ARMOROCR: GROUNDED ADVERSARIAL VISUAL PERCEPTION VIA OBSERVATION-TRANSFERRED SELF-DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large multimodal models (LMMs) have demonstrated strong OCR recognition capabilities, yet remain vulnerable to adversarial visual text that is readable to humans but challenging for models to localize and recognize. Existing OCR benchmarks mainly focus on natural or document-style text, while adversarial OCR evaluations remain limited in scale, task coverage, or region-aware evaluation. In this paper, we formulate adversarial OCR as a **grounded OCR perception** task and introduce **AdvSpot**, the first benchmark for grounded adversarial OCR evaluation. AdvSpot comprises 383 images with region-level annotations, spanning 5 primary categories and 13 fine-grained adversarial OCR types. To address this challenge, we propose **ArmorOCR**, a two-stage training framework for robust adversarial OCR perception. ArmorOCR first acquires missing adversarial OCR perception from privileged transformed observations through On-Policy Self-Distillation (OPSD), and then refines grounded OCR perception through Group Relative Policy Optimization (GRPO) with task-conditioned rewards for localization, recognition, full spotting, and visual question answering (VQA). Experiments on our AdvSpot, other adversarial OCR benchmarks, and general OCR benchmarks demonstrate that ArmorOCR consistently improves adversarial OCR perception while preserving competitive general OCR capability. The benchmark and code will be publicly available.

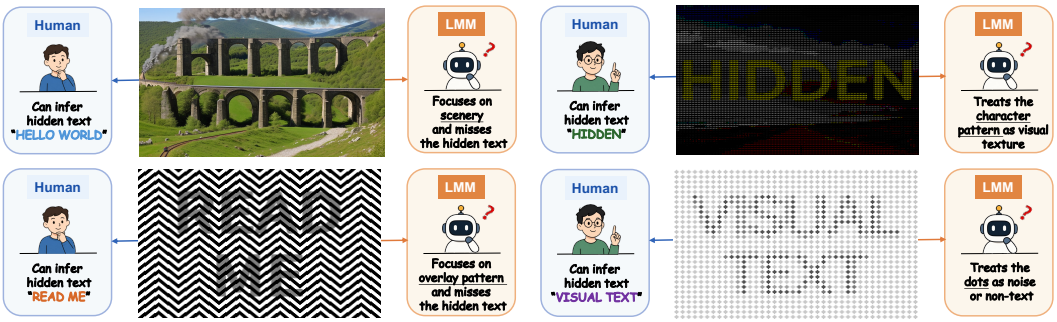


Figure 1: Examples of adversarial OCR patterns that reveal a human–AI perception gap: humans can easily recover the intended text from visual context, while current LMMs may mislocalize, overlook, or misread it.

1 INTRODUCTION

OCR is a fundamental capability of large multimodal models (LMMs) (Zhang et al., 2023; Li et al., 2024; Zhu et al., 2025), yet its reliability under challenging visual conditions remains limited (Li et al., 2025; Gao et al., 2025). As illustrated in Figure 1, certain visual text patterns expose a clear human–AI perception gap: **humans can easily recover the intended text from visual context, whereas current LMMs may mislocalize, overlook, or misread it.** We refer to such patterns as *adversarial OCR patterns*. These patterns expose systematic weaknesses in LMMs' visual text perception, underscoring the need for systematic evaluation and targeted model improvement to bridge this gap and develop more robust OCR capabilities.

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

Existing OCR benchmarks (Liu et al., 2024; Yang et al., 2025; Fu et al., 2026) have advanced general OCR evaluation across diverse natural visual content, but provide limited coverage of adversarial OCR patterns. Recent adversarial OCR benchmarks (Xu et al., 2026; Li et al., 2026) begin to address this setting, but remain limited in scale, task diversity, or region-level annotations. Moreover, recognition-only evaluation provides limited insight into whether a model has identified the relevant text region or merely inferred the answer from global context or linguistic priors.

Motivated by this limitation, we formulate adversarial OCR as a **grounded OCR perception** task, where models are expected to identify relevant text regions, recognize adversarial text content, and answer region-specific questions under challenging visual conditions. This formulation supports region-level verification and more fine-grained analysis of adversarial OCR failures.

To instantiate this formulation, we introduce **AdvSpot**, the first benchmark for grounded adversarial OCR perception. AdvSpot provides the most comprehensive perception taxonomy of adversarial OCR patterns to date, covering 5 primary categories and 13 fine-grained types organized according to their underlying OCR failure mechanisms. It contains 383 images with region-level annotations, including bounding boxes, text transcriptions, category labels, and grounded visual question answering (VQA) pairs. Together, these annotations enable systematic evaluation of grounded adversarial OCR perception with respect to localization, recognition, and region-specific question answering.

Recent approaches (Xu et al., 2026; Li et al., 2025) to adversarial OCR perception rely on "thinking-with-images" strategies (Hong et al., 2025; Song et al., 2025), applying transformations such as cropping, resizing, or flipping at inference time to reveal adversarial text that is difficult to perceive from the original view. Although effective, these approaches introduce additional inference latency and deployment complexity, motivating us to ask: *Can the perceptual information revealed by transformed views be internalized into model parameters during training?*

To answer this question, we propose **ArmorOCR**, a two-stage training framework that internalizes transformation-revealed visual evidence for single-pass grounded adversarial OCR perception, without requiring additional visual transformations or tools at inference time. In Stage 1, we perform On-Policy Self-Distillation (OPSD) with response-region-aware token weighting: the student observes only the original image, while a teacher with the same backbone is conditioned on privileged transformed views. Token-level distribution guidance along the student's on-policy trajectories transfers the perception revealed by privileged observations to the student, establishing a robust perceptual foundation. However, self-distillation remains constrained by the teacher's performance ceiling and lacks explicit optimization of diverse grounded OCR objectives. Stage 2 therefore employs Group Relative Policy Optimization (GRPO) with task-conditioned rewards for localization, recognition, full spotting, and grounded VQA to jointly optimize these complementary capabilities.

Our contributions are summarized as follows:

- We introduce **AdvSpot**, the first benchmark for grounded adversarial OCR perception, with the most comprehensive taxonomy of adversarial OCR patterns to date and region-level annotations enabling fine-grained evaluation of adversarial OCR perception.
- We propose **ArmorOCR**, a two-stage framework that internalizes transformation-revealed perception through privileged observation transfer and refines grounded OCR perception with task-conditioned GRPO.
- Extensive experiments on AdvSpot, existing adversarial OCR benchmarks, and general OCR benchmarks demonstrate that ArmorOCR consistently improves adversarial OCR perception while preserving competitive general OCR capability.

2 Related Work

2.1 OCR Perception Benchmarks

Existing OCR benchmarks (Mathew et al., 2021; Masry et al., 2022; Wang et al., 2024), such as OCRBench (Liu et al., 2024), CCOCR (Yang et al., 2025), and OCRBench-v2 (Fu et al., 2026), evaluate visual text perception across diverse scenarios, including natural scenes, documents, and multilingual content. However, they provide limited coverage of adversarial OCR patterns.

Recent adversarial OCR benchmarks begin to address this gap. AdvOCR (Xu et al., 2026) focuses on adversarial OCR robustness but remains limited in scale and taxonomy, while SmuggleBench (Li et al., 2026) evaluates image-level hidden-text extraction without region grounding or designed question answering. In contrast, our proposed AdvSpot provides bounding boxes, transcriptions, region-grounded VQA and perception-type labels instances under a comprehensive failure-mechanism-based taxonomy.

2.2 Adversarial OCR Perception Methods

Existing approaches to adversarial OCR perception mainly recover difficult visual text at inference time. VACoT (Xu et al., 2026) adopts a "thinking-with-images" paradigm (Chng et al., 2025; Zhang et al., 2025; Zheng et al., 2025b) that dynamically applies visual transformations such as cropping and resizing, while SemVink (Li et al., 2025) shows that zoom-out views can reveal text hidden in AI-generated images. Although effective, these approaches introduce additional inference-time overhead. Li et al. (Li et al., 2026) improve hidden-text extraction through detailed chain-of-thought (CoT) prompting, but such prompt-based strategies require benchmark-specific adaptation. In contrast, our ArmorOCR enables single-pass inference on the original image without additional visual transformations or extended reasoning processes (Hu et al., 2024).

2.3 Self-Distillation and Reinforcement Learning

On-policy self-distillation (OPSD) (Zhao et al., 2026) trains the student on its own sampled trajectories, with a teacher equipped with privileged information to provide token-level distribution guidance. Recent multimodal studies (Yuan et al., 2026; Cai et al., 2026; Tian et al., 2026; Kim et al., 2025) further extend privileged information from textual to visual representations. For example, Vision-OPD (Yuan et al., 2026) transfers fine-grained perception from a crop-conditioned teacher to a full-image student. ArmorOCR extends this principle to adversarial OCR by transferring perception revealed through multiple transformations from a privileged-view teacher to an original-view student.

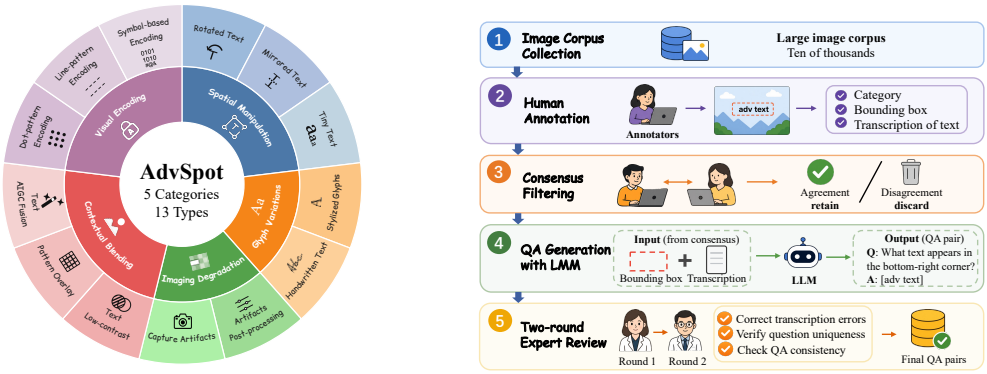
Reinforcement learning has been widely used for LMM alignment by optimizing task-specific rewards (Schulman et al., 2017; Guo et al., 2025; Zheng et al., 2025a; Yu et al., 2026), such as GRPO (Guo et al., 2025), which performs policy optimization using relative rewards among multiple sampled responses without requiring an explicit value model. Such reward-driven optimization is suitable for adversarial OCR perception, where localization, recognition, spotting, and grounded answering can be directly optimized through task-specific rewards.

Table 1: Comparison to prior adversarial OCR benchmarks.

Dimension	**AdvOCR**	**SmuggleBench**	**AdvSpot**
# Images	100	1,700	383
# Perception Types	–	6	**13**
BBox Annotations	✗	✗	✓
Region-aware QA	✗	✗	✓

3 Benchmark: AdvSpot

AdvSpot advances adversarial OCR benchmarking in three respects. First, it offers broader and more systematic coverage of adversarial OCR perception types by organizing them according to their underlying failure mechanisms. Second, it introduces fine-grained annotations, including bounding boxes, transcriptions, perception-type labels, and region-grounded VQA pairs. Third, it provides a region-grounded evaluation framework for adversarial OCR perception, enabling more informative assessment beyond recognition-only evaluation. Table 1 compares AdvSpot with existing adversarial OCR benchmarks.



(a) AdvSpot perception types

(b) Human-in-the-loop pipeline

Figure 2: Overview of AdvSpot: perception types grouped by underlying failure mechanisms and the human-in-the-loop construction pipeline.

3.1 Task Definition

AdvSpot instantiates grounded adversarial OCR perception as a region-grounded VQA task. Each instance is represented as

$$\mathcal{S} = (x, \mathcal{C}^\star, b^\star, t^\star, q, a^\star), \tag{1}$$

where x is an image containing adversarial OCR patterns, $\mathcal{C}^\star$ is the set of perception-type labels associated with the annotated text region $b^\star$, and $t^\star$ is its transcription. The question q uniquely refers to the region–text pair $(b^\star, t^\star)$ through visual attributes or spatial context, and $a^\star$ is the corresponding answer.

During evaluation, the model receives only (x, q). Answering correctly requires the model to ground the question to the relevant text region, recognize the adversarial text, and provide a region-specific answer. This formulation enables grounded evaluation of adversarial OCR perception.

3.2 Taxonomy

AdvSpot comprises 5 primary categories and **13 fine-grained** adversarial OCR types grouped by perception-failure mechanisms, as summarized in Figure 2a. The number of subtypes in each category, along with detailed definitions and examples, is provided in Appendix A.1.

3.3 Annotation and QA Construction

We construct AdvSpot through a *human-in-the-loop* pipeline, as illustrated in Figure 2b. We first collect tens of thousands of candidate images potentially containing adversarial OCR patterns. Each image is independently annotated by two annotators with bounding boxes, transcriptions, and perception-type labels according to our taxonomy. We retain only regions with consistent annotations from both annotators and discard images without agreed regions. For each retained region, we use Qwen3-VL-235B-A22B-Instruct to generate a region-grounded question–answer pair conditioned on the image, bounding box, and transcription; the detailed prompt is provided in Appendix A.2. Each QA pair then undergoes two rounds of expert review to correct errors, verify that the question uniquely refers to the target region, and ensure that the answer is consistent with the verified transcription.

3.4 Evaluation Protocol

We report VQA accuracy as the primary metric and the intersection over union (IoU) between predicted and ground-truth bounding boxes as a localization metric:

$$\mathrm{Acc}(\hat{a}, a^\star) = \mathbb{I}[a^\star \preceq_{\mathrm{sub}} \hat{a}], \qquad \mathrm{IoU}(\hat{b}, b^\star) = \frac{|\hat{b} \cap b^\star|}{|\hat{b} \cup b^\star|}. \tag{2}$$

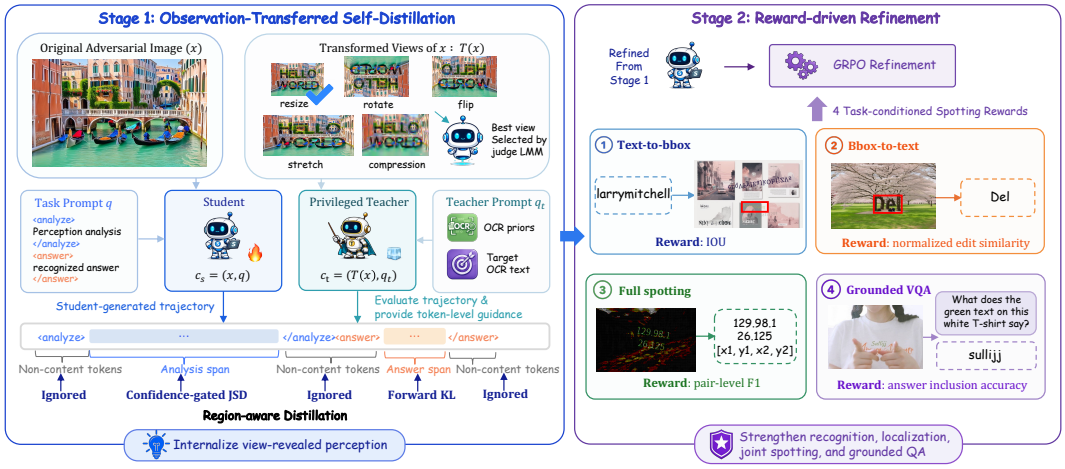


Figure 3: Overview of **ArmorOCR**, a two-stage framework for grounded adversarial OCR perception. Stage 1 transfers transformation-induced perception from privileged observations via OPSD, while Stage 2 optimizes grounded OCR perception through task-conditioned GRPO rewards for localization, recognition, full spotting, and grounded VQA.

Here, $u \preceq_{\text{sub}} v$ means that u is a contiguous substring of v; thus, additional text is allowed in the prediction. Evaluation prompts are provided in Appendix A.3.

4 Method: ArmorOCR

We propose **ArmorOCR** (Figure 3), a two-stage framework for adversarial OCR perception via observation transfer and reward-driven refinement. ArmorOCR is based on the observation that adversarial OCR failures arise from two challenges: (1) the original view may hide critical visual cues required to perceive adversarial text, and (2) adversarial perception alone is insufficient for completing grounded OCR tasks that require localization, recognition, and grounded answering. Therefore, ArmorOCR first transfers transformation-revealed perception from privileged observations to the student through OPSD, and then optimizes grounded OCR perception through GRPO with task-conditioned rewards.

4.1 Stage 1: Observation-Transferred Self-Distillation

Motivated by the observation that adversarial OCR difficulty is often view-dependent (Li et al., 2025), we aim to transfer the perceptual advantages revealed by transformed views into the model during training. Building upon OPSD, we propose Observation-Transferred Self-Distillation (OTSD), where a teacher conditioned on privileged transformed views guides the student.

Let x be the original adversarial image and q be the task prompt. The prompt q instructs the model to provide an intermediate perception analysis within `<analyze>` and `</analyze>` tags, followed by the final recognized answer within `<answer>` and `</answer>` tags. The student sees the original image, whereas the teacher receives a privileged transformed view:

$$c_s = (x, q), \qquad c_t = (\mathcal{T}(x), q_t). \tag{3}$$

Here, $\mathcal{T}(x)$ denotes a transformed view of x. We consider five transformations: resizing, stretching, rotation, flipping, and compression. For each image, the judge LMM, Qwen3-VL-235B-A22B-Instruct, selects the transformed view with the highest recognition accuracy.

The teacher prompt q_t further extends q with adversarial OCR priors and the target OCR text, enabling the teacher to understand the adversarial visual pattern and provide reliable token-level answer guidance. Further details are provided in Appendix B.1.

The student first samples an on-policy response:

$$\hat{y} \sim \pi_\theta(\cdot \mid c_s). \tag{4}$$

Table 2: Fine-grained accuracy and average localization IoU on AdvSpot. All values are reported in %. The best and second-best results are highlighted in **bold** and underlined, respectively. **Avg. Acc.** and **Avg. IoU** denote the sample-weighted averages of accuracy and IoU, respectively. Category-wise IoU results are provided in **Appendix A.4**. The same conventions apply to subsequent tables.

Category	Subtype	*Open-source Qwen3-VL*				*Proprietary LMMs*				*Our LMM*
		8B *(base)*	30B-A3B	32B	235B-A22B	GPT-5.1	Gemini 3 Flash	GLM 5.3 Flash	Kimi-K3	**Armor-OCR**
Imaging Degradation	Capture Artifacts	40.0	53.3	53.3	63.3	26.7	**66.7**	50.0	53.3	43.3
	Post-processing	53.3	53.3	50.0	60.0	43.3	**80.0**	40.0	56.7	60.0
Spatial Manipulation	Rotated Text	48.6	48.6	54.3	60.0	28.6	**62.9**	48.6	51.4	54.3
	Mirrored Text	20.0	26.7	20.0	23.3	13.3	13.3	43.3	30.0	**46.7**
	Tiny Text	60.0	54.3	51.4	57.1	45.7	**68.6**	45.7	**68.6**	57.1
Glyph Variations	Stylized Glyphs	30.0	50.0	46.7	53.3	30.0	56.7	60.0	**70.0**	26.7
	Handwritten Text	36.7	53.3	40.0	53.3	30.0	56.7	53.3	**60.0**	46.7
Visual Encoding	Symbol Encoding	0.0	0.0	0.0	5.7	31.4	20.0	5.7	2.9	**68.6**
	Dot Encoding	23.3	23.3	20.0	23.3	33.3	33.3	26.7	13.3	**63.3**
	Line Encoding	2.9	5.7	5.7	14.3	11.4	11.4	0.0	2.9	**54.3**
Contextual Blending	AIGC Fusion Text	5.7	8.6	5.7	11.4	5.7	17.1	11.4	5.7	**82.9**
	Low Contrast Text	45.7	45.7	48.6	48.6	28.6	**74.3**	45.7	48.6	42.9
	Pattern Overlay	26.7	30.0	23.3	26.7	13.3	36.7	36.7	3.3	**66.7**
Avg. **Acc.**	–	29.8	34.7	31.9	38.3	26.5	45.2	35.2	35.0	**57.1**
Avg. **IoU**	–	54.4	55.0	61.1	58.6	27.8	59.2	40.6	54.9	**63.2**

The teacher then evaluates the student-generated trajectory under privileged observations. At token position i, the student and teacher distributions are:

$$p_s^i = \pi_\theta(\cdot \mid c_s, \hat{y}_{<i}), \quad p_t^i = \pi'_\theta(\cdot \mid c_t, \hat{y}_{<i}). \tag{5}$$

Vanilla OPSD uniformly distills all response tokens, overlooking their distinct roles in the final OCR prediction. Analysis tokens encode potentially uncertain intermediate perception, answer tokens directly determine the transcription, and structural tokens contain no task-relevant content. We therefore introduce a **response-region-aware distillation loss**, partitioning the response into an analysis span $\mathcal{R}_{\text{ana}}$, an answer span $\mathcal{R}_{\text{ans}}$, and non-content structural tokens using predefined tags.

For analysis tokens, uniformly applying distillation may propagate uncertain or noisy intermediate reasoning. We therefore use confidence-gated Jensen–Shannon divergence (JSD) (Lu et al., 2026) to selectively transfer reliable teacher signals:

$$\mathcal{L}_{\text{ana}}^i = g_i D_{\text{JSD}}^\beta(p_t^i \| p_s^i), \quad i \in \mathcal{R}_{\text{ana}}, \tag{6}$$

where D_{JSD}^β denotes the generalized JSD with mixture coefficient β. The token-level confidence weight is defined as

$$g_i = \sigma\left(\gamma\left[\max_k \log p_t^i(k) - \max_k \log p_s^i(k)\right]\right), \tag{7}$$

where $\sigma(\cdot)$ is the sigmoid function and γ controls the sharpness of the confidence gate. The weight g_i increases when the privileged teacher is more confident than the student, strengthening distillation at positions with reliable teacher guidance while suppressing uncertain signals.

For answer tokens, which directly determine the final OCR transcription, we apply stronger supervision using forward KL divergence:

$$\mathcal{L}_{\text{ans}}^i = \text{KL}(p_t^i \| p_s^i), \quad i \in \mathcal{R}_{\text{ans}}. \tag{8}$$

Conditioned on privileged observations and OCR guidance, the teacher provides a reliable target answer distribution. We assign uniform weights to all answer tokens, as each contributes to the final transcription.

Non-content tokens, such as structural markers, are excluded from the loss because they encode only the output format and do not contribute to adversarial text perception. The resulting objective selectively transfers intermediate perception while imposing direct supervision on the final OCR output:

$$\mathcal{L}_{\text{OTSD}} = \frac{\sum_{i \in \mathcal{R}_{\text{ana}}} \mathcal{L}_{\text{ana}}^i + \sum_{i \in \mathcal{R}_{\text{ans}}} \mathcal{L}_{\text{ans}}^i}{\sum_{i \in \mathcal{R}_{\text{ana}}} g_i + |\mathcal{R}_{\text{ans}}| + \epsilon}. \tag{9}$$

Table 3: Results on other adversarial OCR benchmarks: AdvOCR and SmuggleBench.

Benchmark	Task	*Open-source Qwen3-VL*				*Closed-source LMMs*		*Specialized LMMs*		
		8B *(Our Base)*	30B-A3B	32B	235B-A22B	GPT-5	Gemini 2.5 Flash	VACoT	Smuggle-CoT	**Armor-OCR**
AdvOCR	Real-world	30.0	34.0	44.0	44.0	12.0	34.0	**62.0**	–	44.0
	Synthetic	12.0	12.0	8.0	12.0	22.0	12.0	48.0	–	**68.0**
	Avg.	21.0	23.0	26.0	28.0	17.0	23.0	55.0	–	**56.0**
SmuggleBench	Tiny Text	26.1	25.9	23.6	25.1	18.2	26.3	–	30.2	**33.0**
	Occluded Text	18.2	23.2	19.1	19.6	9.7	24.1	–	**26.1**	17.5
	Low Contrast	3.0	8.8	3.5	6.5	4.0	**12.5**	–	8.5	6.5
	Handwritten	31.7	30.4	29.3	**33.7**	15.8	15.4	–	31.7	27.0
	Artistic	14.1	14.5	11.6	15.6	11.1	8.7	–	18.6	**19.0**
	AI Illusions	0.0	0.3	0.8	0.8	0.3	0.0	–	0.0	**8.2**
	Avg.	13.3	14.8	14.3	14.6	8.5	12.4	–	16.4	**17.1**

4.2 Stage 2: Reward-driven Refinement

Stage 1 establishes the model's ability to perceive adversarial text, but teacher-guided distillation remains constrained by teacher supervision and does not directly optimize the diverse objectives required by grounded OCR tasks. We therefore apply GRPO with task-conditioned rewards to align the model with grounded adversarial OCR perception. Specifically, we construct four complementary tasks: text-to-bbox for localization, bbox-to-text for recognition, full spotting for joint region–text prediction, and grounded VQA for region-specific question answering.

For **text-to-bbox**, the model performs text-conditioned region localization by predicting the bounding box corresponding to a given text string. The localization reward is defined as the IoU between the predicted and ground-truth bounding boxes:

$$R_{\text{t2b}} = \text{IoU}(\hat{b}, b^\star). \tag{10}$$

For **bbox-to-text**, the model performs region-conditioned text recognition by predicting the transcription corresponding to a given bounding box. Let $\hat{t}$ and $t^\star$ denote the predicted and ground-truth transcriptions, respectively. We adopt normalized Levenshtein similarity as the recognition reward:

$$R_{\text{b2t}} = 1 - \frac{\text{ED}(\hat{t}, t^\star)}{\max(|\hat{t}|, |t^\star|)}, \tag{11}$$

where $\text{ED}(\cdot, \cdot)$ denotes the Levenshtein edit distance.

For **full spotting**, the model jointly predicts text regions and their corresponding transcriptions. We greedily match predicted and reference pairs if both the localization and recognition criteria are satisfied, and use pair-level F1 as the reward:

$$\text{IoU}(\hat{b}, b^\star) \geq \theta_{\text{iou}}, \quad \text{Sim}(\hat{t}, t^\star) \geq 1 - \theta_{\text{edit}}; \qquad R_{\text{joint}} = \frac{2PR}{P + R}. \tag{12}$$

We set $\theta_{\text{iou}} = 0.5$ and $\theta_{\text{edit}} = 0.1$; $\text{Sim}(\cdot, \cdot)$ denotes normalized Levenshtein similarity, and P and R denote precision and recall over matched pairs.

For **grounded VQA**, the model answers questions grounded on the annotated adversarial text region. We use normalized answer inclusion as the reward:

$$R_{\text{vqa}} = \mathbb{I}[a^\star \preceq_{\text{sub}} \hat{a}]. \tag{13}$$

Given the sampled response $\hat{y}$, its ground-truth target $y^\star$, and the task type τ, the final reward is defined as:

$$R(\hat{y}, y^\star, \tau) = \begin{cases} R_{\text{t2b}}, & \tau = \texttt{text_to_bbox}, \\ R_{\text{b2t}}, & \tau = \texttt{bbox_to_text}, \\ R_{\text{joint}}, & \tau = \texttt{full_spotting}, \\ R_{\text{vqa}}, & \tau = \texttt{vqa}. \end{cases} \tag{14}$$

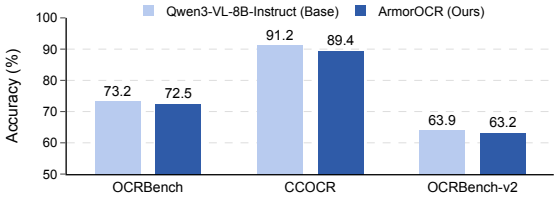


Figure 4: General OCR performance of ArmorOCR.

Table 4: Comparison of Stage 1 training methods on AdvSpot.

Method	IoU	Acc.
Base model	54.4	29.8
SFT	49.2	32.9
OPSD	54.5	33.9
GRPO	56.6	37.3
OTSD (ours)	**58.7**	**46.8**

5 Experiments

5.1 Experimental Setup

Benchmarks. We evaluate models on two groups of benchmarks. For **adversarial OCR perception**, we evaluate models on our proposed AdvSpot and two existing adversarial OCR-related benchmarks, AdvOCR (Xu et al., 2026) and SmuggleBench (Li et al., 2026). For **general OCR perception**, we further evaluate on three widely used general OCR benchmarks: CCOCR (Yang et al., 2025), OCRBench (Liu et al., 2024), and OCRBench-v2 (Fu et al., 2026).

Models. We compare open-source and proprietary LMMs. For open-source models, we evaluate the Qwen3-VL series at different scales (Bai et al., 2025). For proprietary models, we include GPT-5.1 (OpenAI, 2025) and Gemini-3 Flash (Comanici et al., 2025), GLM-5.3-Flash (Zeng et al., 2026) and Kimi-K3 (Team et al., 2026). Additional benchmark results are reported in Appendix A.4. For existing adversarial OCR benchmarks, we also compare benchmark-specific methods: VACoT (Xu et al., 2026) for AdvOCR and Smuggle-CoT (Li et al., 2026) for SmuggleBench[1].

Training Details. Due to the scarcity of large-scale adversarial OCR data, we synthesize stage-specific training sets with controllable adversarial patterns. Stage 1 uses 50K examples to establish adversarial OCR perception, while Stage 2 uses 70K samples enriched with harder adversarial cases for fine-grained grounded OCR optimization. ArmorOCR is built on Qwen3-VL-8B-Instruct. In Stage 1, we keep the teacher frozen and train the student with OTSD. In Stage 2, we apply GRPO with task-conditioned rewards and sample eight responses per prompt. Both stages are trained on PPU-810E accelerators. All benchmarks are evaluated zero-shot. Data generation and training configurations are detailed in Appendix C.1 and C.2, respectively.

Metrics. For AdvSpot, we report accuracy and IOU. For AdvOCR, we report pass@1 accuracy evaluated by Qwen3-30B-A3B-2507 (Bai et al., 2025), following its official evaluation protocol. For SmuggleBench, we focus on its "Perceptual Blindness" subset, which is most aligned with our task, and report the TER metric introduced in the original paper. For the three general OCR benchmarks, we adopt the open-source evaluation toolkit VLMEvalKit (Duan et al., 2024), whose results are largely consistent with the official metrics, and report the overall benchmarks scores.

5.2 Performance Analysis

Results on AdvSpot. As shown in Table 2, both open-source and proprietary LMMs struggle with several AdvSpot subtypes. Symbol Encoding and AIGC Fusion Text remain highly challenging, with most models achieving nearly zero accuracy. These failures indicate that current LMMs are still vulnerable to adversarial OCR patterns, especially those involving visual encoding and contextual blending. In contrast, ArmorOCR achieves the best overall performance, improving over its base model by 27.3%. The gain is particularly pronounced on AIGC Fusion Text, where ArmorOCR outperforms the strongest competing model by 65.8%. ArmorOCR also achieves the highest average localization IoU, indicating its strong capability in both region localization and grounded adversarial OCR perception. Moreover, the mismatch between IoU and accuracy trends across different models suggests that answer accuracy alone does not guarantee reliable grounding. For example, GLM-5.3-Flash and Kimi-K3 achieve similar accuracy, yet GLM-5.3-Flash obtains only 40.6% IoU, 14.3%

[1]Since VACoT is not open-sourced, we report its official results. Smuggle-CoT refers to the SmuggleBench-specific CoT-based test-time scaling method using Qwen3-VL-235B-A22B-Instruct with detailed prompting.

lower than Kimi-K3. This highlights the importance of jointly evaluating localization and perception for fine-grained analysis of adversarial OCR failures.

Results on Other Adversarial OCR Benchmarks. As shown in Table 3, ArmorOCR achieves the highest average scores on both AdvOCR and SmuggleBench. On AdvOCR, ArmorOCR outperforms the tool-assisted VACoT by 20% on the synthetic split, which focuses on more challenging adversarial patterns. On SmuggleBench, ArmorOCR achieves better overall performance than Smuggle-CoT, with a notable gain of 8.2% on the AI Illusions category, despite the latter relying on a much larger 235B model with detailed CoT prompting. These results demonstrate the effectiveness and robustness of ArmorOCR across diverse adversarial OCR scenarios.

General OCR Capability. A potential concern is that improving adversarial OCR perception may degrade general OCR ability. We therefore evaluate both the base model and ArmorOCR on three general OCR benchmarks. As shown in Figure 4, ArmorOCR maintains comparable performance with the base model across all benchmarks, despite using no additional general OCR training data. This demonstrates that our training strategy improves adversarial OCR robustness while largely preserving general OCR capability.

Table 5: Stage 1 component ablations.

Metric	Base model	w/o visual transfer	w/o region-aware distillation	Full Stage 1
IoU	54.4	54.0	54.5	**58.7**
Acc.	29.8	34.7	44.2	**46.8**

Table 6: Stage 2 reward ablations.

Metric	w/o $R_{\mathtt{t2b}}$	w/o $R_{\mathtt{b2t}}$	w/o $R_{\mathtt{joint}}$	w/o $R_{\mathtt{vqa}}$	Full Stage 2
IoU	61.9	60.3	62.8	62.4	**63.2**
Acc.	54.8	51.2	56.6	53.7	**57.1**

5.3 Ablation Study

We conduct all ablation studies on AdvSpot, as it provides a comprehensive evaluation setting for grounded adversarial OCR perception.

Stage 1 Training-Method Comparison. We compare OTSD with SFT, standard OPSD, and GRPO using the same 50K Stage 1 training examples. As shown in Table 4, SFT, standard OPSD, and GRPO improve VQA accuracy from 29.8% for the base model to 32.9%, 33.9%, and 37.3%, respectively, but all remain below OTSD at 46.8%. These results suggest that the Stage 1 gains reflect the training design, rather than exposure to synthetic data alone. The component ablations in Table 5 further support the importance of transferring perception revealed by transformed views through response-region-aware distillation.

Stage 1 Component Ablations. As shown in Table 5, we further analyze the two key designs in Stage 1: visual transfer and response-region-aware distillation. Removing visual transfer conditions the teacher only on the original image and privileged textual information, yielding a marginal accuracy improvement of 4.9%, thus indicating that **privileged transformed observations provide more effective supervision than textual information alone.** Replacing our response-region-aware objective with the standard OPSD JSD loss also degrades performance, demonstrating that importance-guided selective distillation enables more effective knowledge transfer.

Stage 2 Reward Ablations. As shown in Table 6, we finally evaluate the contribution of each task-conditioned reward in Stage 2. Removing any reward leads to a performance drop, showing that recognition, localization, spotting, and VQA objectives each provide useful supervision. Their complementary effects jointly strengthen grounded adversarial OCR perception.

6 Conclusion

In this work, we introduced **AdvSpot**, a grounded adversarial OCR benchmark with a comprehensive taxonomy for evaluating adversarial OCR perception. AdvSpot goes beyond existing OCR benchmarks by focusing on visually manipulated text that is readable to humans but challenging for LMMs. We further proposed **ArmorOCR**, a two-stage framework for robust grounded adversarial OCR perception. ArmorOCR first internalizes transformation-revealed perception through observation-transferred self-distillation, and then uses task-conditioned GRPO rewards to jointly optimize localization, recognition, spotting, and grounded VQA. Experiments demonstrate that ArmorOCR consistently improves adversarial OCR perception while preserving general OCR capability. We hope AdvSpot and ArmorOCR will facilitate future research on reliable and generalizable visual text perception under challenging conditions.

AI USE STATEMENT

In this work, we used generative AI tools for benchmark construction and the generation and filtering of synthetic training data, as described in Section 3.3 and Appendix C.1. We also used AI tools to polish the writing of this paper. We did not use generative AI tools for research design, experimental procedures, or analysis. We take responsibility for the final content of this work, including text and figures produced with the aid of generative AI.

ETHICS STATEMENT

We reviewed the AdvSpot images, OCR transcriptions, annotations, and question–answer pairs to ensure that the benchmark does not contain real-world personally identifiable information (PII), such as private phone numbers, physical addresses, or email addresses. We also screened the benchmark for potentially harmful or unlawful content and checked the use of its source images against applicable data licenses and privacy requirements.

REPRODUCIBILITY STATEMENT

The main paper provides a detailed description of AdvSpot's taxonomy, annotation procedure, and evaluation protocol (Session 3), as well as the design of our ArmorOCR method (Session 4). The appendix provides further details on benchmark construction (Appendix A), the training-data synthesis pipeline (Appendix C), and the core prompts used in our experiments (Appendix B). We plan to release the benchmark and code upon publication.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Yishuo Cai, Jiahui Liu, Yuanxin Liu, Haobo Deng, Linli Yao, Yuhao Zheng, Kun Ouyang, Zhimo Li, Ziyue Wang, Xu Sun, et al. Thinking without images: Internalizing visual manipulation with on-policy self-distillation. *arXiv preprint arXiv:2606.08719*, 2026.

Yong Xien Chng, Tao Hu, Wenwen Tong, Xueheng Li, Jiandong Chen, Haojia Yu, Jiefan Lu, Hewei Guo, Hanming Deng, Chengjun Xie, et al. Sensenova-mars: Empowering multimodal agentic reasoning and search via reinforcement learning. *arXiv preprint arXiv:2512.24330*, 2025.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 11198–11201, 2024.

Ling Fu, Zhebin Kuang, Jiajun Song, Mingxin Huang, Biao Yang, Yuzhe Li, Linghao Zhu, Qidi Luo, Xinyu Wang, Hao Lu, et al. Ocrbench v2: An improved benchmark for evaluating large multimodal models on visual text localization and reasoning. *Advances in Neural Information Processing Systems*, 38, 2026.

Hongcheng Gao, Zihao Huang, Lin Xu, Jingyi Tang, Xinhao Li, Yue Liu, Haoyang Li, Taihang Hu, Minhua Lin, Xinlong Yang, et al. Pixels, patterns, but no poetry: To see the world like humans. *arXiv preprint arXiv:2507.16863*, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Jack Hong, Chenxiao Zhao, ChengLin Zhu, Weiheng Lu, Guohai Xu, and Xing Yu. Deepeyesv2: Toward agentic multimodal model. *arXiv preprint arXiv:2511.05271*, 2025.

Yushi Hu, Otilia Stretcu, Chun-Ta Lu, Krishnamurthy Viswanathan, Kenji Hata, Enming Luo, Ranjay Krishna, and Ariel Fuxman. Visual program distillation: Distilling tools and programmatic reasoning into vision-language models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9590–9601. IEEE, 2024.

Sanghwan Kim, Rui Xiao, Mariana-Iuliana Georgescu, Stephan Alaniz, and Zeynep Akata. Cosmos: Cross-modality self-distillation for vision language pre-training. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14690–14700. IEEE, 2025.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

Sifan Li, Yujun Cai, and Yiwei Wang. Semvink: Advancing vlms' semantic understanding of optical illusions via visual global thinking. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 27155–27165, 2025.

Zhiheng Li, Zongyang Ma, Yuntong Pan, Ziqi Zhang, Xiaolei Lv, Bo Li, Jun Gao, Jianing Zhang, Chunfeng Yuan, Bing Li, et al. Making mllms blind: Adversarial smuggling attacks in mllm content moderation. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 20142–20161, 2026.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 67(12):220102, 2024.

Zhengxi Lu, Zhiyuan Yao, Zhuowen Han, Zi-Han Wang, Jinyang Wu, Qi Gu, Xunliang Cai, Weiming Lu, Jun Xiao, Yueting Zhuang, et al. Self-distilled agentic reinforcement learning. *arXiv preprint arXiv:2605.15155*, 2026.

Ahmed Masry, Jia Qing Tan, Shafiq Joty, Enamul Hoque, et al. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the association for computational linguistics: ACL 2022*, pp. 2263–2279, 2022.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 2200–2209, 2021.

OpenAI. Gpt-5.1. `https://openai.com/index/gpt-5-1/`, 2025.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Qi Song, Honglin Li, Yingchen Yu, Haoyi Zhou, Lin Yang, Song Bai, Qi She, Zilong Huang, and Yunqing Zhao. Codedance: A dynamic tool-integrated mllm for executable visual reasoning. *arXiv preprint arXiv:2512.17312*, 2025.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, Y Charles, et al. Kimi k3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026.

Kanghui Tian, Siyuan Liu, Ziang Yan, Sheng Xia, Shuai Dong, and Yi Wang. Vicur: Visual cues as recoverable privilege for multimodal on-policy distillation. *arXiv preprint arXiv:2606.05718*, 2026.

Zirui Wang, Mengzhou Xia, Luxi He, Howard Chen, Yitao Liu, Richard Zhu, Kaiqu Liang, Xindi Wu, Haotian Liu, Sadhika Malladi, et al. Charxiv: Charting gaps in realistic chart understanding in multimodal llms. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

Zhengzhuo Xu, Chong Sun, SiNan Du, Chen Li, Jing Lyu, and Chun Yuan. Vacot: Rethinking visual data augmentation with vlms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9780–9790, 2026.

Zhibo Yang, Jun Tang, Zhaohai Li, Pengfei Wang, Jianqiang Wan, Humen Zhong, Xuejing Liu, Mingkun Yang, Peng Wang, Shuai Bai, et al. Cc-ocr: A comprehensive and challenging ocr benchmark for evaluating large multimodal models in literacy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 21744–21754, 2025.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *Advances in Neural Information Processing Systems*, 38:113222–113244, 2026.

Qianhao Yuan, Jie Lou, Xing Yu, Hongyu Lin, Le Sun, Xianpei Han, and Yaojie Lu. Vision-opd: Learning to see fine details for multimodal llms via on-policy self-distillation. *arXiv preprint arXiv:2605.18740*, 2026.

Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, et al. Glm-5: from vibe coding to agentic engineering. *arXiv preprint arXiv:2602.15763*, 2026.

Pan Zhang, Xiaoyi Dong, Bin Wang, Yuhang Cao, Chao Xu, Linke Ouyang, Zhiyuan Zhao, Haodong Duan, Songyang Zhang, Shuangrui Ding, et al. Internlm-xcomposer: A vision-language large model for advanced text-image comprehension and composition. *arXiv preprint arXiv:2309.15112*, 2023.

YiFan Zhang, Xingyu Lu, Shukang Yin, Chaoyou Fu, Wei Chen, Xiao Hu, Bin Wen, Kaiyu Jiang, Changyi Liu, Tianke Zhang, et al. Thyme: Think beyond images. In *NeurIPS 2025 Fourth Workshop on Deep Learning for Code*, 2025.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, et al. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025a.

Ziwei Zheng, Michael Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, Chao Shen, and Xing Yu. Deepeyes: Incentivizing" thinking with images" via reinforcement learning. *arXiv preprint arXiv:2505.14362*, 2025b.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

A MORE DETAILS OF OUR BENCHMARK

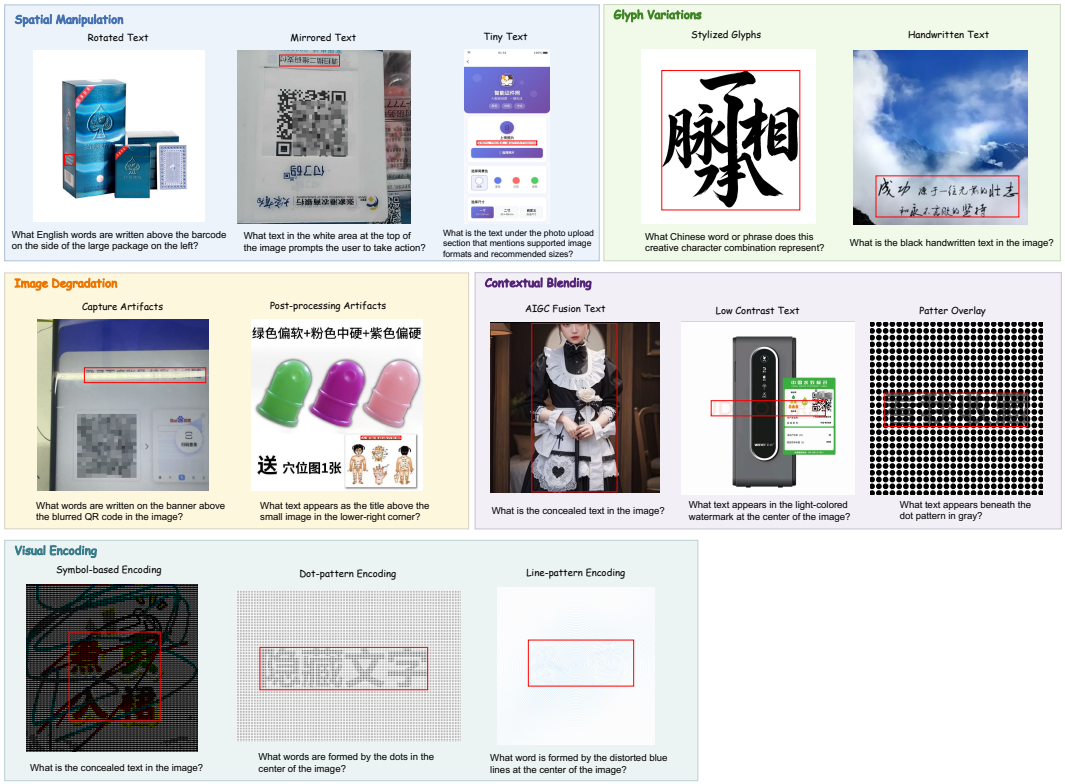


Figure 5: Representative examples of the 5 primary categories and 13 fine-grained subtypes in AdvSpot. Each example includes a grounded VQA pair for region-level adversarial OCR perception evaluation.

A.1 DETAILED DEFINITION OF EACH TAXONOMY

Our taxonomy is organized according to the underlying OCR failure mechanisms. Specifically, AdvSpot categorizes adversarial OCR patterns into 5 primary categories and 13 fine-grained subtypes, as detailed below.

- **Spatial Manipulation.** This category contains adversarial patterns that alter the spatial layout or geometric properties of text, making localization and recognition more challenging.
  - **Rotated Text.** Text is rotated with arbitrary orientations or perspective distortions, requiring the model to recognize text under diverse geometric transformations.
  - **Mirrored Text.** Text is horizontally or vertically flipped, violating the normal reading direction.
  - **Tiny Text.** The text occupies only a very small portion of the image, making it difficult to localize and recognize.
- **Glyph Variations.** This category modifies the appearance of characters while preserving their semantic meaning.
  - **Stylized Glyphs.** Characters are intentionally distorted through stroke splitting, occlusion, truncation, or other artistic modifications while remaining readable to humans.
  - **Handwritten Text.** Non-standard handwritten characters exhibiting connected strokes, exaggerated deformation, or irregular writing styles.
- **Image Degradation.** This category reduces OCR performance through imaging artifacts introduced during image acquisition or post-processing.

Table 7: Distribution of AdvSpot samples across the 5 primary categories and 13 fine-grained subtypes.

Category	Subtype	Number of Samples
Spatial Manipulation	Rotated Text	35
	Mirrored Text	30
	Tiny Text	35
Glyph Variations	Stylized Glyphs	30
	Handwritten Text	30
Image Degradation	Capture Artifacts	30
	Post-processing Artifacts	30
Contextual Blending	AIGC Fusion Text	35
	Low Contrast Text	35
	Pattern Overlay	30
Visual Encoding	Symbol-based Encoding	35
	Dot-pattern Encoding	30
	Line-pattern Encoding	35

  - **Capture Artifacts.** Imaging degradations caused during acquisition, such as motion blur, reflections, film grain, over-/under-exposure, or excessive filtering.
  - **Post-processing Artifacts.** Image quality degradation introduced after capture, including blur, compression artifacts, additive noise, and other image processing operations.
- **Contextual Blending.** This category conceals text by reducing its visual distinguishability from the surrounding context.
  - **AIGC Fusion Text.** AI-generated images seamlessly blend text with surrounding visual content, making text boundaries difficult to distinguish.
  - **Low Contrast Text.** Text exhibits low contrast with its background due to similar colors or high transparency.
  - **Pattern Overlay.** Lines, dot patterns, textures, or other visual patterns are superimposed on text, interfering with character perception.
- **Visual Encoding.** This category encodes textual information into non-standard visual patterns instead of conventional glyphs.
  - **Symbol-based Encoding.** Characters are represented through the spatial arrangement of symbols or special characters.
  - **Dot-pattern Encoding.** Characters are encoded using the positions, sizes, or densities of dot patterns.
  - **Line-pattern Encoding.** Characters are encoded through line orientation, length, width, or connectivity.

Figure 5 presents representative examples for each fine-grained subtype in our taxonomy. These examples illustrate the diverse adversarial OCR patterns covered by AdvSpot and highlight the distinct visual challenges introduced by different failure mechanisms.

Table 7 summarizes the distribution of AdvSpot across different taxonomy categories and fine-grained subtypes. Counts are computed at the adversarial pattern level, where a single image or VQA instance may contain multiple adversarial patterns. AdvSpot contains 383 images with 389 grounded VQA pairs in total.

A.2 Detailed Construction of VQA Pairs

For each retained text region, we construct a grounded visual question answering (VQA) pair using Qwen3-VL-235B-A22B-Instruct. The model receives the original image together with the target bounding box and its OCR transcription as privileged information, and is instructed to generate a question whose unique answer is exactly the provided transcription.

As shown in Figure 6, the prompt explicitly requires the generated question to (1) uniquely identify the target region using visual attributes (e.g., object category, color, shape, size, or relative position), (2) avoid any ambiguity with other text regions in the image, (3) never expose the OCR transcription or bounding-box coordinates in the question itself, and (4) ensure that the provided transcription is the only correct answer. These constraints encourage visually grounded questions that require both accurate localization and text recognition, while preventing shortcuts based on coordinate information or answer leakage.

Prompt 6: System Prompt for Grounded VQA Data Construction

You are an expert VQA data annotator.
Your task is to construct Visual Question Answering (VQA) training data based on a specified text region in an image.
Target Region:
The bounding box coordinates are: [{x_min}, {y_min}, {x_max}, {y_max}].
The coordinate format is [x_min, y_min, x_max, y_max], normalized to the range of 0--1000 with respect to the image width and height.
The OCR text within this region is: {text_content}.
Your Task:

1. **Visual Analysis.**
   Observe the target region and identify its visual characteristics, including object type, color, shape, size, and relative position.
2. **Question Construction.**
   Construct a question asking about the text content within the target region.
3. **Unique Grounding Requirement.**
   The question must uniquely identify the target region in the entire image and should not introduce ambiguity with other regions.
4. **Content Constraints.**
   The question must not contain coordinate values, coordinate numbers, or the OCR text itself. The only correct answer to the question must exactly match the provided OCR text.

Output Format Requirement

Question: [Your generated question]

Important Notes:
The question must never reveal the provided OCR text or use coordinates to refer to the target region. The OCR text is only provided to understand the semantic role of the target region and its relationship with other visual elements in the image.

Figure 6: The system prompt used for constructing grounded VQA training data.

A.3 Detailed Testing Prompt

Figure 7 presents the testing prompt used throughout all experiments. Given a grounded VQA question, the model is instructed to output the recognized text and its corresponding bounding box

Prompt 2: Detailed Testing Prompt for Grounded OCR Perception Evaluation

```
Please output the answer and the corresponding text bounding box in
JSON format:
{"text":"answer content","bbox":[x_min,y_min,x_max,y_max]}
The coordinate values are in the range of 0--1000.
```

Figure 7: The testing prompt used for grounded OCR perception evaluation.

in a predefined JSON format. The bounding box coordinates are normalized to the range of 0–1000 with respect to the image size.

A.4 More Experiments on AdvSpot

To provide a more comprehensive evaluation, we report the complete benchmarking results on AdvSpot across an expanded collection of both open-source and proprietary LMMs. Specifically, Table 8 presents the localization IoU results, while Table 9 reports the corresponding grounded VQA accuracy.

B More Details of Method

B.1 More Details of Stage 1

Figure 8 and 9 illustrates two representative training examples of Observation-Transferred Self-Distillation (OTSD). In each example, the student receives only the original adversarial image together with the task prompt, while the teacher is conditioned on an additional privileged transformed observation. Among multiple transformed views, the view with the highest recognition accuracy selected by the judge LMM is provided to the teacher as the privileged observation.

Beyond the transformed view, the teacher prompt q_t incorporates additional privileged textual information, including adversarial OCR priors and the ground-truth OCR text. The adversarial OCR priors describe the construction process of synthetic adversarial patterns introduced in Section C.1, providing the teacher with contextual knowledge about the visual perturbations and helping it understand the underlying perception challenges. The ground-truth OCR text further provides explicit recognition guidance, improving the reliability of the teacher's generated responses.

Table 8: **Localization IOU** results of our AdvSpot benchmark across the expanded model zoo. **Category legend: 1** = Imaging Degradation (1-1: Capture Artifacts; 1-2: Post-processing Artifacts); **2** = Spatial Manipulation (2-1: Rotated Text; 2-2: Mirrored Text; 2-3: Tiny Text); **3** = Glyph Variations (3-1: Stylized Glyphs; 3-2: Handwritten Text); **4** = Visual Encoding (4-1: Symbol-based Encoding; 4-2: Dot-pattern Encoding; 4-3: Line-pattern Encoding); **5** = Contextual Blending (5-1: AIGC Fusion Text; 5-2: Low Contrast Text; 5-3: Pattern Overlay).

Model	**1**		**2**			**3**		**4**			**5**			**Overall**
	1-1	1-2	2-1	2-2	2-3	3-1	3-2	4-1	4-2	4-3	5-1	5-2	5-3	
OpenAI GPT Family														
GPT-5.1	10.5	18.8	11.5	13.8	3.8	40.6	37.2	44.7	50.0	42.9	28.1	7.5	38.7	27.8
GPT-5.5	39.7	42.5	34.7	42.4	18.7	62.9	59.6	78.7	78.8	68.4	33.1	37.5	69.2	51.5
GPT-5.6-sol	59.3	56.7	52.6	66.1	46.0	72.9	76.4	81.0	80.9	72.2	47.8	55.7	69.8	64.2
Google Gemini Family														
Gemini-3-Flash	45.0	56.5	43.5	45.7	39.6	66.7	54.0	69.5	77.5	80.7	52.4	37.5	91.0	59.2
Gemini-3.6-Flash	31.4	33.4	22.8	40.3	22.6	62.2	54.9	68.0	82.8	80.0	53.7	19.0	90.5	52.0
Anthropic Claude Family														
Claude-Haiku-4.5	3.7	7.2	1.1	6.0	6.0	22.2	6.3	16.5	21.2	24.5	16.1	6.3	22.0	12.8
Claude-Sonnet-4.5	4.8	9.7	10.3	12.1	0.0	32.3	19.8	3.2	4.0	11.7	5.8	6.0	12.9	10.4
Claude-Opus-4.5	4.0	8.1	4.1	7.8	4.7	12.5	13.7	16.3	19.6	26.0	16.7	5.1	25.1	13.1
Moonshot Family														
Kimi-K2.5	50.5	41.7	44.9	44.6	42.0	74.6	60.2	43.1	54.9	46.9	29.1	33.4	56.4	47.3
Kimi-K2.6	49.1	40.5	52.0	46.7	40.3	75.8	65.5	45.5	60.1	51.8	30.9	34.1	57.5	49.3
Kimi-K3	50.9	49.6	61.6	55.0	47.3	83.3	73.9	50.7	64.6	49.6	39.5	40.6	51.3	54.9
Zhipu Family														
GLM-4.6V	47.1	44.0	49.9	53.6	31.6	73.7	74.7	45.4	78.4	70.5	36.9	46.1	75.1	56.1
GLM-5.3-Flash	30.3	33.2	28.8	33.1	13.0	62.9	51.8	44.2	56.7	61.5	22.6	26.9	56.0	40.6
Alibaba Qwen Family														
Qwen3 Series														
Qwen3-VL-8B	59.1	43.3	68.0	65.7	45.9	76.2	71.2	28.7	65.5	43.5	38.1	52.4	62.3	54.4
Qwen3-VL-30B-A3B	61.0	45.9	69.0	73.6	46.8	73.1	71.4	39.8	60.4	56.4	21.3	47.6	62.0	55.0
Qwen3-VL-32B	70.5	55.0	69.7	72.0	60.8	79.6	76.8	50.8	63.1	53.2	42.5	49.8	66.8	61.1
Qwen3-VL-235B-A22B	59.7	46.8	71.7	68.3	48.9	74.6	75.7	50.2	61.4	53.7	38.3	54.2	67.6	58.6
Qwen3.5 Series														
Qwen3.5-35B-A3B	39.2	35.7	54.2	44.2	26.5	67.8	56.4	41.8	52.0	45.0	31.4	35.2	44.8	43.9
Qwen3.5-122B-A10B	50.2	43.7	51.6	51.0	44.6	61.2	57.9	25.7	48.1	44.0	19.7	49.2	58.3	45.8
Qwen3.5-397B-A17B	29.2	28.6	33.8	32.6	12.0	58.8	44.5	41.8	53.8	49.4	28.7	27.2	42.2	37.3
Qwen3.8 Series														
Qwen3.8-27B	57.1	40.6	59.2	66.3	40.6	73.7	56.9	43.7	64.6	50.8	22.2	42.9	60.4	51.2
Our Proposed Method														
ArmorOCR	58.9	40.3	70.3	60.9	41.2	70.1	61.0	72.4	73.0	77.0	61.5	41.0	89.1	63.2

Table 9: **VQA Accuracy** results of our AdvSpot benchmark across the expanded model zoo.

Model	1		2			3		4			5			Overall
	1-1	1-2	2-1	2-2	2-3	3-1	3-2	4-1	4-2	4-3	5-1	5-2	5-3	
OpenAI GPT Family														
GPT-5.1	26.7	43.3	28.6	13.3	45.7	30.0	30.0	31.4	33.3	11.4	5.7	28.6	13.3	26.5
GPT-5.5	26.7	40.0	28.6	20.0	51.4	50.0	46.7	2.9	10.0	2.9	2.9	37.1	6.7	24.9
GPT-5.6-sol	23.3	33.3	40.0	20.0	51.4	36.7	40.0	2.9	10.0	2.9	0.0	37.1	3.3	22.9
Google Gemini Family														
Gemini-3-Flash	66.7	80.0	62.9	13.3	68.6	56.7	56.7	20.0	33.3	11.4	17.1	74.3	36.7	45.2
Gemini-3.6-Flash	73.3	80.0	65.7	23.3	77.1	63.3	66.7	8.6	40.0	17.1	8.6	74.3	33.3	47.6
Anthropic Claude Family														
Claude-Haiku-4.5	3.3	16.7	3.3	0.0	26.7	6.7	3.3	0.0	0.0	0.0	0.0	8.6	0.0	5.0
Claude-Sonnet-4.5	3.3	26.7	6.7	6.7	50.0	13.3	13.3	0.0	3.3	0.0	0.0	11.4	0.0	9.8
Claude-Opus-4.5	33.3	50.0	30.0	13.3	60.0	30.0	26.7	0.0	3.3	3.3	5.0	31.4	2.9	21.2
Moonshot Family														
Kimi-K2.5	50.0	63.3	45.7	13.3	60.0	76.7	56.7	8.6	13.3	2.9	8.6	54.3	10.0	34.4
Kimi-K2.6	53.3	60.0	45.7	20.0	62.9	73.3	70.0	11.4	13.3	0.0	5.7	54.3	16.7	37.0
Kimi-K3	53.3	56.7	51.4	30.0	68.6	70.0	60.0	2.9	13.3	2.9	5.7	48.6	3.3	35.0
Zhipu Family														
GLM-4.6V	50.0	36.7	31.4	23.3	51.4	36.7	36.7	5.7	23.3	5.7	14.3	28.6	33.3	30.3
GLM-5.3-Flash	50.0	40.0	48.6	43.3	45.7	60.0	53.3	5.7	26.7	0.0	11.4	45.7	36.7	35.2
Alibaba Qwen Family														
Qwen3 Series														
Qwen3-VL-8B	40.0	53.3	48.6	20.0	60.0	30.0	36.7	0.0	23.3	2.9	5.7	45.7	26.7	29.8
Qwen3-VL-30B-A3B	53.3	53.3	48.6	26.7	54.3	50.0	53.3	0.0	23.3	5.7	8.6	45.7	30.0	34.7
Qwen3-VL-32B	53.3	50.0	54.3	20.0	51.4	46.7	40.0	0.0	20.0	5.7	5.7	48.6	23.3	31.9
Qwen3-VL-235B-A22B	63.3	60.0	60.0	23.3	57.1	53.3	53.3	5.7	23.3	14.3	11.4	48.6	26.7	38.3
Qwen3.5 Series														
Qwen3.5-35B-A3B	43.3	53.3	48.6	20.0	40.0	43.3	50.0	8.6	20.0	0.0	8.6	34.3	26.7	30.1
Qwen3.5-122B-A10B	23.3	33.3	34.3	10.0	48.6	46.7	26.7	0.0	10.0	0.0	2.9	34.3	23.3	22.1
Qwen3.5-397B-A17B	33.3	36.7	28.6	6.7	31.4	46.7	20.0	2.9	16.7	0.0	0.0	25.7	13.3	19.5
Qwen3.8 Series														
Qwen3.8-27B	46.7	43.3	42.9	13.3	48.6	36.7	26.7	0.0	16.7	8.6	0.0	37.1	16.7	25.4
Our Proposed Method														
ArmorOCR	43.3	60.0	54.3	46.7	57.1	26.7	46.7	68.6	63.3	54.3	82.9	42.9	66.7	57.1

Example 1: Student and Teacher Prompts for Stage 1

Student Prompt:



```
<image>
Recognize the text in the image.
Put your reasoning process inside <analyze></analyze>,
and put the recognized text inside <answer></answer>.
```

Teacher Prompt (Resize Transformation):




```
<image>
Recognize the text in the image.
Put your reasoning process inside <analyze></analyze>,
and put the recognized text inside <answer></answer>.

Reference Information:
The text is generated with a 164px large bold stroke style,
rotated counterclockwise by approximately 8 degrees,
and blended with an AI-generated natural scene background.
The target text is "VR".

Now generate your own reasoning process and answer.
```

Figure 8: Example 1 of student and teacher prompts in Stage 1 observation-transferred self-distillation. The student receives only the original image, while the teacher receives additional privileged information revealed by transformed observations.

Example 2: Student and Teacher Prompts for Stage 1

Student Prompt:



```
<image>
What is the text written on the semi-transparent white uppercase
letters
overlaid on the roasted duck?

Put your reasoning process inside <analyze></analyze>,
and put the recognized text inside <answer></answer>.
```

Teacher Prompt (Rotation Transformation):




```
<image>
What is the text written on the semi-transparent white uppercase
letters
overlaid on the roasted duck?

Put your reasoning process inside <analyze></analyze>,
and put the recognized text inside <answer></answer>.

Reference Information:
The image has been processed with an inverse rotation
transformation.
The target answer is "CHARME".

Now generate your own reasoning process and answer.
```

Figure 9: Example 2 of student and teacher prompts in Stage 1 observation-transferred self-distillation.

B.2 More Details of Stage 2

Stage 2 employs GRPO with four task-conditioned objectives to explicitly optimize different aspects of grounded adversarial OCR perception. Specifically, we formulate four tasks, including text-to-bbox localization, bbox-to-text recognition, full spotting, and grounded VQA. Each task uses a task-specific prompt and corresponding reward function. The detailed prompts are shown in Figure 10.

Task-specific Prompts for Stage 2 GRPO Optimization

1. Text-to-BBox Localization

```
<image>
## Task Description
Locate the position of the following text in the image:
"{text content}".

## Output Requirement
Output the bounding box directly in the format:
[x_min, y_min, x_max, y_max].
The coordinates are normalized integers within the range of 0--1000.
```

2. BBox-to-Text Recognition

```
<image>
## Task Description
Recognize the text within the following bounding box region:
[{x_min}, {y_min}, {x_max}, {y_max}].
The bounding box format is [x_min, y_min, x_max, y_max],
where coordinates are normalized integers within 0--1000.
```

3. Full Spotting

```
<image>
## Task Description
Detect and recognize all text content in the image.

## Output Requirement
Output in JSON format:
[ {"bbox":  [x_min, y_min, x_max, y_max],
  "text":  "recognized text"} ]
```

4. Grounded VQA

```
<image>
## Task Description
Answer the following grounded VQA question:
{grounded VQA question}.

## Output Requirement
bbox:[x_min, y_min, x_max, y_max]
answer:[text content]
```

Figure 10: Task-specific prompts used for Stage 2 GRPO optimization.

C More Details of Training

C.1 Training Data Construction

Due to MLLM's limited perception of adversarial OCR, building such datasets relies heavily on manual annotation, which is difficult to scale. Therefore we design an adversarial OCR data syn-

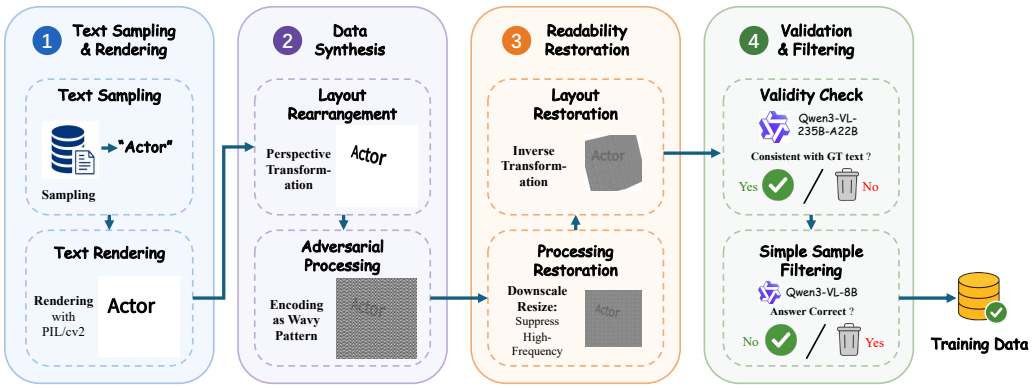


Figure 11: Adversarial OCR Data Synthesis Framework.

thesis framework that batch-generates adversarial OCR samples with full annotations (including adversarial prior, text transcription, and bounding box).

The overall pipeline is shown in 11, which consists of the following steps: text generation, rendering, adversarial processing, readability restoration, validation and filtering. Text sources come from LLM-generated corpora and *Faker* library[2]. In the rendering step, each text instance is drawn on a separate mini-canvas with its bounding box recorded. All instances are blended into a single image after layout rearrangement and bounding bbox based collision detection. Then a range of adversarial processing steps (such as AIGC fusion, ASCII art) are applied to the template image, making the text hard to recognize. To ensure data usability, we introduce a readability-restoration step that partially recovers the readability of the adversarial OCR sample. The resulting degraded sample is verified by Qwen3-VL-235B-A22B, which rejects samples where adversarial processing destroyed the original text. Finally, Qwen-VL-8B filters out samples that can already be correctly recognized without further training.

C.2 Training Configurations

Table 11 summarizes the detailed training configurations used in Stage 1 and Stage 2 of ArmorOCR, including the common model settings and stage-specific hyper-parameters for observation-transferred self-distillation and Reward-driven Refinement.

D More Details of Inference

We report the mean end-to-end inference latency per question on the 389 grounded VQA pairs of AdvSpot (Table 10). We compare ArmorOCR with Smuggle-CoT, a prompting-based chain-of-thought baseline, and Thyme (Zhang et al., 2025), a publicly available thinking-with-images method that executes image-processing operations during inference. We could not measure VACoT under the same setting because no public inference implementation was available at the time of evaluation; we therefore use Thyme as a reproducible tool-augmented visual reasoning baseline. ArmorOCR achieves the lowest measured inference latency, as it requires neither extended chain-of-thought prompting nor image-transformation tool calls at inference time.

[2]https://github.com/joke2k/faker

Table 10: Mean end-to-end inference latency on the 389 grounded VQA pairs of AdvSpot.

Method	**Mean latency (s/question)**
Smuggle-CoT	74.48
Thyme	47.31
ArmorOCR (ours)	0.81

Table 11: Training configurations of ArmorOCR for two stages.

Training Configuration	**Value**	**Description**
Common Configuration		
Model Configuration		
Backbone	*Qwen3-VL-8B-Instruct*	/
FREEZE_LLM	*False*	All LLM parameters are trainable
FREEZE_ALIGNER	*False*	All aligner parameters are trainable
FREEZE_VIT	*False*	All vision encoder parameters are trainable
Optimization Configuration		
Optimizer	AdamW	/
Learning Rate	1×10^{-6}	/
Weight Decay	0.05	/
Warmup Ratio	0.03	/
Precision	*bfloat16*	/
DeepSpeed	*ZeRO-2*	/
Vision Processing Configuration		
Minimum Image Tokens	64	/
Maximum Image Tokens	4096	/
Maximum Sequence Length	8192	/
Stage 1: Observation-Transferred Self-Distillation		
OPSD Configuration		
JSD On-policy Sampling Ratio λ	1.0	Fully on-policy
Generalized JSD Weight β	0.5	Weight in the generalized JSD loss
Sampling Temperature	1.0	/
Maximum Completion Length	1024	Maximum number of generated tokens
vLLM Configuration		
Use vLLM	*True*	/
vLLM Mode	*Colocate*	vLLM shares the same devices with training
vLLM Maximum Model Length	8192	/
vLLM GPU Memory Utilization	0.4	/
Training Configuration		
Training Samples	50K	/
Accelerators	16	PPU-810E accelerators
Per-device Batch Size	1	/
Gradient Accumulation Steps	2	/
Training Epochs	2	/
Stage 2: Reward-driven Refinement		
GRPO Configuration		
Number of Rollouts	8	Number of sampled responses per prompt
Steps per Generation	4	Policy update steps per generation
KL Coefficient β	0.04	KL regularization coefficient
Maximum Completion Length	1024	/
vLLM Configuration		
Use vLLM	*True*	/
vLLM Mode	*Server*	External vLLM server for rollout generation
vLLM Maximum Model Length	8192	/
vLLM Tensor Parallel Size	1	/
vLLM Data Parallel Size	16	Computed from 16 rollout accelerators
vLLM Prefix Caching	*True*	Enabled during rollout generation
Asynchronous Generation	*True*	/
Training Configuration		
Training Samples	70K	/
Accelerators	128	PPU-810E accelerators
Training Accelerators	112	14 per node $\times$ 8 nodes
Rollout Accelerators	16	2 per node $\times$ 8 nodes
Number of Nodes	8	/
Processes per Node	14	Training processes only
Per-device Batch Size	4	/
Gradient Accumulation Steps	1	/
Training Epochs	3	/

E CASE STUDY



Question: Please identify the hidden text in the image.

Grouned Truth: WPK俱乐部

Our ArmorOCR

<analyze>
The image features stylized text that appears to be "WPK俱乐部".
The text is rendered in a glowing, slightly blurred font, superimposed on an AI-generated background depicting a night scene with bamboo stalks and a moon. Th
e characters are arranged in three lines: "WP" on the top, "K俱乐部"
on the middle and bottom. Although the image is designed to blend the text into t
he environment, the text is still legible and appears to be a coherent phrase. The
phrase "WPK俱乐部" is likely a proper name, perhaps for a club or organization.
</analyze>
<answer>WPK俱乐部</answer> ✅

GPT-5

<analyze>
I examined the image carefully for any legible or hidden text and did not find any.
</analyze>
<answer>No recognizable text found in the image.</answer> ❌

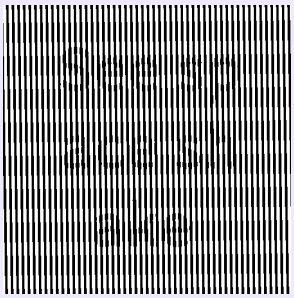
Gemini-2.5-Flash

<analyze>
I will analyze the provided image and its crops to identify any hidden text.
1. Examine the overall image: The image depicts a serene night scene with bamboo forests, a river, lanterns, and a moonlit sky in an illustrative or digital-art style.
2. Look for anomalies or patterns: I scan the entire image, paying attention to areas that might contain hidden text.
3. Utilize the cropped images: The crops help zoom in on specific sections without losing context, so I go through them systematically.
Crop 1 (Top left): focuses on the upper sky, moons, and bamboo. No obvious text.
Crop 2 (Top right): focuses on the upper sky, moons, bamboo, and a hanging lantern. No obvious text.
Crop 3 (Bottom left): focuses on the left bank of the river, bamboo, and ground lanterns. Looking closely at the reflections of the three lanterns near the bottom left corner, each reflection shows faint, distorted shapes. It is very subtle, but these reflections seem to contain letter-like forms.
Crop 4 (Bottom right): focuses on the right bank of the river, bamboo, and ground lanterns. No obvious text.
4. Re-examine the original image with this location in mind: I focus on the reflections of the three lanterns on the left bank, closest to the viewer. The left reflection looks like a distorted "A" or "R". The middle reflection looks like a distorted "I" or "L". The right reflection looks like a distorted "A" or "N". Although the text is extremely faint and distorted by the water reflection, the most plausible reading from left to right is "AIA".
</analyze>
<answer>The hidden text is "AIA". </answer> ❌

Qwen3-VL-235B-A22B-Instruct

<analyze>
Upon careful inspection of the image, I looked for any text subtly embedded in the lighting, reflections, foliage, lanterns, sky, or water. The image shows a serene night scene with bamboo stalks, glowing lanterns, a moonlit sky, a reflective river, and distant misty mountains. I examined the lanterns, bamboo, sky, water, and background architecture, but found no visible inscriptions, mirrored text, or hidden symbols. I also checked for steganographic or watermark-like elements, but nothing appears embedded in the composition. Therefore, I conclude that no text is hidden in this image.
</analyze>
<answer>
No hidden text found.
</answer> ❌

Figure 12: Case study 1 from AdvSpot (AIGC Fusion Text), illustrating the performance comparison between ArmorOCR and existing LMMs on adversarial OCR perception.



Question: What is the faintly visible text in the central region of the pattern formed by dense vertical black lines?

Grouned Truth: See space Shake

Our ArmorOCR

<analyze>
1. The image displays a pattern of dense vertical black lines on a light bac kground.
2. These lines are not uniform; they vary in spacing and density, creating a steganographic effect where text is hidden within the pattern.
3. By closely examining the pattern, one can discern the faint outlines of letters formed by the spacing and alignment of the lines.
4. The text appears to be arranged in three lines:
  - The first line clearly shows "See sp"
  - The second line shows "ace Sh"
  - The third line shows "ake"
5. Combining these lines gives the phrase "See space Shake".
6. Although the text is faint, the letterforms are recognizable as "See s pace Shake".
</analyze>
<answer>See space Shake</answer> ✅

GPT-5

<analyze>
The image shows a moiré-like pattern with dense vertical black lines. In the lighter central region, three lines of faint text appear. The top line reads “Open up,” the middle line “about,” and the bottom line “AI.” The letter shapes match these words when focusing past the lines.
</analyze>
<answer>Open up about AI</answer> ❌

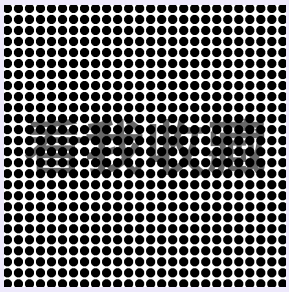
Gemini-2.5-Flash

<analyze>
The image is an optical illusion with text hidden within dense vertical black lines. To identify it, I look for subtle variations in the lines' thickness, spacing, or alignment in the central region. Slightly defocusing or viewing from a short distance helps the hidden pattern emerge.
1. Initial scan: I focus on the central area where the lines deviate from a uniform pattern.
2. Pattern recognition: Some lines appear slightly thicker or shifted, forming faint letter shapes.
3. Letter identification: On the left, the shape resembles an “H”. Next is an “I”. Then a “D”. Followed by another “D”. Then an “E”. On the far right, an "N".
4. Consolidation: Together, these letters form the word "HIDDEN".
5. Verification with crops: Crop 1 (top left) shows the “H” and the start of the “I”. Crop 2 (top right) shows the “I” and the first “D”. Crop 3 (bottom left) shows the second “D” and the “E”. Crop 4 (bottom right) shows the “E” and the “N”. The crops support the same reading, so the hidden text is "HIDDEN".
</analyze>
<answer> The faintly visible text in the central region of the pattern is “HIDDEN”. </answer> ❌

Qwen3-VL-235B-A22B-Instruct

<analyze>
The image consists of dense vertical black lines on a white background, creating a pattern that resembles a barcode or moiré effect. At first glance, it appears uniform, but closer inspection—especially with a slight change in viewing angle, brightness, or by squinting—reveals that certain vertical lines are slightly broken, offset, or spaced differently to encode hidden text.
This technique is known as steganography or a “secret message in plain sight” using visual patterns. The interruptions in the vertical lines form discernible shapes aligned horizontally, which spell out letters.
By mentally grouping the gaps or disruptions in the vertical lines into horizontal segments, it becomes possible to decode the text: In the upper-central area, the pattern of gaps forms the word “HELLO”. Below it, another line of gaps forms “WORLD”. Together, the faint visible text reads “HELLO WORLD”.
This is a common placeholder or test message in programming and visual puzzles.
</analyze>
<answer>Hello World</answer> ❌

Figure 13: Case study 2 from AdvSpot (Line-pattern Encoding).



Question: What text appears beneath the dot pattern in gray?

Grouned Truth: 看我收藏

Our ArmorOCR

<analyze >
The image displays a background of a black and white dot grid, which is a common pattern for steganography or visual effects. On top of this pattern, there is a text overlay. The text is in gray and appears to be in Chinese characters. The characters are clearly legible and read as "看我收藏
". This phrase translates to "Look at my collection" or "Check out my collection".
Therefore, the text that appears beneath the dot pattern is "看我收藏".
</analyze>
<answer>看我收藏</answer>

GPT-5

<analyze>
By squinting and mentally defocusing the regular dot pattern, the darker gray regions form four blocky letters. The last clearly resembles an E. The second letter has the triangular shape of an A, and the third shows the diagonal arms of a K. The first letter appears as a vertical with horizontal bars, consistent with F. Put together, the hidden word reads FAKE.
</analyze>
<answer>FAKE</answer>

Gemini-2.5-Flash

<analyze>
The user asks me to identify the gray text hidden beneath the dot pattern. I examine both the main image and the provided crops, since the crops help isolate different parts of the word.

Looking at the crops:
1. The first crop (left side) shows the left part, where I can make out "Tech".
2. The second crop (middle-left) confirms "Tech" and begins to show the next letter, "b".
3. The third crop (middle-right) shows "blog".
4. The fourth crop (right side) confirms the "blog" part.

Combining these parts, the word appears to be "Techblog".

I then verify this with the full image. The gray text is semi-transparent and overlaid on black dots on a white background. Reading from left to right, the letters are "T", "e", "c", "h", "b", "l", "o", and "g", which together form "Techblog".
</analyze>
<answer>Techblog</answer>

Qwen3-VL-235B-A22B-Instruct

<analyze>
The image shows a pattern of black dots on a white background, with some areas where the dots are missing, revealing gray text underneath. By examining the gaps in the dot pattern, we can recognize characters.
Looking at the central area:
- The first visible character has the shape of an 'R' (rounded top, diagonal leg).
- Following that, there's a character resembling 'O', then 'B', 'O', 'T', and finally a 'S'.
These letters together form the word "ROBOTS".
This is consistent with the style of a "dot matrix" or "stippled" reveal, commonly used in puzzles or hidden messages where removing or offsetting dots exposes underlying text.
Thus, the gray text revealed beneath the dots is "ROBOTS".
</analyze>
<answer>ROBOTS</answer>

Figure 14: Case study 3 from AdvSpot (Patter Overlay).



Question: What is the white text above the QR code?

Grounded Truth: 分享密码给朋友

Our ArmorOCR

<analyze>
The image shows a QR code with white text positioned above it. The text is clearly legible and reads "分享密码给朋友", which translates to "Share password with friends" in English. This is a common phrase used in applications or services that require sharing access credentials with others. The text is written in a standard Chinese font, and its placement above the QR code suggests it is an instruction or description related to the QR code's purpose.
</analyze>
<answer>分享密码给朋友</answer> ✅

GPT-5

<analyze>The white text above the QR code is short Chinese UI text commonly seen on login pages. From visual inspection, it reads "支持扫码登录." Confidence is moderate due to the image's low resolution..analyze>
<answer>支持扫码登录</answer> ❌

Gemini-2.5-Flash

<analyze>The user wants to identify the white text above the QR code. I will examine the image, focusing on the white characters, and transcribe them one by one from left to right.
1. **First character:** Appears to be 艾 (ài).
2. **Second character:** Appears to be 朗 (lǎng).
3. **Third character:** Appears to be 酷 (kù).
4. **Fourth character:** Appears to be 密 (mì).
5. **Fifth character:** Appears to be 毫 (háo).
6. **Sixth character:** Appears to be 贷 (dài).
Combining these characters gives "艾朗酷密毫贷".</analyze>
<answer>艾朗酷密毫贷</answer> ❌

Qwen3-VL-235B-A22B-Instruct

<analyze>The image shows a QR code with some white text above it on a dark background. Although the image is somewhat blurry and grainy, the characters are still legible upon close inspection. The text appears to be in Traditional Chinese. Reading from left to right:
- The first character looks like "支" (zhī).
- The second character resembles "付" (fù).
- The third character is "金" (jīn).
- The fourth character is "碼" (mǎ).
- The fifth character is "尊" (zūn).
- The sixth character is "享" (xiǎng).
Together, these characters form the phrase "支付金碼尊享" — which may be part of a promotional message, possibly meaning "Enjoy exclusive privileges with payment code" or similar. This phrase is commonly seen in contexts involving QR code payments in Chinese-speaking regions.
Note: In some cases, "支" and "付" could be part of "支付" (payment), and "金碼" could be a brand or product name ("golden code"), followed by "尊享" meaning "exclusive enjoyment" or "premium experience."
Therefore, the white text above the QR code is: 支付金碼尊享.</analyze>
<answer>支付金碼尊享</answer> ❌

Figure 15: Case study 4 from AdvSpot (Mirrored Text).

F DISCUSSION

AdvSpot and ArmorOCR address two connected challenges: diagnosing adversarial OCR failures at the region level and transferring information revealed by transformed views into a model that operates on the original image. The following discussion connects these contributions to the reported evidence and clarifies the scope of the conclusions.

Benchmark contribution and annotation quality. AdvSpot provides bounding boxes, transcriptions, perception-type labels, and region-specific questions across 5 categories and 13 fine-grained types (Table 1). These annotations support more detailed diagnosis than image-level recognition scores alone. The construction pipeline includes independent annotation by two annotators, consensus filtering, and two rounds of expert review of generated QA pairs (Section 3.3). These steps target transcription errors, ambiguous region references, and QA inconsistency. The benchmark contains 383 images and 389 grounded VQA pairs; Table 7 reports the number of annotated adversarial patterns in each fine-grained subtype. A single image or VQA pair may contain patterns from multiple subtypes. Together, the taxonomy and region-level annotations support subtype-specific analysis of recognition and localization failures, providing a structured basis for diagnosing adversarial OCR behavior.

Meaning of grounded evaluation. Each AdvSpot question refers to an annotated region–text pair, and the evaluation prompt requests both an answer and its corresponding bounding box (Appendix A.3). For each instance, the predicted answer and bounding box are evaluated against the same reference pair using accuracy and IoU, respectively (Equation 2). This enables sample-level analysis of whether a correct answer is accompanied by accurate localization, as well as identification of cases where recognition and localization outcomes diverge. Reporting accuracy and IoU separately further provides complementary summaries of these capabilities, supporting more informative diagnosis than answer-only evaluation.

Method contribution and inherited components. ArmorOCR builds on OPSD and GRPO as established training frameworks, with specific designs for adversarial OCR (Section 4). In Stage 1, we construct multiple candidate views through resizing, stretching, rotation, flipping, and compression, and use a judge LMM to select the view with the highest recognition accuracy for the teacher. The student receives the original image and learns from the teacher's token-level guidance. Response-region-aware distillation further distinguishes spans of the generated response, applying confidence-gated JSD to analysis tokens, forward KL to answer tokens, and excluding non-content structural tokens. Table 5 evaluates the contributions of visual transfer and this distillation objective. In Stage 2, task-conditioned GRPO rewards directly optimize localization, recognition, joint region–text matching, and region-specific question answering through text-to-bbox, bbox-to-text, full spotting, and grounded VQA tasks, respectively. Table 6 evaluates the contribution of each reward. Together, these designs connect supervision from selected transformed views with explicit optimization of grounded OCR capabilities.

Privileged information during training. The Stage 1 teacher receives a selected transformed view, adversarial OCR priors, and the target OCR text, while the student generates its trajectory from the original image and task prompt (Appendix B.1). The teacher evaluates that trajectory to provide token-level guidance. These additional inputs constitute training-time supervision; inference uses the trained student without the privileged teacher inputs or the view-selection judge. Table 5 ablates visual transfer and the response-region-aware distillation objective separately. Removing visual transfer retains both privileged textual information and the response-region-aware objective; replacing this objective with standard JSD retains both transformed observations and privileged textual information. Together, these comparisons support the contributions of transformed visual supervision and the proposed distillation objective within OTSD.

What the training comparisons establish. Table 4 compares four training methods using the same 50K Stage 1 examples. SFT uses the ground-truth OCR text as the target for teacher-forced supervision. Standard OPSD conditions the teacher on the original image and privileged textual information, without visual transformations, and applies the standard JSD distillation loss. GRPO uses an accuracy reward computed against the ground-truth OCR text. OTSD's advantage over these

alternatives supports the effectiveness of its Stage 1 training framework in developing adversarial OCR perception beyond exposure to synthetic data alone.

Experiment Performance. On AdvSpot, ArmorOCR improves accuracy from 29.8% to 57.1% and average IoU from 54.4% to 63.2% over its base model (Table 2). Although ArmorOCR does not outperform proprietary LMMs on every subtype, the Qwen3-VL-8B-based model achieves substantial gains on challenging visual encoding and contextual blending patterns. In particular, it reaches 68.6%, 63.3%, 54.3%, and 82.9% accuracy on Symbol Encoding, Dot Encoding, Line Encoding, and AIGC Fusion Text, respectively, exceeding all competing models reported in Table 9 on these subtypes. Table 3 further demonstrates ArmorOCR's applicability beyond AdvSpot, with the highest average scores among the compared methods on both AdvOCR and SmuggleBench. Although the overall margins are modest, ArmorOCR achieves substantial gains on specific adversarial subsets, outperforming VACoT by 20% on the AdvOCR synthetic split and Smuggle-CoT by 8.2% on the SmuggleBench AI Illusions category. These results highlight its effectiveness on particularly challenging adversarial OCR patterns.

Practical inference implications. ArmorOCR moves view transformation and selection into training. At inference time, the student processes the original image without an additional transformation pipeline or external tool calls. Appendix D reports mean end-to-end inference latency on AdvSpot: ArmorOCR achieves the lowest inference latency among the evaluated methods, supporting the practical efficiency of internalizing transformed-view supervision during training.