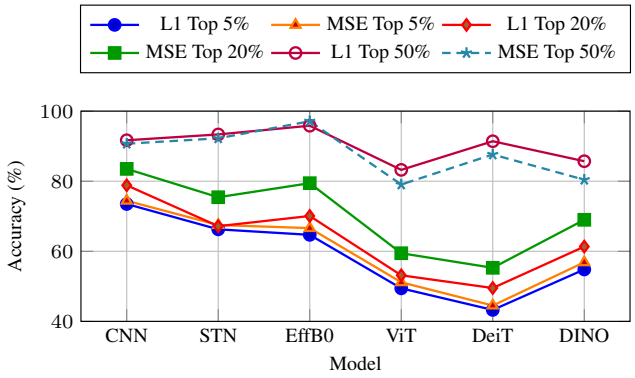

Figure 4. Classification accuracy across six target models under different loss functions and top-k attention coverage. Lower is better. All experiments use StripShadow with DeiT attention, averaged over $\gamma \in [0.1, 1.0]$.

4.4. Average Performance Across Intensities

Table 2 reports average accuracy and `seq@50` across the full intensity range ($\gamma \in [0.1, 1.0]$), for both black-box settings and gray-box cases where the same transformer is used to guide the attack and as the classification target.

StripShadow (DeiT) produces the lowest average accuracy across all six models. The gap is largest on ViT and DeiT, where attention-aligned perturbations most directly disrupt the model's localization mechanism. ViT is the most vulnerable target overall, which follows from its heavy reliance on patch-level attention for classification. GTSRB-CNN shows the most resilience, consistent with convolutional architectures being less sensitive to global saliency disruption.

4.5. Loss Function Ablation

We compare L1 and MSE alignment losses over the top-k% most salient attention values, with $k \in \{5, 20, 50\}$, using StripShadow with DeiT attention averaged over $\gamma \in$

$[0.1, 1.0]$.

Broader coverage yields stronger attacks. Top-50% variants achieve the lowest accuracy across all models. Widening the attention coverage from 5% to 50% forces the shadow to occlude a larger share of discriminative pixels, which translates directly into larger accuracy drops.

MSE slightly outperforms L1 at low coverage. At top-5%, MSE produces marginally better results, likely because its squared penalty provides smoother gradients during optimization. The gap closes at higher k values where both losses have enough signal to work with.

Transformer models are more vulnerable. ViT and DeiT show the lowest final accuracies across all configurations, which is consistent with their reliance on global attention. CNN-based models retain higher accuracy throughout, as their local receptive fields are less affected by region-level occlusion.

Taken together, these results show that the top-k alignment loss is a practical design choice: even L1 at top-50% produces strong black-box attacks without requiring target model access.

4.6. Qualitative Attention Alignment

Figure 5 shows attention maps before and after the shadow attack for each of the three shadow types. Each row contains: (1) the original frame, (2) the DeiT attention map on the clean input, (3) the optimized shadow mask, and (4) the attention map on the perturbed frame. In all three cases the shadow lands on the highest-attention regions, and the post-attack attention map is visibly weaker or displaced. This confirms that the accuracy drops in Tables 1 and 2 are driven by disruption of the model's internal localization, not just pixel-level degradation.

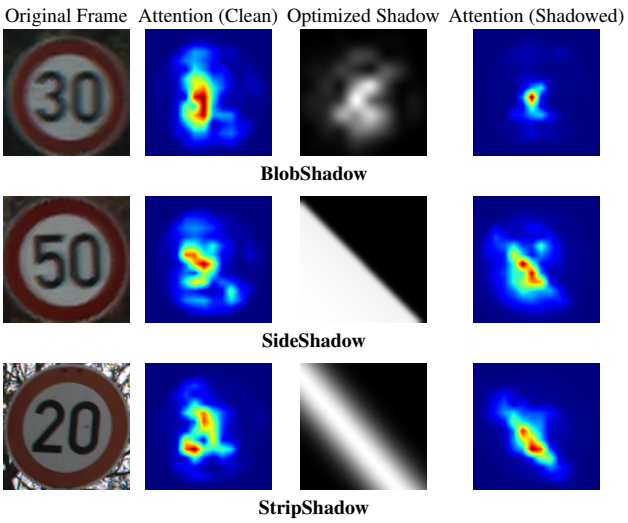

Figure 5. Attention maps before and after the shadow attack for each shadow type. The shadow lands on the highest-saliency regions (col. 2), and the post-attack map (col. 4) shows weakened or displaced attention.

5. Conclusion

We presented **SASA** (Sequence-Aware Shadow Attack), a black-box attack that uses differentiable shadow generators and frozen transformer attention maps (DINO and DeiT) to place physically plausible, temporally coherent perturbations on TSR video sequences without querying the target model.

On GTSRB, SASA drops frame-level accuracy by up to 86 percentage points on ViT and over 60 percentage points on EfficientNet, and degrades sequence-level accuracy by over 90 percentage points at high intensity. StripShadow guided by DeiT is the strongest variant across all four architectures, which is consistent with class-discriminative attention being more useful for shadow placement than object-centric attention. The ablation shows that top-50% coverage with MSE loss produces the largest accuracy drops, though even L1 at top-50% is competitive.

Sequential TSR systems are substantially more sensitive to temporally stable, attention-aligned perturbations than per-frame accuracy numbers suggest. Current evaluations that report only frame-level metrics understate the practical risk. This work is limited to GTSRB under simulated video sequences; validating SASA with physical shadow casting in real driving conditions, and building defenses against attention-aware video attacks, are the natural next steps.

Acknowledgments

We gratefully acknowledge the support provided by the U.S. Department of Transportation (DOT) through the National Center for Transportation Cybersecurity and Resiliency (TraCR) under Grant No. 69A3552344812-2027534 and 69A3552348317.

References

[1] Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. *arXiv preprint arXiv:2005.00928*, 2020. 4

[2] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pages 274–283. PMLR, 2018. 2

[3] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *International conference on machine learning*, pages 284–293. PMLR, 2018. 1, 2

[4] Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017. 2

[5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Pro-*

ceedings of the IEEE/CVF international conference on computer vision, pages 9650–9660, 2021. 4, 5

[6] Dan Ciregan, Ueli Meier, and Jürgen Schmidhuber. Multi-column deep neural networks for image classification. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3642–3649. IEEE, 2012. 5

[7] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4312–4321, 2019. 1, 2

[8] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, et al. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1625–1634, 2018. 1, 2

[9] David Fernandez, Pedram MohajerAnsari, Cigdem Kokenoz, Amir Salarpour, Bing Li, and Mert D Pesé. Wip: From detection to explanation: Using llms for adversarial scenario analysis in vehicles. In *Proceedings of the 3rd USENIX Symposium on Vehicle Security and Privacy (VehicleSec '25)*. USENIX Association, 2025. 2

[10] David Fernandez, Pedram MohajerAnsari, Amir Salarpour, and Mert D Pesé. Avoiding the crash: A vision-language model evaluation of critical traffic scenarios. Technical report, SAE Technical Paper, 2025. 1

[11] David Fernandez, Pedram MohajerAnsari, Amir Salarpour, Long Cheng, Abolfazl Razi, and Mert D. Pesé. Comparative analysis of patch attack on vlm-based autonomous driving architectures, 2026. 2

[12] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*, 2015. 1, 2

[13] Teng-Fang Hsiao, Bo-Lun Huang, Zi-Xiang Ni, Yan-Ting Lin, Hong-Han Shuai, Yung-Hui Li, and Wen-Huang Cheng. Natural light can also be dangerous: Traffic sign misinterpretation under adversarial natural light attacks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3915–3924, 2024. 2

[14] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International conference on machine learning*, pages 2137–2146. PMLR, 2018. 2

[15] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *Advances in neural information processing systems*, 28, 2015. 5

[16] Hsueh-Ti Derek Liu, Michael Tao, Chun-Liang Li, et al. Beyond pixel norm-balls: Parametric adversaries using an analytically differentiable renderer. In *International Conference on Machine Learning (ICML)*, pages 224–233, 2018. 2

[17] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2018. 2

[18] Pedram MohajerAnsari, Amir Salarpour, David Fernandez, Cigdem Kokenoz, Bing Li, and Mert D Pesé. Attention-aware temporal adversarial shadows on traffic sign sequences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 1, 2

[19] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against deep learning systems using adversarial examples. *arXiv preprint arXiv:1602.02697*, 1(2):3, 2016. 2

[20] Svetlana Pavlitska, Nico Lambing, and J Marius Zöllner. Adversarial attacks on traffic sign recognition: A survey. In *2023 3rd International conference on electrical, computer, communications and mechatronics engineering (ICECCME)*, pages 1–6. IEEE, 2023. 2

[21] Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K Reiter. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 acm sigsac conference on computer and communications security*, pages 1528–1540, 2016. 2

[22] Chawin Sitawarin, Arjun Nitin Bhagoji, Arsalan Mosenia, Mung Chiang, and Prateek Mittal. Darts: Deceiving autonomous cars with toxic signs. *arXiv preprint arXiv:1802.06430*, 2018. 1

[23] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, 32: 323–332, 2012. 5

[24] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. 1

[25] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021. 4, 5

[26] Donghua Wang, Wen Yao, Tingsong Jiang, Chao Li, and Xiaoqian Chen. Rfla: A stealthy reflected light adversarial attack in the physical world. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4455–4465, 2023. 2

[27] Jie Wang, Zhaoxia Yin, Jing Jiang, and Yang Du. Attention-guided black-box adversarial attacks with large-scale multi-objective evolutionary optimization. *International Journal of Intelligent Systems*, 37(10):7526–7547, 2022. 2

[28] LI Yufeng, YANG Fengyu, LIU Qi, LI Jiangtao, and CAO Chenhong. Light can be dangerous: Stealthy and effective physical-world adversarial attack by spot light. *Computers & Security*, 132:103345, 2023. 1, 2

[29] Yiqi Zhong, Xianming Liu, Deming Zhai, Junjun Jiang, and Xiangyang Ji. Shadows can be dangerous: Stealthy and effective physical-world adversarial attack by natural phenomenon. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15345–15354, 2022. 1, 2, 6