# VIESCORE2: UNIFIED IMAGE EVALUATION WITH SPATIALLY GROUNDED EXPLANATIONS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Existing synthetic image evaluators typically provide only a scalar quality score and do not identify the image regions that support it. We introduce **VIEScore2**, a unified evaluator for image generation and editing tasks with optional conditioning images. VIEScore2 represents an image as an $N \times N$ grid and jointly predicts quality scores and defect locations in a single model pass. Its text-native grid representation provides a common interface for heterogeneous spatial supervision and enables directly verifiable post-training objectives. We train on 38K examples spanning score-only, localization-only, and joint supervision across generation and editing tasks. Starting from supervised fine-tuning, we further apply GRPO to improve defect localization using rewards that combine cell-level Dice overlap, score accuracy, and output-format validity. A parameter-free parser converts the structured predictions into readable explanations. On the primary suite, VIEScore2 achieves an overall-score SRCC of 0.601, compared with 0.491 for Gemini-3-Flash, the strongest zero-shot general-purpose VLM baseline under matched inputs. For defect localization, VIEScore2 outperforms both general-purpose VLMs and specialized spatial evaluators on three of six benchmarks in per-image grid IoU and ranks among the top three on five, including datasets beyond its training sources.

## 1 INTRODUCTION

Recent image generation and editing models have achieved increasingly realistic visual synthesis (Rombach et al., 2022; Saharia et al., 2022). Despite this remarkable progress, generated images still frequently contain localized failures, such as implausible structures, object distortions, visual artifacts, or violations of the input prompt (Liang et al., 2024; Zhang et al., 2023). Manually identifying these defects is costly, inefficient, and difficult to scale (Ku et al., 2024b). Consequently, reliable automatic evaluators have become increasingly important. Yet, what should an evaluator produce beyond a single quality score? Humans can often recognize image quality intuitively, but automatic evaluators still struggle to explain their scores and identify defective regions. Most learned evaluators return a scalar score (Hessel et al., 2021; Xu et al., 2023; Fu et al., 2023). VIEScore (Ku et al., 2024a) prompts a frozen vision-language model (VLM) to produce scores and free-form explanations, but these explanations are not necessarily grounded in image regions. Spatial methods instead predict boxes, heatmaps, or masks (Zhang et al., 2026; Guo et al., 2026; Zhang et al., 2023), but many focus on text-to-image generation. Extending these methods to image editing and generation tasks that require conditioning images, quality scores and defect categories remains challenging.

We introduce VIEScore2, a unified evaluator for diverse image generation and editing tasks. It takes a generated image, its prompt, optional conditioning images, and an evaluation instruction that specifies the requested output fields. VIEScore2 supports score-only, localization-only, and joint evaluation with the same model. It represents defects as sparse cells on an $N \times N$ grid and can separately localize two defect categories: visual artifacts and semantic misalignments. The predicted scores, defect categories, and cell locations are further converted into readable explanations that summarize the scores and spatial evidence. Figure 1 provides a complete evaluation example. Methodologically, we first apply supervised fine-tuning (SFT) to train a vision-language model (VLM) to predict the quality scores and defect grid. However, its token-level objective does not directly optimize the overlap between predicted and ground-truth cells. We therefore post-train the model using group relative policy optimization (GRPO) (Shao et al., 2024). For each input, GRPO samples multiple



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.

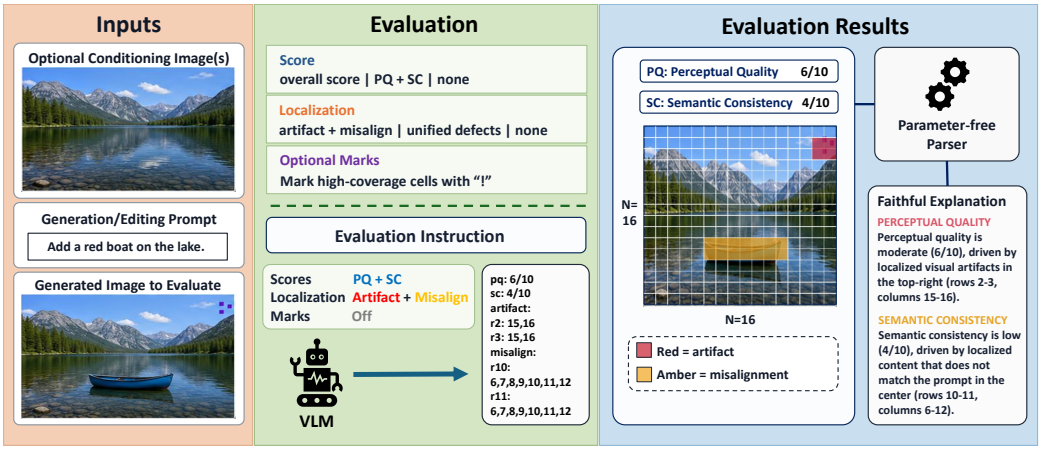



Figure 1: An evaluation example with VIEScore2. The model predicts perceptual quality (PQ), semantic consistency (SC), and a $16 \times 16$ defect grid, where red and amber denote visual artifacts and semantic misalignments. A parameter-free parser converts the predictions into natural language.

responses and reinforces those with higher rewards. Our reward combines cell-level Dice overlap with ground-truth defects (Milletari et al., 2016), score accuracy, and output format validity. We unify data from five sources to train a single evaluator across image generation and editing tasks. The training data include both quality scores and defect locations, although individual examples may provide only one type of supervision. Section 3.3 details the fine-tuning process. Figure 2 summarizes the data preparation, training, and inference pipeline of VIEScore2. Our contributions are threefold:

1. **Unified scoring, localization, and explanation.** We introduce a single evaluator for image generation and editing tasks with zero, one, or multiple conditioning images. Given a generated image, its prompt, optional conditioning images, and an evaluation instruction, VIEScore2 produces quality scores and defect locations without task-specific prediction heads. A parameter-free parser converts these predictions into readable, spatially grounded feedback.
2. **Verifiable spatial post-training.** Defects are represented as sparse cells on an image grid. This representation is directly parseable and supports a coverage-weighted cell-level Dice reward that directly rewards localization overlap.
3. **Learning and evaluation across heterogeneous sources.** We normalize the scores and spatial annotations from five data sources into a common training format while retaining only the supervision available for each example. Across generation and editing tasks, VIEScore2 achieves higher aggregate overall-score correlation than all four general-purpose VLM baselines under the same input setting, alongside competitive defect localization.

## 2 Related Work

**Vision-Language Models as Image Evaluators.** Recent works have explored using VLMs as automatic image evaluators. Early methods such as TIFA (Hu et al., 2023) and DSG (Cho et al., 2024) decompose prompts into structured questions to measure text–image alignment. VQAScore (Lin et al., 2024) uses the probability of an affirmative VQA response as an alignment score, while Q-Align (Wu et al., 2024) teaches a VLM to predict text-defined quality levels. EvalMuse-40K (Han et al., 2026) introduces FGA-BLIP2, an evaluator that predicts overall and fine-grained text–image alignment scores. VIEScore (Ku et al., 2024a) and its agentic extension CIGEval (Wang et al., 2025a) further demonstrate that general-purpose VLMs can provide holistic evaluations across multiple conditional image synthesis tasks. These methods primarily return scalar or textual judgments and do not explicitly ground their evaluations in defective image regions. In contrast, VIEScore2 jointly predicts quality scores and spatial defect locations within the same VLM, while supporting generation and editing tasks with zero, one, or multiple conditioning images.

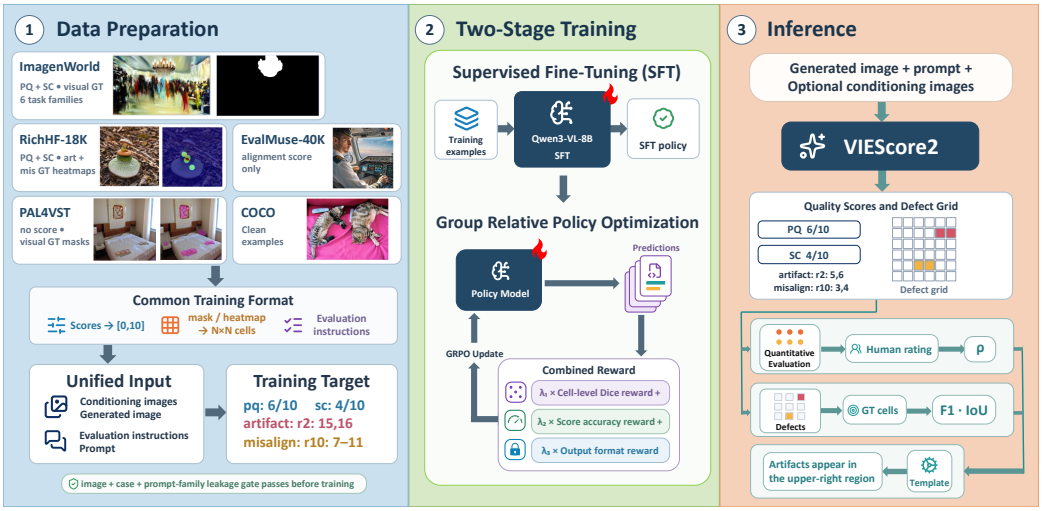



Figure 2: Overview of VIEScore2: data preparation, SFT and GRPO training, and inference. GRPO combines Dice, score, and format rewards. The parameter-free parser converts predicted scores and defect grids into explanations. GT denotes ground truth.

**Visual Explanation through Defect Localization.** Spatial evaluators identify defective image regions. PAL (Zhang et al., 2023) predicts pixel-level artifact masks, while RAHF (Liang et al., 2024) predicts quality scores and separate artifact and misalignment heatmaps. LEGION (Kang et al., 2025) combines synthetic-image detection, artifact segmentation, and explanation. HEIE (Yang et al., 2025) combines implausibility scores, heatmaps, and explanations. These approaches provide spatial feedback, but typically use dedicated spatial outputs such as masks or heatmaps and are mainly designed for text-to-image evaluation. VIEScore2 instead represents defects as sparse grid cells generated directly as text, enabling a single model to produce scores and spatial evidence without task-specific prediction heads. It also supports conditional generation and editing with optional conditioning images, including multiple references. A parameter-free parser converts the structured predictions into readable descriptions without additional training. Appendix C.4 compares their output interfaces.

**Reinforcement Learning for Image Evaluation.** Reinforcement learning has been used to improve image-quality scoring and reasoning (Li et al., 2025; Wu et al., 2025; Wang et al., 2025b) and, with IoU-based verifiable rewards, visual grounding (Liu et al., 2025). ImageDoctor (Guo et al., 2026) is the closest to our work and uses grounding, score, and heatmap rewards for text-to-image evaluation. SDG (Zhang et al., 2026) uses rewards for box localization, description consistency, and importance estimation. These methods demonstrate that verifiable spatial supervision can improve image evaluation, but rely on heatmap- or box-based grounding objectives. VIEScore2 instead optimizes the same sparse textual grid representation used at inference through a cell-level Dice reward, together with score and format rewards. This directly aligns post-training with the model's structured output while jointly optimizing quality scoring and defect localization.

# 3 Method

## 3.1 Evaluation Inputs and Outputs

The input $x_i$ contains a generated image, a generation or editing prompt, and zero, one, or multiple conditioning images. Conditioning images can be source images for editing or reference images for generation. They precede the generated image in the model input. An evaluation instruction $\tau_i$ specifies the requested output fields: **1. scores**: an overall score or separate scores for perceptual quality (PQ) and semantic consistency (SC), all on a 0–10 scale; **2. localization**: defect cells on an $N \times N$ grid, either in a single grid or in separate grids for visual artifacts and semantic misalignments; and **3. coverage marks**: optional ! marks on defect cells, as defined in Section 3.2. The instruction can request scores, localization, or both, and specifies whether to include coverage marks. During

training, the requested fields match the available supervision. VIEScore2 generates structured predictions in a single autoregressive pass. The scores and cell coordinates can be parsed directly for reward computation and evaluation. The same VLM supports these output configurations without task-specific prediction heads or additional fine-tuning for each configuration.

### 3.2 Sparse Grid Representation

We divide the generated image into an $N \times N$ grid and represent defects as a set of cell coordinates $G \subseteq \{1, \ldots, N\} \times \{1, \ldots, N\}$. We encode these coordinates as sparse text sequences for autoregressive generation (Lan et al., 2025). This format accommodates disconnected defect regions and separate defect categories. An empty set $G = \varnothing$ indicates no defects. We use $N = 16$ by default to balance localization accuracy and output length. Details of the resolution ablation are provided in Section 4.5.

**Grid construction.** The spatial annotations in the training data are masks or heatmaps. We use resizing and thresholding to convert them into grid cells. For sources that annotate the two defect categories separately, we retain separate cell sets for visual artifacts ($G_{\text{art}}$) and semantic misalignments ($G_{\text{mis}}$). We evaluate each set separately and use their union, $G = G_{\text{art}} \cup G_{\text{mis}}$, for aggregate localization. Sources without separate defect categories use a single grid. For heatmap annotations, we mark defect cells whose resized original heatmap value is at least $0.9$. We take the union of marked cells across defect categories to form $M \subseteq G$. These coverage marks reflect annotation support, not defect severity, and are used to weight the cell-level Dice reward (Section 3.4).

**Text output.** The model lists defect cells by row using 1-based indices. For example, `r5: 8,9!` identifies columns 8 and 9 in row 5 and marks column 9 as having high annotation coverage. It omits rows without defects and outputs `none` for an empty grid. The header lines `artifact:` and `misalign:` distinguish the two defect categories when separate grids are requested. This format preserves all defect-cell coordinates without listing all $N^2$ cells.

### 3.3 Supervised Fine-Tuning

We combine the five data sources, ImagenWorld (Mahdizadeh Sani et al., 2026), RichHF-18K (Liang et al., 2024), EvalMuse-40K (Han et al., 2026), PAL4VST (Zhang et al., 2023), and COCO (Lin et al., 2014), into a common training format. The resulting training corpus contains around 38K examples spanning score-only, localization-only, and joint supervision, including both unconditional and conditional generation/editing settings. We normalize scores to the shared scale and convert spatial annotations into defect grids as described in Section 3.2. Each example includes an evaluation instruction and a target response containing only the available supervision. ImagenWorld provides PQ and SC supervision and defect masks across generation and editing tasks, including tasks with conditioning images. EvalMuse's alignment score serves as the overall-score target, PAL4VST provides defect locations, and RichHF provides PQ and SC scores with separate artifact and misalignment grids and coverage marks. COCO images provide defect-free examples. Appendix A details source mappings, grid construction, and data separation. We fine-tune Qwen3-VL-8B (Bai et al., 2025) to generate these target responses using autoregressive cross-entropy as illustrated in Equation 1:

$$\mathcal{L}_{\text{SFT}} = -\sum_{i} \sum_{t=1}^{|y_i|} \log p_\theta \left( y_{i,t} \mid x_i, \tau_i, y_{i,<t} \right), \tag{1}$$

where $\theta$ denotes the trainable model parameters, $y_i$ is the target response, and $y_{i,t}$ is its $t$-th token. The model predicts each token from the input $x_i$, the evaluation instruction $\tau_i$, and the preceding target tokens $y_{i,<t}$. Only target-response tokens contribute to the loss. Missing supervision is distinct from an empty grid: score-only examples omit the grid field and are never trained to output `none` for localization. We cap clean examples with localization supervision at $10\%$ of the training corpus to limit the frequency of empty-grid targets.

### 3.4 Reinforcement Learning with GRPO

Starting from the SFT checkpoint, we apply GRPO (Shao et al., 2024) to optimize the rewards in Eq. 2. For each input, the model samples four responses. We parse each response into scores and de-

fect cells and compute its reward against the available ground truth. GRPO normalizes the rewards within each group and uses them to update the model with a clipped policy objective. This encourages responses with higher rewards relative to others for the same input. Throughout the paper, VIEScore2 denotes the final model after GRPO; VIEScore2 ($-\mathrm{GRPO}$) denotes the SFT checkpoint without GRPO. Parenthetical minus signs indicate omitted training stages or reward components. The total reward combines defect localization, score accuracy, and output format validity:

$$r_{\text{total}} = \lambda_{\text{dice}} r_{\text{dice}} + \lambda_{\text{score}} r_{\text{score}} + \lambda_{\text{format}} r_{\text{format}}, \tag{2}$$

where we use $(\lambda_{\text{dice}}, \lambda_{\text{score}}, \lambda_{\text{format}}) = (1, 0.3, 0.1)$. All rewards are computed from the parsed outputs without a learned reward model.

**Cell-level Dice reward.** We adapt the Dice-based objective (Milletari et al., 2016) to measure overlap between the ground-truth defect set $G$ and its prediction $\hat{G}$. For separate artifact and misalignment grids, both sets are unions across the two categories, as in Section 3.2. This reward measures defect localization but does not penalize category swaps. For each cell $c$, we set $w_c = 2$ if $c \in M$ and $w_c = 1$ otherwise, where $M$ is the union of ground-truth coverage marks. We count correctly predicted and missed defect cells as $\mathrm{TP}_w = \sum_{c \in G \cap \hat{G}} w_c$ and $\mathrm{FN}_w = \sum_{c \in G \setminus \hat{G}} w_c$. False positives remain unweighted: $\mathrm{FP} = |\hat{G} \setminus G|$. The parameter $\beta$ controls the precision–recall trade-off:

$$r_{\text{dice}} = \frac{(1+\beta^2)\mathrm{TP}_w}{(1+\beta^2)\mathrm{TP}_w + \beta^2 \mathrm{FN}_w + \mathrm{FP}}. \tag{3}$$

Larger $\beta$ places more emphasis on missed defects (false negatives), while smaller $\beta$ places more emphasis on incorrectly identified defects (false positives). We use $\beta = 1$ because it achieves the highest development grid IoU (Section 4.4). At this setting, the reward is a coverage-weighted Dice score. With unit cell weights, it reduces to standard Dice, equivalently cell-level $F_1$.

The Dice reward is 1 when both grids are empty and 0 when the predicted grid cannot be parsed. Reward computation and evaluation use the same rules to parse defect grids.

**Score and format rewards.** For a ground-truth score $s \in [0, 10]$ and its prediction $\hat{s}$, we use $r_{\text{score}} = 1 - |\hat{s} - s|/10$ as its score reward. If both PQ and SC are supervised, we average their individual score rewards. A missing or unparseable score receives zero reward. For examples with localization supervision, the format reward $r_{\text{format}}$ is 1 if the grid can be parsed. For score-only examples, it is 1 if a score can be parsed. Otherwise, it is 0. If an example lacks score or localization supervision, we set the corresponding reward term to zero for every response in the group. That term contributes no within-group reward difference.

### 3.5 Parameter-Free Parser

To make VIEScore2's predictions easy to read, a parameter-free parser converts the structured output into a textual explanation using fixed rules and templates. It extracts the predicted scores and defect grids, groups connected defect cells, and describes their locations. This step has no trainable parameters. When defect categories are predicted separately, it associates visual artifacts with perceptual quality and semantic misalignments with semantic consistency. The templates use only the predicted scores, defect categories, and locations, keeping explanations faithful to the structured predictions by construction. Appendix B provides training and inference settings and the rules used by the parameter-free parser.

## 4 Experiments

We assess VIEScore2 on quality scoring and defect localization across image generation and editing tasks and compare it with existing evaluators. We also examine the effects of GRPO and the trade-offs of the sparse grid representation.

### 4.1 Evaluation Setup

**Evaluation suites.** Our primary evaluation suite contains 1,300 held-out examples from the five data sources described in Section 3.3. Each metric uses only examples with the corresponding ground

Table 1: Defect localization on six test sets using a shared $16 \times 16$ grid. Methods use their supported inputs. IoU averages per-image overlap; $F_1$ pools cell counts. †API models use fixed prompts and temperature 0; §LEGION uses its released intermediate checkpoint.

| | Training sources | | | | Additional datasets | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | RichHF | | PAL4VST | | AbHuman | | HAD | | SynthScars | | SDG-30K | |
| Method | IoU | $F_1$ | IoU | $F_1$ | IoU | $F_1$ | IoU | $F_1$ | IoU | $F_1$ | IoU | $F_1$ |
| Qwen3-VL-8B | 0.022 | 0.137 | 0.026 | 0.124 | 0.024 | 0.108 | 0.044 | 0.111 | 0.026 | 0.117 | <u>0.114</u> | 0.269 |
| GPT-5.6-terra† | 0.096 | 0.218 | 0.118 | 0.239 | 0.171 | 0.217 | 0.148 | 0.251 | 0.162 | 0.281 | 0.058 | 0.217 |
| GPT-5.6-sol† | 0.156 | 0.274 | 0.141 | 0.248 | <u>0.217</u> | 0.282 | **0.211** | 0.298 | 0.186 | 0.290 | 0.078 | 0.258 |
| Gemini-3-Flash† | 0.047 | 0.082 | 0.048 | 0.068 | 0.057 | 0.079 | 0.075 | 0.119 | 0.062 | 0.101 | 0.031 | 0.068 |
| Claude Opus 5.5† | 0.167 | 0.285 | 0.140 | 0.225 | 0.146 | **0.308** | 0.142 | 0.235 | 0.196 | 0.358 | 0.084 | 0.216 |
| PAL | 0.014 | 0.035 | <u>0.306</u> | **0.535** | 0.009 | 0.016 | 0.009 | 0.038 | 0.027 | 0.046 | 0.007 | 0.014 |
| SegFormer-b0 | 0.274 | 0.450 | 0.146 | 0.237 | 0.130 | 0.246 | 0.156 | 0.286 | 0.170 | 0.283 | 0.103 | 0.260 |
| RAHF | <u>0.285</u> | **0.474** | 0.105 | 0.169 | 0.141 | 0.260 | 0.164 | 0.281 | 0.172 | 0.294 | 0.068 | 0.164 |
| ImageDoctor | 0.284 | <u>0.471</u> | 0.100 | 0.167 | 0.144 | <u>0.293</u> | 0.158 | 0.275 | 0.169 | 0.305 | 0.086 | 0.207 |
| LEGION§ | 0.086 | 0.176 | 0.074 | 0.130 | 0.174 | 0.152 | 0.191 | **0.324** | 0.218 | <u>0.363</u> | 0.038 | 0.120 |
| SDG | 0.140 | 0.243 | 0.115 | 0.196 | **0.253** | 0.292 | 0.154 | 0.228 | <u>0.222</u> | 0.330 | **0.137** | **0.353** |
| VIEScore2 | **0.299** | 0.466 | **0.335** | <u>0.439</u> | 0.215 | 0.194 | <u>0.197</u> | <u>0.314</u> | **0.234** | **0.368** | 0.102 | <u>0.272</u> |

truth. We evaluate localization on $1{,}100$ examples ($\mathcal{E}_G$), overall scores on $900$ examples, and PQ and SC scores on $700$ examples. COCO images are included in localization evaluation but excluded from score correlations because their score targets are assigned rather than human-rated. We also evaluate on larger held-out sets from PAL4VST and RichHF and a 2K EvalMuse subset. To assess transfer beyond the training sources, we use four additional localization datasets: AbHuman (Fang et al., 2024), HAD (Wang et al., 2024), SynthScars (Kang et al., 2025), and SDG-30K (Zhang et al., 2026).

**Inputs and decoding.** We distinguish two input settings: $\chi_0$ uses the generated image and prompt, while $\chi_K$ also includes the available conditioning images. The main scoring and joint-evaluation comparisons use $\chi_K$; the $\chi_0$ setting is examined separately.

For each source, VIEScore2 uses a fixed evaluation instruction to specify the requested scores, defect categories, and coverage marks (Section 3.1). We use greedy decoding and parse the generated text into scores and defect cells.

**Metrics.** We evaluate localization on the union of the applicable defect categories. Let $G_i$ and $\hat{G}_i$ denote the ground-truth and predicted cell sets for image $i$. We pool true-positive, false-positive, and false-negative cell counts across $\mathcal{E}_G$ to compute micro precision, recall, and $F_1$. We also report mean per-image grid IoU over $\mathcal{E}_G^+$, the examples with non-empty ground-truth grids:

$$\mathrm{IoU}_p^{\mathrm{grid}} = \frac{1}{|\mathcal{E}_G^+|} \sum_{i \in \mathcal{E}_G^+} \frac{|G_i \cap \hat{G}_i|}{|G_i \cup \hat{G}_i|}. \tag{4}$$

Per-image grid IoU measures localization overlap for each image and gives every image with annotated defects equal weight. This matches our goal of providing useful spatial feedback for individual images, whose defect regions can vary substantially in size. We therefore emphasize per-image grid IoU when comparing localization performance. Micro $F_1$ complements this measure by pooling cell counts across images, giving greater influence to images with larger ground-truth or predicted defect regions. We also report false alarms on clean images. Pixel-IoU is used only for native dense-mask and grid-resolution comparisons. When both PQ and SC are predicted, their geometric mean gives the overall score. For score evaluation, we report Spearman rank correlation (SRCC). Overall, PQ, and SC correlations each use examples with the corresponding score annotations. Correlations use successfully parsed prediction–target pairs and are undefined for constant score vectors. We report valid-pair counts when parsing coverage is incomplete.

**Compared models.** We compare VIEScore2 with general-purpose VLMs and specialized image evaluators. The general-purpose models are Qwen3-VL-8B (Bai et al., 2025), GPT-5.6-terra (Ope-

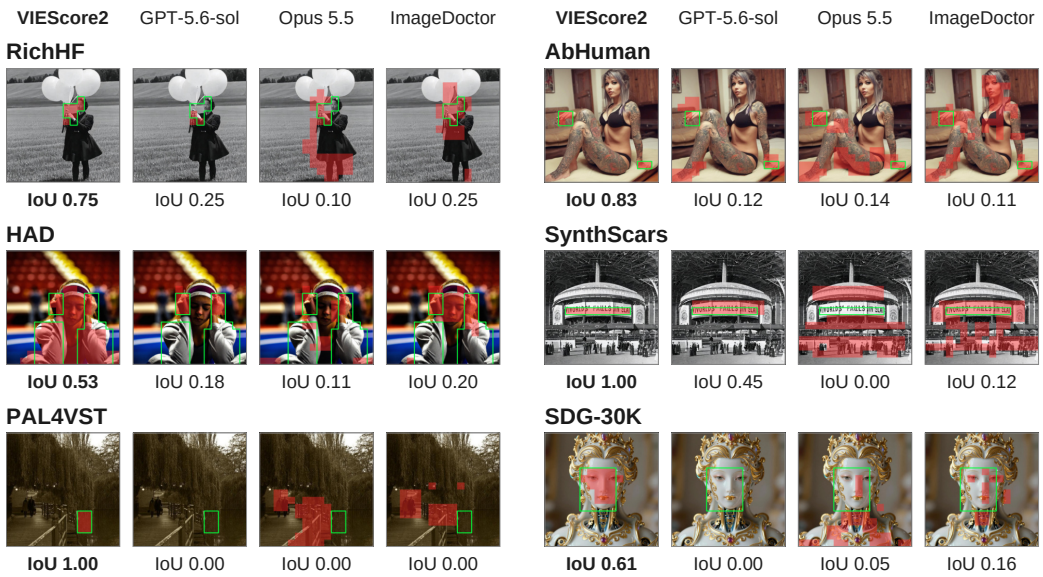



Figure 3: Selected defect-localization examples across six benchmarks. Red cells denote predictions and green outlines denote ground-truth defect regions. Numbers report per-image grid IoU. Additional qualitative examples and failure cases are provided in Appendix F.

nAI, 2026b), GPT-5.6-sol (OpenAI, 2026a), Gemini-3-Flash (Google, 2025), and Claude Opus 5.5 (Anthropic, 2026). Spatial evaluators include RAHF (Liang et al., 2024), ImageDoctor (Guo et al., 2026), PAL (Zhang et al., 2023), LEGION (Kang et al., 2025), and SDG (Zhang et al., 2026). We also train SegFormer-b0 (Xie et al., 2021) on the same spatial supervision as a dense-localization baseline. Each method is evaluated only on the outputs it supports. Appendix C provides the evaluation subsets, metric conventions, and baseline settings. In tables with ranking annotations, **bold** denotes the best value and underlining the second-best distinct value in each column. Tied values share the same marking.

### 4.2 Strong Localization across Benchmarks

Table 1 compares all spatial predictions on the common grid. The baselines cover heatmaps (RAHF and ImageDoctor), masks (PAL and LEGION), and boxes (SDG). We use their released models and fixed spatial-conversion settings. Each method uses its supported inputs; a shared grid aligns the evaluation space, not the input interfaces.

VIEScore2 ranks first in per-image grid IoU on RichHF (0.299), PAL4VST (0.335), and SynthScars (0.234), second on HAD, and third on AbHuman. It also leads in SynthScars $F_1$ (0.368). Other methods lead on different benchmarks: GPT-5.6-sol in HAD IoU, Claude Opus 5.5 in AbHuman $F_1$, and SDG on SDG-30K. The top-three IoU rankings on five benchmarks show competitive localization across training sources and additional datasets. GRPO improves grid IoU over VIEScore2 ($-$GRPO) on all six datasets and improves $F_1$ on five. Section 4.4 separates the effect of GRPO from additional SFT and examines the reward components.

**Qualitative comparison.** Figure 3 compares selected examples from six benchmarks. VIEScore2 more closely matches the annotated regions, including the sign on SynthScars and tree region on PAL4VST. Competing predictions miss these regions or cover surrounding areas. The HAD example also exposes a limitation: VIEScore2 marks unannotated parts of the face.

### 4.3 Joint Scoring and Localization across Tasks

**Quality scoring.** Table 2 compares overall-score predictions under the same $\chi_K$ input setting. On the primary suite, VIEScore2 achieves the highest aggregate SRCC (0.601), followed by Gemini-3-Flash (0.491) and GPT-5.6-sol (0.437). All five models receive the generated image, prompt,

Table 2: Overall-score SRCC on the primary suite (900 examples) with matched $\chi_K$ inputs. †API correlations use successfully parsed scores. ‡Qwen3-VL-8B uses our evaluation instructions without fine-tuning.

| Method | All | RichHF | EvalMuse | TIG | TIE | SRIG | SRIE | MRIG | MRIE |
|---|---|---|---|---|---|---|---|---|---|
| Qwen3-VL-8B judge‡ | 0.373 | 0.456 | 0.473 | 0.570 | 0.066 | 0.128 | 0.393 | 0.537 | −0.290 |
| GPT-5.6-terra† | 0.402 | 0.440 | 0.567 | **0.640** | 0.493 | 0.258 | 0.110 | 0.523 | 0.539 |
| GPT-5.6-sol† | 0.437 | 0.470 | 0.655 | 0.623 | 0.590 | 0.275 | 0.253 | **0.539** | **0.684** |
| Gemini-3-Flash† | 0.491 | 0.515 | 0.535 | 0.531 | **0.686** | 0.356 | **0.404** | 0.418 | 0.677 |
| VIEScore2 | **0.601** | **0.692** | **0.803** | 0.589 | 0.429 | **0.375** | 0.403 | 0.509 | 0.417 |

Table 3: Joint evaluation with matched $\chi_K$ inputs. Localization uses 1,100 examples; PQ/SC SRCC uses 700. Grid IoU excludes empty ground-truth grids. †GPT-5.6-terra returns valid scores for $554/700$ examples.

| Model | Grid IoU | $F_1$ | PQ SRCC | SC SRCC |
|---|---|---|---|---|
| Qwen3-VL-8B | 0.070 | 0.273 | 0.181 | 0.352 |
| GPT-5.6-terra† | 0.112 | 0.247 | 0.357 | 0.508 |
| VIEScore2 | **0.324** | **0.506** | **0.558** | **0.564** |

and available conditioning images. ImagenWorld covers text-to-image generation (TIG), text-guided editing (TIE), and generation/editing with one (SRIG/SRIE) or multiple conditioning images (MRIG/MRIE). Each task contains 50 examples. VIEScore2 also leads on RichHF, EvalMuse, and SRIG. Gemini-3-Flash leads on TIE and SRIE, GPT-5.6-sol on MRIG and MRIE, and GPT-5.6-terra on TIG.

**Joint evaluation.** Table 3 compares joint localization and quality scoring under the same $\chi_K$ input setting. VIEScore2 achieves higher localization $F_1$, grid IoU, and PQ and SC correlations than the two baselines evaluated in this joint setting. On the same 250 conditional examples, providing the available conditioning images improves overall-score SRCC from 0.420 to 0.451 and localization F1 from 0.514 to 0.536. This comparison keeps the model, evaluation instruction, decoding, grid-parsing rules, and evaluation support fixed, isolating the contribution of the conditioning inputs. The MMRB2 preference evaluation shows competitive text-to-image performance but weaker editing accuracy than the general-purpose VLM baselines. Appendix D provides the full task-level results and the separate $\chi_0$ comparisons.

### 4.4 GRPO Improves Localization beyond Additional SFT

Across three independent runs, GRPO improves grid IoU by 0.024–0.029 over the shared SFT checkpoint, whereas an additional SFT epoch gives 0.001. These gains show that additional SFT does not reproduce the localization improvement from GRPO. We compare reward components under matched training examples, sampling budget, and optimization steps. VIEScore2 ($-r_{\text{score}}, -r_{\text{format}}$) uses only the Dice reward and raises grid IoU from 0.296 to 0.320, accounting for most of the gain. The full reward reaches 0.324 IoU and 0.506 F1 at an overall-score SRCC of 0.601. The Dice-only variant reaches the same SRCC (Table 18), so the score reward does not measurably change scoring on the primary suite; its main effect is a small F1 gain (0.481 to 0.506). The format reward has no measurable effect because every variant already parses on all localization examples. We retain both terms as safeguards for training runs where parse failures or score drift could occur, but the localization gain is attributable to the cell-level Dice reward alone.

The precision–recall ablation supports our choice of $\beta = 1$ for grid IoU. On the primary suite, GRPO improves both precision (0.415 to 0.466) and recall (0.545 to 0.552). Appendix E.1 reports the complete GRPO ablations, seed comparisons, and per-category and clean-image diagnostics. Appendix F also shows improved overlap and missed defects after GRPO.

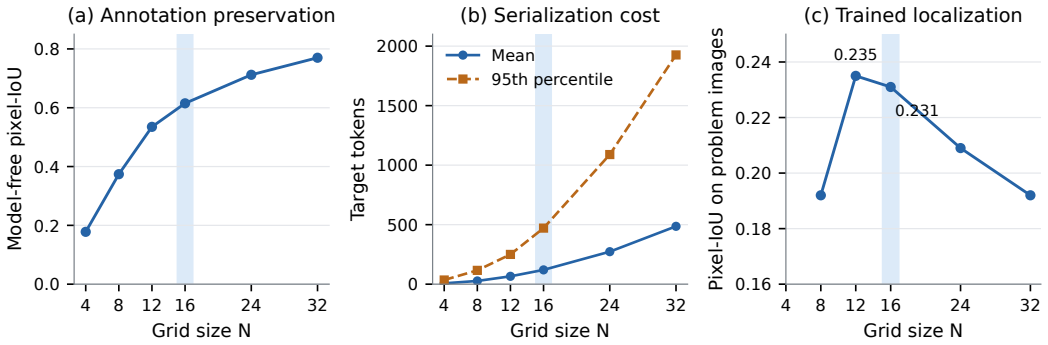



Figure 4: Grid resolution versus annotation fidelity, output length, and trained localization. Annotation fidelity compares grids with native masks. Shading marks $N = 16$.

### 4.5 Grid Resolution Balances Fidelity and Output Length

Figure 4 shows how grid resolution affects annotation fidelity, output length, and trained localization. Trained pixel-IoU$_p$ is similar at $N = 12$ and $N = 16$ (0.235/0.231), then falls to 0.192 at $N = 32$. We choose $N = 16$ for its higher annotation fidelity than $N = 12$ (model-free pixel-IoU 0.615/0.535), while its 95th-percentile target is shorter than at $N = 32$ (471/1,926 tokens). This choice balances spatial detail and output length rather than maximizing every metric. Although bounding boxes achieve higher $F_1$ in the matched-budget localization-only ablation (0.463 vs. 0.397), we use sparse cells for their simple text format and direct cell-level reward computation. Appendix E.2 provides the complete representation and resolution comparisons.

## 5 Limitations

**Model choice.** We validate VIEScore2 only with Qwen3-VL-8B, so its effectiveness across other VLM backbones and model scales remains unknown. In addition, comparisons with released evaluators do not disentangle the effects of architecture, training data, and training procedure.

**Representation and explanation scope.** The fixed grid can miss small defects and only coarsely approximate irregular boundaries. The parameter-free parser produces explanations that are faithful to the predicted scores and defect grids, but does not independently verify their correctness or identify defect causes. GRPO can increase false alarms on some clean-image subsets, reflecting a trade-off between stronger defect localization and conservative clean-image recognition. Appendix E provides the full representation and resolution comparisons.

## 6 Conclusion

We introduced VIEScore2, a unified framework for explainable image evaluation across generation and editing tasks. Its sparse grid representation unifies heterogeneous spatial supervision across datasets and supports joint prediction of quality scores and defect locations. Using 38K training examples spanning score-only, localization-only, and joint supervision, we train the evaluator through supervised fine-tuning followed by GRPO with verifiable rewards. A parameter-free parser then produces explanations grounded in the structured predictions. Experiments demonstrate stronger alignment with human quality ratings than general-purpose VLMs under matched inputs, together with competitive localization across six benchmarks, including datasets beyond the training sources. Ablations further show that GRPO improves localization beyond additional supervised fine-tuning.

### AI USE STATEMENT

In this work, we used generative AI tools to help to refine hypotheses, provide feedback on research methodology and experiments, implement methods, assist with translation. We have not used generative AI tools to generate synthetic datasets, help develop theoretical models or conceptual frameworks, formulate mathematical claims. Other required disclosure categories are not applicable to this work. Additionally, we used generative AI tools to create figures, summarize or analyze existing literature, format references. We have reviewed all AI-assisted work. AI-assisted methodological suggestions, hypothesis refinement, code, figures, literature summaries and analyses, translations, and reference formatting were manually inspected and validated against the relevant experimental results, source materials, and citation records, and any AI-assisted manuscript content was reviewed and edited by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

### ETHICS STATEMENT

VIEScore2 evaluates generated images using existing research datasets; no new human annotations were collected. We use these datasets under their respective licenses. The model may flag legitimate image content as defective or miss actual defects. Its explanations reflect its predictions and do not independently verify them. Scores, defect grids, and explanations should therefore support human judgment rather than replace it.

### REPRODUCIBILITY STATEMENT

Training, evaluation, and data preparation scripts are available in our anonymous repository https://anonymous.4open.science/r/VIEScore2-E532/. The scripts construct training data from public sources using fixed seeds. Data separation checks cover evaluation identifiers, alternate annotations of the same image, ImagenWorld cases, and exact and perceptual image matches. Appendices A.2, B, and C detail the data splits, training settings, evaluation instructions, and metric conventions.

## REFERENCES

Anthropic. Introducing Claude Opus 5.5. https://www.anthropic.com/claude-opus-5-5, 2026.

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.

Jaemin Cho, Yushi Hu, Jason Baldridge, Roopal Garg, Peter Anderson, Ranjay Krishna, Mohit Bansal, Jordi Pont-Tuset, and Su Wang. Davidsonian scene graph: Improving reliability in fine-grained evaluation for text-to-image generation. In *International Conference on Learning Representations*, pp. 15625–15645, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/446905abfec9db8856f7f7465bbd5552-Paper-Conference.pdf.

Guian Fang, Wenbiao Yan, Yuanfan Guo, Jianhua Han, Zutao Jiang, Hang Xu, Shengcai Liao, and Xiaodan Liang. HumanRefiner: Benchmarking abnormal human generation and refining with coarse-to-fine pose-reversible guidance. In *European Conference on Computer Vision*, 2024.

Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. DreamSim: Learning new dimensions of human visual similarity using synthetic data. In *Advances in Neural Information Processing Systems*, volume 36, pp. 50742–50768, 2023. doi: 10.52202/075280-2208. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/9f09f316a3eaf59d9ced5ffaefe97e0f-Paper-Conference.pdf.

Google. Gemini 3 Flash Preview. https://ai.google.dev/gemini-api/docs/models/gemini-3-flash-preview, 2025.

Yuxiang Guo, Jiang Liu, Ze Wang, Hao Chen, Ximeng Sun, Yang Zhao, Jialian Wu, Xiaodong Yu, Zicheng Liu, and Emad Barsoum. ImageDoctor: Diagnosing text-to-image generation via grounded image reasoning. In *International Conference on Learning Representations*, pp. 137704–137724, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/deb0e85779f2f003b10528de72b1ebf1-Paper-Conference.pdf.

Shuhao Han, Haotian Fan, Jiachen Fu, Liang Li, Tao Li, Junhui Cui, Yunqiu Wang, Yang Tai, Jingwei Sun, Chun-Le Guo, and Chongyi Li. EvalMuse-40K: A fine-grained benchmark with comprehensive human annotations for text-to-image generation model alignment evaluation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(6):4583–4591, 2026. doi: 10.1609/aaai.v40i6.42458. URL https://ojs.aaai.org/index.php/AAAI/article/view/42458.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 7514–7528, 2021.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A Smith. TIFA: Accurate and interpretable text-to-image faithfulness evaluation with question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20406–20417, 2023. URL https://openaccess.thecvf.com/content/ICCV2023/html/Hu_TIFA_Accurate_and_Interpretable_Text-to-Image_Faithfulness_Evaluation_with_Question_Answering_ICCV_2023_paper.html.

Yushi Hu, Reyhane Askari-Hemmat, Melissa Hall, Emily Dinan, Luke Zettlemoyer, and Marjan Ghazvininejad. Multimodal RewardBench 2: Evaluating omni reward models for interleaved text and image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 36904–36915, 2026.

Hengrui Kang, Siwei Wen, Zichen Wen, Junyan Ye, Weijia Li, Peilin Feng, Baichuan Zhou, Bin Wang, Dahua Lin, Linfeng Zhang, et al. LEGION: Learning to ground and explain for synthetic image detection. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18937–18947. IEEE, 2025.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-Pic: An open dataset of user preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, volume 36, pp. 36652–36663, 2023. doi: 10.52202/075280-1594. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/73aacd8b3b05b4b503d58310b523553c-Paper-Conference.pdf.

Max Ku, Dongfu Jiang, Cong Wei, Xiang Yue, and Wenhu Chen. VIEScore: Towards explainable metrics for conditional image synthesis evaluation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12268–12290, 2024a.

Max Ku, Tianle Li, Kai Zhang, Yujie Lu, Xingyu Fu, Wenwen Zhuang, and Wenhu Chen. ImagenHub: Standardizing the evaluation of conditional image generation models. In *The Twelfth International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=OuV9ZrkQlc.

Mengcheng Lan, Chaofeng Chen, Yue Zhou, Jiaxing Xu, Yiping Ke, Xinjiang Wang, Litong Feng, and Wayne Zhang. Text4seg: Reimagining image segmentation as text generation, 2025. URL https://arxiv.org/abs/2410.09855.

Weiqi Li, Xuanyu Zhang, Shijie Zhao, Yabin Zhang, Junlin Li, Li Zhang, and Jian Zhang. Q-Insight: Understanding image quality via visual reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 36802–36827, 2025. doi: 10.52202/085713-1237. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/3479362c2eb7321ba0490d98eb032772-Paper-Conference.pdf.

Youwei Liang, Junfeng He, Gang Li, Peizhao Li, Arseniy Klimovskiy, Nicholas Carolan, Jiao Sun, Jordi Pont-Tuset, Sarah Young, Feng Yang, et al. Rich human feedback for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19401–19411, 2024.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation, 2024. URL https://arxiv.org/abs/2404.01291.

Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. Visual-rft: Visual reinforcement fine-tuning, 2025. URL https://arxiv.org/abs/2503.01785.

Yiting Lu, Fengbin Guan, Yixin Gao, Yan Zhong, Xinge Peng, Jiakang Yuan, Yihao Liu, Bo Zhang, Xin Li, Zhibo Chen, and Weisi Lin. OmniQuality-R: Advancing reward models through all-encompassing quality assessment. *arXiv preprint arXiv:2510.10609*, 2025.

Samin Mahdizadeh Sani, Max Ku, Nima Jamali, Matina Sani, Paria Khoshtab, Wei-Chieh Sun, Parnian Fazel, Zhi Rui Tam, Thomas Chong, Edisy Kin Wai Chan, Donald Tsang, Chiao-Wei Hsu, Ting Lam, Ho Ng, Chiafeng Chu, Chak-Wing Mak, Keming Wu, Wong Hiu-Tung, Yik Ho, Chi Ruan, Zhuofeng Li, I-Sheng Fang, Shih-Ying Yeh, Ho Kei Cheng, Ping Nie, and Wenhu Chen. ImagenWorld: Stress-testing image generation models with explainable human evaluation on open-ended real-world tasks. In *International Conference on Learning Representations*, pp. 41886–41916, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/458fa8ee331566383d8e74bdb647f829-Paper-Conference.pdf.

Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pp. 565–571, Los Alamitos, CA, USA, October 2016. IEEE Computer Society. doi: 10.1109/3DV.2016.79. URL https://doi.ieeecomputersociety.org/10.1109/3DV.2016.79.

OpenAI. GPT-5.6 Sol model. https://developers.openai.com/api/docs/models/gpt-5.6-sol, 2026a.

OpenAI. GPT-5.6 Terra model. https://developers.openai.com/api/docs/models/gpt-5.6-terra, 2026b.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

Seyed Sadegh Mohseni Salehi, Deniz Erdogmus, and Ali Gholipour. Tversky loss function for image segmentation using 3d fully convolutional deep networks, 2017. URL https://arxiv.org/abs/1706.05721.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Jifang Wang, Xue Yang, Longyue Wang, Zhenran Xu, Yiyu Wang, Yaowei Wang, Weihua Luo, Kaifu Zhang, Baotian Hu, and Min Zhang. A unified agentic framework for evaluating conditional image generation, 2025a. URL https://arxiv.org/abs/2504.07046.

Kaihong Wang, Lingzhi Zhang, and Jianming Zhang. Detecting human artifacts from text-to-image models. *arXiv preprint arXiv:2411.13842*, 2024.

Yibin Wang, Zhimin Li, Yuhang Zang, Chunyu Wang, Qinglin Lu, Cheng Jin, and Jiaqi Wang. Unified multimodal chain-of-thought reward model through reinforcement fine-tuning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 159130–159157, 2025b. doi: 10.52202/085713-5315. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/e95e9f0c127aa1cfa2628adb2f3cb107-Paper-Conference.pdf.

Haoning Wu, Zicheng Zhang, Weixia Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Yixuan Gao, Annan Wang, Erli Zhang, Wenxiu Sun, Qiong Yan, Xiongkuo Min, Guangtao Zhai, and Weisi Lin. Q-Align: Teaching LMMs for visual scoring via discrete text-defined levels. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 54015–54029. PMLR, 2024. URL https://proceedings.mlr.press/v235/wu24ah.html.

Tianhe Wu, Jian Zou, Jie Liang, Lei Zhang, and Kede Ma. VisualQuality-R1: Reasoning-induced image quality assessment via reinforcement learning to rank. In *Advances in Neural Information Processing Systems*, volume 38, pp. 88167–88190, 2025. doi: 10.52202/085713-2947. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/7f8f7bf2d357a09bdfff04b2e0f5c4e-Paper-Conference.pdf.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023.

Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. SegFormer: Simple and efficient design for semantic segmentation with transformers. In *Advances in Neural Information Processing Systems*, 2021.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. ImageReward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

Fan Yang, Ru Zhen, Jianing Wang, Yanhao Zhang, Haoxiang Chen, Haonan Lu, Sicheng Zhao, and Guiguang Ding. HEIE: MLLM-Based hierarchical explainable AIGC image implausibility evaluator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3856–3866, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Yang_HEIE_MLLM-Based_Hierarchical_Explainable_AIGC_Image_Implausibility_Evaluator_CVPR_2025_paper.html.

Huaisong Zhang, Hao Yu, Yuxuan Zhang, Jiahe Wang, Xinrui Chen, Haoxiang Cao, Feng Lu, Wendong Zhang, Changqian Yu, and Chun Yuan. Where, what, why, and importance: Structured defect grounding for text-to-image feedback. *arXiv preprint arXiv:2606.06113*, 2026.

Lingzhi Zhang, Zhengjie Xu, Connelly Barnes, Yuqian Zhou, Qing Liu, He Zhang, Sohrab Amirghodsi, Zhe Lin, Eli Shechtman, and Jianbo Shi. Perceptual artifacts localization for image synthesis tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7579–7590, 2023.

# A Datasets and Annotation Mappings

## A.1 Source Supervision and Target Construction

**Source supervision.** Table 4 summarizes the score and spatial supervision available from each source.

Table 4: Supervision and requested output fields by source. "–" indicates an omitted field.

| Source | Scores | Localization | Raw supervision |
|---|---|---|---|
| ImagenWorld | PQ, SC | single grid | ratings, masks |
| RichHF | PQ, SC | artifact and misalignment grids | ratings, heatmaps |
| PAL4VST | – | single grid | masks |
| EvalMuse | overall | – | alignment ratings |
| COCO | PQ, SC | single grid | clean constraint |

**Score normalization.** The source mappings retain the meaning of each available rating. For RichHF, the normalized PQ target is the mean of the artifact and aesthetic ratings, and SC is the normalized misalignment rating. For ImagenWorld, each 1–5 rating $r$ is mapped to 0–1 by $(r-1)/4$; PQ averages the normalized artifact and aesthetic-quality ratings, and SC uses prompt relevance. Both are then scaled to 0–10. EvalMuse's 1–5 mean alignment rating $r$ is mapped to 0–10 by $2.5(r-1)$. Training targets are rounded to integer scores. COCO supplies clean examples with PQ and SC targets of 10 and empty defect grids. These assigned targets are excluded from score correlations.

**Spatial annotation processing.** Binary masks are resized to $N \times N$ using LANCZOS and thresholded at $0.5$. RichHF heatmaps are first binarized at $0.5$ before resizing; the two defect categories are processed separately and combined only for aggregate evaluation. Coverage marks identify defect cells whose resized original heatmap response is at least $0.9$. They measure annotation support rather than severity. Bounding boxes are filled before rasterization.




Figure 5: RichHF annotations and their $16 \times 16$ grid targets. Red and blue denote artifact and misalignment; "!" marks high annotation coverage.

## A.2 Corpus Composition and Data Separation

**Training corpus.** Table 5 summarizes the training, validation, and primary evaluation splits. COCO images use human-written captions and provide clean-image supervision.

**Training and evaluation separation.** Training excludes evaluation images, alternate annotations of the same image, and held-out ImagenWorld cases. Exact and perceptual image hashes are checked for overlap.

A prompt-overlap audit changes overall SRCC by less than $0.001$ after removing the affected EvalMuse example. The primary suite shares generators with training; transfer is assessed separately on additional localization datasets.

Table 5: Data splits by source. Counts are records, which may share an image.

| Source | Train | Validation | Primary eval. | Conditioning images |
|---|---|---|---|---|
| RichHF | 13,920 | 1,518 | 400 | no |
| EvalMuse | 9,000 | 1,000 | 200 | no |
| PAL4VST | 8,177 | 908 | 300 | no |
| ImagenWorld editing/reference | 3,642 | 415 | 250 | yes |
| ImagenWorld TIG | 1,165 | 121 | 50 | no |
| COCO | 2,218 | 272 | 100 | no |
| Total | 38,122 | 4,234 | 1,300 | |

# B Implementation Details

## B.1 Optimization and Inference

**Input processing.** Generated and conditioning images are resized with preserved aspect ratio to at most $384^2$ and $256^2$ pixels, respectively. The SFT sequence limit is $2{,}048$ tokens.

**SFT.** We fine-tune Qwen3-VL-8B with LoRA (Hu et al., 2021) on all linear layers (rank 32, scaling 16, dropout $0.05$). AdamW uses a learning rate of $2 \times 10^{-4}$, cosine decay, $3\%$ warmup, zero weight decay, gradient clipping at $1.0$, and effective batch size 32. Training runs for five epochs; epoch 4 initializes GRPO, and epoch 5 provides the additional-SFT control. A separate $1\%$ training subset monitors loss.

**GRPO.** The epoch-4 adapter is merged into the backbone for full-parameter post-training. We use 8-bit AdamW with learning rate $10^{-5}$, one epoch, four responses per example, eight gradient-accumulation steps, and no KL penalty. Each run samples $2{,}400$ training examples stratified by supervision type, conditioning, and defect density. Completion length is limited to $768$ tokens. Rewards follow Sec. 3.4 with $(\lambda_{\text{dice}}, \lambda_{\text{score}}, \lambda_{\text{format}}) = (1, 0.3, 0.1)$. The Dice reward is computed on the union of the artifact and misalignment grids. The format reward checks that the grid parses for examples with localization supervision and that the score parses for score-only examples. Unavailable supervision contributes no advantage. Three runs use independent seeds and sampled subsets.

**Inference.** We use greedy decoding with a limit of $1{,}024$ output tokens.

## B.2 Evaluation Instructions and Parameter-Free Parser

Each source requests only its supported fields (Table 4). EvalMuse has no grid target, whereas a clean localization example uses `none`. Conditioning images precede the generated image. The full evaluation instructions are given in Appendix G.

The parameter-free parser groups four-connected cells separately for each defect category and describes up to three components, ordered by size. Each description gives a coarse location and coordinate bounds; additional components are counted. Scores below $5$, from $5$ to below $8$, and at least $8$ are described as low, moderate, and high. Coverage wording depends on the fraction of marked cells. Figure 7 shows a complete output and its generated explanation. The explanation preserves the prediction's content but does not establish its correctness.

# C Evaluation Setup and Baselines

## C.1 Evaluation Support

**Primary evaluation suite.** Table 6 gives the examples supporting each metric. Scores and localization are evaluated only where the corresponding annotations are available. COCO score targets are assigned and excluded from score correlations; EvalMuse has no localization targets.

Table 6: Examples available for each metric in the primary suite.

| Primary source | Total | Localization | Overall score | PQ/SC |
|---|---|---|---|---|
| RichHF | 400 | 400 | 400 | 400 |
| PAL4VST | 300 | 300 | 0 | 0 |
| EvalMuse | 200 | 0 | 200 | 0 |
| ImagenWorld (six tasks) | 300 | 300 | 300 | 300 |
| COCO | 100 | 100 | 0 | 0 |
| Total | 1,300 | 1,100 | 900 | 700 |

**Extended evaluation sets.** Table 7 lists the evaluation sets beyond the primary suite. RichHF reserves 100 images for threshold tuning. The AbHuman (Fang et al., 2024) subset uses seed 42; HAD (Wang et al., 2024) includes DALL-E 2/3 ($159/199$ images) and Midjourney/SDXL (350 each).

Table 7: External evaluation subsets. AbHuman, HAD, SynthScars, and SDG-30K report localization on non-empty ground-truth grids.

| Dataset | Evaluation subset | $n$ |
|---|---|---|
| PAL4VST | Full test set | 1,405 |
| RichHF | Test split after threshold tuning | 840 |
| EvalMuse | External score subset | 2,000 |
| AbHuman | 2K validation subset | 1,407 |
| HAD | Validation subset | 709 |
| SynthScars | Official test split (1,000 images) | 951 |
| SDG-30K | Official test split (1,154 images) | 887 |
| MMRB2 T2I / Edit | Preference pairs | 1,000 each |

### C.2 Scores, Empty Grids, and Parse Failures

When a model emits both PQ and SC, its overall prediction is their geometric mean; a score-only evaluator uses its native scalar. Predictions are compared with the source's stored overall target. SRCC uses average ranks for ties, and correlations are undefined for constant vectors. A scorer's valid-pair count must accompany a correlation whenever parsing coverage is incomplete. A same-support comparison additionally restricts both methods to the same valid pairs.

Micro precision, recall, and $F_1$ pool cell counts across the localization support. In contrast, $\mathrm{IoU}_p^{\mathrm{grid}}$ averages per-image IoU only over non-empty ground-truth grids, as in Eq. 4. A successfully parsed `none` is an empty prediction. An unparseable grid receives no true positives and misses all positive ground-truth cells; it remains a parse failure and must not be reported as successful clean-image recognition. For clean images, false-alarm counts report how many images have at least one predicted defect cell. They complement $\mathrm{IoU}_p^{\mathrm{grid}}$, which does not measure clean-image behavior.

For the 900-example score comparison in Table 2, unparseable scores exclude 154 examples for GPT-5.6-terra and at most 2 for each other API model. Assigning the scale midpoint to these failures reduces GPT-5.6-terra's aggregate SRCC to 0.357 without changing the other rows at the reported precision.

### C.3 Baseline Adaptation

Spatial predictions are rasterized into the same $16 \times 16$ grid when reporting grid metrics. Artifact and misalignment heatmaps are mean-pooled to $16 \times 16$ and combined as $\max(\text{artifact}, \text{misalignment})$; box and mask outputs use the same grid conversion as the ground truth. All released thresholds and tuned settings remain fixed across benchmarks. Table 8 lists these settings.

Table 8: Spatial conversion settings, fixed across benchmarks.

| Method | Spatial output | Threshold | Setting source |
|---|---|---|---|
| RAHF | Two heatmaps | 0.06 | RichHF (100) |
| ImageDoctor | Two heatmaps | 0.03 | RichHF (100) |
| PAL | Artifact mask | 0.5 | Released default |
| SegFormer-b0 | Defect probabilities | 0.75 | Validation |
| LEGION | Artifact mask | 0.5 cell fraction | Released rule |
| SDG | Defect boxes | – | Released output |

### C.4 Evaluator Output Interfaces

Table 9 summarizes the output interfaces of closely related evaluators.

Table 9: Evaluator output interfaces. Cond. denotes support for conditioning images; "–" denotes an unsupported output or training setting.

| Method | Scores | Spatial output | Text feedback | GRPO | Cond. |
|---|---|---|---|---|---|
| VIEScore (Ku et al., 2024a) | ✓ | – | Rationale | – | ✓ |
| RAHF (Liang et al., 2024) | ✓ | Heatmaps | – | – | – |
| ImageDoctor (Guo et al., 2026) | ✓ | Heatmap | Reasoning | ✓ | – |
| PAL (Zhang et al., 2023) | – | Mask | – | – | – |
| LEGION (Kang et al., 2025) | – | Mask | Explanation | – | – |
| FGA-BLIP2 (Han et al., 2026) | ✓ | – | – | – | – |
| HEIE (Yang et al., 2025) | ✓ | Heatmap | Analysis | – | – |
| SDG (Zhang et al., 2026) | – | Boxes | Defect text | ✓ | – |
| VIEScore2 (ours) | ✓ | $16 \times 16$ grid | Parameter-free parser | ✓ | ✓ |

RAHF and ImageDoctor predict four scores; VIEScore2 predicts an overall score or separate PQ and SC scores, as requested. GRPO optimizes scores and heatmaps in ImageDoctor, structured defects in SDG, and the cell-level Dice reward and scores in VIEScore2.

## D Additional Evaluation Results

### D.1 Primary-Suite Results by Metric and Task

Table 10 expands the joint evaluation in Table 3 with precision, recall, and VIEScore2 ($-\mathrm{GRPO}$). Relative to SFT, GRPO improves SC correlation but lowers PQ correlation from $0.580$ to $0.558$. Table 11 reports VIEScore2 results by task and source. Each ImagenWorld task contains $50$ examples, so these task-level comparisons are diagnostic.

Table 10: Full primary-suite results: localization on $1{,}100$ examples and PQ/SC SRCC on $700$. Rankings compare the $\chi_K$ VLMs; SegFormer uses only the generated image. †GPT-5.6-terra returns valid scores for $554/700$ examples.

| Model | $Precision$ | $Recall$ | $F_1$ | $\mathrm{IoU}_p^{\mathrm{grid}}$ | PQ SRCC | SC SRCC |
|---|---|---|---|---|---|---|
| Qwen3-VL-8B | 0.211 | 0.388 | 0.273 | 0.070 | 0.181 | 0.352 |
| GPT-5.6-terra† | 0.257 | 0.239 | 0.247 | 0.112 | 0.357 | 0.508 |
| VIEScore2 ($-\mathrm{GRPO}$) | <u>0.415</u> | <u>0.545</u> | <u>0.471</u> | <u>0.296</u> | **0.580** | <u>0.532</u> |
| VIEScore2 | **0.466** | **0.552** | **0.506** | **0.324** | <u>0.558</u> | **0.564** |
| SegFormer-b0 (matched loc.) | 0.433 | 0.573 | 0.493 | 0.272 | – | – |

Table 11: Primary-suite results by task and source. $\rho_o$ denotes overall-score SRCC; each ImagenWorld task has 50 examples.

| Task/source | $F_1$ | $\rho_o$ | Task/source | $F_1$ | $\rho_o$ |
|---|---|---|---|---|---|
| TIG | 0.572 | 0.589 | MRIG | 0.545 | 0.509 |
| TIE | 0.692 | 0.429 | MRIE | 0.496 | 0.417 |
| SRIG | 0.512 | 0.375 | RichHF | 0.465 | 0.692 |
| SRIE | 0.412 | 0.403 | PAL4VST | 0.376 | – |
| | | | EvalMuse | – | 0.803 |

### D.2 Conditioning Images and Score Evaluation

Table 12 reports the $\chi_0$ setting, which uses only the generated image and prompt. These results are separate from the $\chi_K$ comparison in Table 2 because access to conditioning images differs. Score evaluators include ImageReward (Xu et al., 2023), PickScore (Kirstain et al., 2023), HPSv2 (Wu et al., 2023), VQAScore (Lin et al., 2024), FGA-BLIP2 (Han et al., 2026), Q-Align (Wu et al., 2024), Q-Insight (Li et al., 2025), and OmniQuality-R (Lu et al., 2025). ImageDoctor (Guo et al., 2026) also provides quality scores under this setting. Among these evaluators, FGA-BLIP2 achieves the highest aggregate and EvalMuse SRCC, while ImageDoctor leads on RichHF. Table 13 isolates the effect of conditioning images using the same VIEScore2 model on the same conditional examples.

Table 12: Overall-score SRCC with the generated image and prompt only ($\chi_0$). Rankings compare methods within this input setting.

| Method | All | RichHF | EvalMuse | TIG | TIE | SRIG | SRIE | MRIG | MRIE |
|---|---|---|---|---|---|---|---|---|---|
| ImageReward | 0.239 | 0.227 | 0.595 | 0.107 | −0.278 | 0.195 | 0.151 | 0.035 | 0.263 |
| PickScore | 0.233 | 0.263 | 0.441 | 0.412 | −0.143 | <u>0.288</u> | 0.184 | 0.391 | **0.407** |
| HPSv2 | 0.166 | 0.182 | 0.473 | 0.209 | 0.034 | 0.181 | 0.177 | **0.486** | <u>0.406</u> |
| VQAScore | 0.224 | 0.238 | 0.440 | 0.110 | 0.081 | 0.145 | 0.085 | 0.105 | 0.140 |
| FGA-BLIP2 | **0.423** | <u>0.452</u> | **0.918** | <u>0.422</u> | −0.141 | 0.255 | 0.151 | −0.062 | 0.233 |
| ImageDoctor | <u>0.411</u> | **0.714** | 0.280 | **0.459** | 0.160 | 0.271 | **0.355** | <u>0.479</u> | 0.299 |
| Q-Align | 0.095 | 0.081 | 0.274 | 0.113 | **0.231** | 0.142 | 0.238 | 0.320 | 0.181 |
| Q-Insight | 0.054 | 0.090 | 0.253 | 0.037 | −0.011 | −0.106 | <u>0.243</u> | 0.086 | −0.102 |
| OmniQuality-R | 0.343 | 0.371 | <u>0.744</u> | 0.325 | <u>0.167</u> | **0.342** | 0.132 | 0.265 | 0.275 |

**Conditioning images.** Table 13 compares the same 250 conditional examples with and without conditioning images, keeping the model, evaluation instruction, decoding, grid-parsing rules, and evaluation support fixed. Aggregate gains are 0.030 in overall-score SRCC and 0.022 in $F_1$, computed before rounding.

Table 13: Paired comparison with ($\chi_K$) and without ($\chi_0$) conditioning images. Changes are computed before rounding.

| Task | $n$ | $\rho_o[\chi_K]$ | $\rho_o[\chi_0]$ | $F_1[\chi_K]$ | $F_1[\chi_0]$ | $\Delta F_1$ |
|---|---|---|---|---|---|---|
| TIE | 50 | 0.429 | 0.403 | 0.692 | 0.683 | +0.009 |
| SRIG | 50 | 0.375 | 0.378 | 0.512 | 0.503 | +0.009 |
| SRIE | 50 | 0.403 | 0.265 | 0.412 | 0.370 | +0.042 |
| MRIG | 50 | 0.509 | 0.470 | 0.545 | 0.522 | +0.023 |
| MRIE | 50 | 0.417 | 0.509 | 0.496 | 0.458 | +0.039 |
| All | 250 | 0.451 | 0.420 | 0.536 | 0.514 | +0.022 |

### D.3 EvalMuse Subset Analysis

Per-generator score correlations on the external EvalMuse 2K subset are reported in Table 14.

**Prompt-family analysis.** On the EvalMuse 2K subset, SRCC is 0.7786 on the 1,062 examples whose prompt families are absent from the EvalMuse training booster and 0.7599 on the 938 exam-

Table 14: Per-generator SRCC of VIEScore2 on the EvalMuse 2K subset.

| Generator | $n$ | SRCC | Generator | $n$ | SRCC |
|---|---|---|---|---|---|
| SD_v2.1 | 208 | 0.616 | SDXL | 190 | 0.655 |
| Kandinsky3 | 120 | 0.776 | SDXL-Lightning | 120 | 0.741 |
| Playground_v2.5 | 113 | 0.754 | PixArt-Sigma | 108 | 0.786 |
| SSD1B | 106 | 0.716 | LCM-SDXL | 106 | 0.780 |
| IF | 103 | 0.721 | SD_v1.5 | 100 | 0.630 |
| Midjourney_v6.1 | 96 | 0.841 | LCM-SSD1B | 96 | 0.746 |
| LCM-PixArt | 95 | 0.853 | SD_v1.2 | 93 | 0.818 |
| SDXL-Turbo | 93 | 0.829 | Dreamina_v2.0Pro | 93 | 0.640 |
| PixArt-alpha | 84 | 0.744 | HunyuanDiT | 29 | 0.841 |
| SD3 | 24 | 0.787 | Kolors | 23 | 0.895 |

ples whose prompt families appear in the booster. The two subsets therefore show comparable score correlation.

**EvalMuse subsets.** Table 15 compares the primary-suite and external EvalMuse subsets. Official test labels are not public.

Table 15: Score correlation on two EvalMuse subsets.

| EvalMuse slice | $n$ | VIEScore2 SRCC | FGA-BLIP2 SRCC |
|---|---|---|---|
| Primary suite | 200 | 0.803 | 0.918 |
| External 2K | 2,000 | 0.7734 | 0.910 |

### D.4 Preference Evaluation on MMRB2

We test whether the same scalar outputs support reward-model use on Multimodal Reward-Bench 2 (Hu et al., 2026), using its text-to-image and image-editing subsets ($1{,}000$ expert-annotated preference pairs each). Candidates are scored independently, and the higher score determines the preference. Ties and unparseable pairs receive $0.5$, with all pairs retained per task. VIEScore2 parses every pair, with $415$ T2I and $494$ editing ties. GPT-5.6-terra has $59$/$150$ unparseable pairs and $285$/$241$ ties on T2I/editing.

Qwen3-VL-8B uses the same evaluation instructions as VIEScore2 without fine-tuning.

Among the $\chi_K$ evaluators in Table 16, VIEScore2 ranks second in text-to-image accuracy ($0.574$), below GPT-5.6-terra ($0.614$) and above Qwen3-VL-8B ($0.566$). Its editing accuracy ($0.547$) is lower than that of GPT-5.6-terra ($0.619$) and Qwen3-VL-8B ($0.579$).

Table 16: MMRB2 preference accuracy ($1{,}000$ pairs per task). Rankings are separate for $\chi_0$ and $\chi_K$. Ties and unparseable pairs receive $0.5$.

| Method | Input | T2I | Edit |
|---|---|---|---|
| ImageReward | $\chi_0$ | 0.530 | <u>0.559</u> |
| PickScore | $\chi_0$ | **0.574** | **0.574** |
| HPSv2 | $\chi_0$ | 0.546 | 0.541 |
| VQAScore | $\chi_0$ | 0.537 | 0.554 |
| ImageDoctor | $\chi_0$ | 0.546 | 0.525 |
| Q-Align | $\chi_0$ | 0.513 | 0.523 |
| Q-Insight | $\chi_0$ | 0.492 | 0.521 |
| OmniQuality-R | $\chi_0$ | <u>0.569</u> | 0.555 |
| GPT-5.6-terra | $\chi_K$ | **0.614** | **0.619** |
| Qwen3-VL-8B judge | $\chi_K$ | 0.566 | <u>0.579</u> |
| VIEScore2 | $\chi_K$ | <u>0.574</u> | 0.547 |

# E Ablation Studies

## E.1 GRPO and Reward Design

**Transfer across benchmarks.** Table 17 compares VIEScore2 ($-$GRPO) with the final model. GRPO improves per-image grid IoU on all six benchmarks and $F_1$ on five. On AbHuman, grid IoU rises slightly while pooled $F_1$ falls.

Table 17: Localization before and after GRPO. $\Delta$ is the final model minus VIEScore2 ($-$GRPO), computed from the displayed values.

| | Grid IoU | | | $F_1$ | | |
|---|---|---|---|---|---|---|
| Dataset | VIEScore2 ($-$GRPO) | VIEScore2 | $\Delta$ | VIEScore2 ($-$GRPO) | VIEScore2 | $\Delta$ |
| RichHF | 0.268 | 0.299 | $+0.031$ | 0.428 | 0.466 | $+0.038$ |
| PAL4VST | 0.313 | 0.335 | $+0.022$ | 0.389 | 0.439 | $+0.050$ |
| AbHuman | 0.212 | 0.215 | $+0.003$ | 0.242 | 0.194 | $-0.048$ |
| HAD | 0.157 | 0.197 | $+0.040$ | 0.264 | 0.314 | $+0.050$ |
| SynthScars | 0.158 | 0.234 | $+0.076$ | 0.217 | 0.368 | $+0.151$ |
| SDG-30K | 0.052 | 0.102 | $+0.050$ | 0.193 | 0.272 | $+0.079$ |

**Reward ablation.** Table 18 compares the reward components under matched training examples, sampling budget, and optimization steps. All variants use the same initialization and primary-suite evaluation support. We fix $\lambda_{\text{dice}} = 1$ and use weights $0.3$ and $0.1$ for included score and format rewards. $-r_{\text{score}}$ and $-r_{\text{format}}$ indicate that the corresponding reward weights are set to zero. Grid parse rate is $100\%$ for every variant.

Table 18: Reward ablation on the primary suite with matched initialization, data, sampling budget, and optimization steps.

| Variant | Grid IoU | $F_1$ | Overall SRCC |
|---|---|---|---|
| VIEScore2 ($-$GRPO) | 0.296 | 0.471 | <u>0.596</u> |
| VIEScore2 ($-r_{\text{score}}, -r_{\text{format}}$) | 0.320 | 0.481 | **0.601** |
| VIEScore2 ($-r_{\text{score}}$) | <u>0.323</u> | <u>0.496</u> | <u>0.596</u> |
| VIEScore2 | **0.324** | **0.506** | **0.601** |

**Seed replication.** All three GRPO runs improve both per-image grid IoU and $F_1$ over the shared SFT initialization (Table 19), with grid-IoU gains of $0.024$–$0.029$.

Table 19: Results across three GRPO runs and additional SFT. Changes are relative to SFT epoch 4 and computed before rounding.

| Run | $F_1$ | $\text{IoU}_p^{\text{grid}}$ | $\Delta F_1$ | $\Delta\,\text{IoU}_p^{\text{grid}}$ |
|---|---|---|---|---|
| VIEScore2 ($-$GRPO) (epoch 4) | 0.471 | 0.296 | – | – |
| VIEScore2 ($-$GRPO) (epoch 5) | 0.472 | 0.297 | $+0.001$ | $+0.001$ |
| VIEScore2 (seed 42) | 0.506 | 0.324 | $+0.034$ | $+0.028$ |
| VIEScore2 (seed 2) | 0.483 | 0.320 | $+0.012$ | $+0.024$ |
| VIEScore2 (seed 3) | 0.493 | 0.325 | $+0.022$ | $+0.029$ |

**Precision–recall trade-off.** Increasing $\beta$ favors recall, while decreasing it favors precision. Table 20 compares the resulting development-set performance. We choose $\beta = 1$ for its grid IoU; the Tversky variant gives higher $F_1$ but lower grid IoU. The Tversky reward (Salehi et al., 2017) is $\text{TP}_w/(\text{TP}_w + \alpha\,\text{FP} + (1{-}\alpha)\,\text{FN}_w)$, using the cell weights in Section 3.4; $\alpha$ controls the false-positive penalty.

Table 20: Development-set precision–recall trade-offs. $\mathrm{FP}_{\mathrm{PAL}}/40$ counts false alarms on 40 clean PAL4VST images.

| Policy | $P$ | $R$ | $F_1$ | $\mathrm{IoU}_p^{\mathrm{grid}}$ | $\mathrm{FP}_{\mathrm{PAL}}/40$ |
|---|---|---|---|---|---|
| VIEScore2 ($-$GRPO) | 0.374 | 0.548 | 0.445 | 0.303 | **14** |
| GRPO $\beta = 0.5$ | **0.452** | 0.450 | 0.451 | 0.300 | 19 |
| GRPO $\beta = 1$ | 0.411 | 0.534 | 0.464 | **0.322** | 26 |
| GRPO $\beta = 2$ | 0.334 | **0.695** | 0.451 | 0.310 | 27 |
| GRPO Tversky $\alpha$=0.7 | 0.446 | 0.494 | **0.469** | 0.313 | 21 |

**Per-category accuracy and false alarms.** Table 21 separates per-category accuracy from clean-image false alarms. GRPO improves artifact F1 (0.488 to 0.493) and misalignment F1 (0.165 to 0.198), and raises both precision and recall on the primary suite (Table 10). It also lowers coverage F1 (0.330 to 0.297) and increases false alarms on clean PAL4VST images (63 to 85 of 115), while COCO false alarms remain at zero. Misalignment localization remains weak in absolute terms for both models. These costs are consistent with the development-set trade-off in Table 20 and with the clean-image limitation noted in Section 5.

Table 21: Per-category localization on RichHF (400 examples) and clean-image false alarms. Coverage evaluates "!" marks.

| Model | Artifact $F_1$ | Misalign. $F_1$ | Coverage $F_1$ | COCO FP/100 | PAL FP/115 |
|---|---|---|---|---|---|
| VIEScore2 ($-$GRPO) | 0.488 | 0.165 | 0.330 | 0 | 63 |
| VIEScore2 | 0.493 | 0.198 | 0.297 | 0 | 85 |
| SegFormer-b0 | – | – | – | 34 | 62 |

### E.2 Output Representation and Grid Resolution

**Output representation.** We compare several text-native spatial encodings using stand-alone localizers trained with matched data and compute. Results are shown in Table 22.

Table 22: Localization representations under matched training data and compute.

| Output form | $P$ | $R$ | $F_1$ | $\mathrm{IoU}_p^{\mathrm{grid}}$ |
|---|---|---|---|---|
| Dense bitmap | **0.580** | 0.099 | 0.169 | 0.116 |
| Point list | 0.250 | 0.446 | 0.320 | 0.206 |
| Bounding boxes | 0.354 | **0.671** | **0.463** | **0.280** |
| Sparse cells | 0.298 | 0.596 | 0.397 | 0.246 |

Bounding boxes achieve higher precision, recall, $F_1$, and grid IoU than sparse cells in this stand-alone comparison. Nevertheless, we use sparse cells because the same coordinates support text generation, cell-level Dice rewards, and spatial explanations without an additional box-to-grid conversion. Cells can also represent disconnected regions and separate defect categories, although this experiment does not establish an accuracy advantage for those cases.

**Grid resolution.** Table 23 examines the trade-off between spatial granularity and autoregressive sequence length. The model-free columns measure how well each grid preserves the original spatial annotation, while the trained columns compare models trained under otherwise matched settings. Model-free metrics use 933 non-empty annotations. In the table, "GT vanished" is the fraction of non-empty masks that become empty after grid conversion.

See Figure 4 in the main text for the resolution trade-off curves.

Table 23: Grid resolution, annotation fidelity, and output length. Rankings apply to trained-model results.

| | Representation ceiling (model-free) | | | Serialized target | | Trained model | |
|---|---|---|---|---|---|---|---|
| $N$ | pixel-F1 | pixel-IoU | GT vanished | tokens (mean) | tokens (p95) | F1@$N$ | pixel-IoU$_p$ |
| 4 | 0.219 | 0.178 | 68.4% | 7 | 35 | — | — |
| 8 | 0.477 | 0.374 | 24.1% | 27 | 116 | **0.425** | 0.192 |
| 12 | 0.657 | 0.535 | 9.8% | 66 | 250 | 0.398 | **0.235** |
| 16 | 0.731 | 0.615 | 6.4% | 120 | 471 | <u>0.415</u> | <u>0.231</u> |
| 24 | 0.813 | 0.712 | 2.8% | 273 | 1,089 | 0.388 | 0.209 |
| 32 | 0.857 | 0.770 | 1.5% | 486 | 1,926 | 0.366 | 0.192 |

# F Qualitative Examples and Failure Analysis

We show held-out examples with ground-truth and predicted grids. In the walkthrough and task examples, red denotes artifacts, blue misalignment, and orange single-grid defects.

**Selection for cross-benchmark comparisons.** For Figure 3, candidate images have 6–70 ground-truth defect cells. Within each benchmark, we select the image with the largest grid-IoU margin between VIEScore2 and the highest-scoring competing prediction in the selection pool. We display the same four models on every example: VIEScore2, GPT-5.6-sol, Claude Opus 5.5, and ImageDoctor. Every panel retains the ground-truth outline, so a separate input column is unnecessary. This selection highlights favorable cases; aggregate comparisons are reported in Table 1.

## F.1 Before and After GRPO

Figure 6 compares VIEScore2 ($-$GRPO) with the final VIEScore2 model. In (a), GRPO shifts the predicted cells toward the annotated defect region, increasing grid IoU from $0.00$ to $0.87$. In (b), it misses much of the annotated region, reducing IoU from $0.79$ to $0.11$. An additional SFT epoch leaves both predictions unchanged. These examples illustrate the benefits and limitations of post-training; the aggregate gains do not imply improvement on every image.

## F.2 A Complete Evaluation Walkthrough

Figure 7 shows a RichHF image, the complete model output, and the explanation from the parameter-free parser.

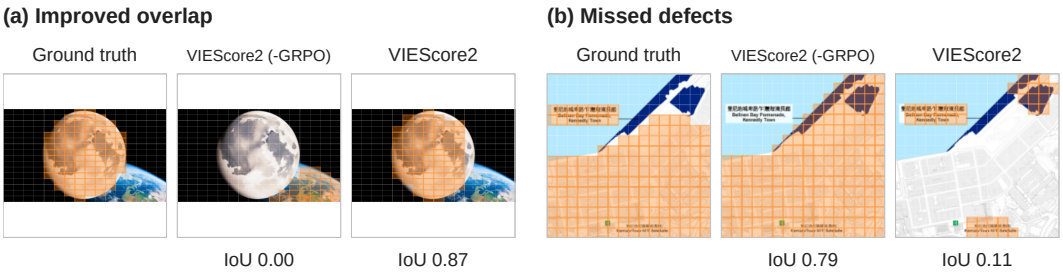



Figure 6: Localization before and after GRPO. Orange cells mark defects; values are per-image grid IoU. VIEScore2 (−GRPO) uses the SFT checkpoint; epochs 4 and 5 give identical predictions. VIEScore2 denotes the final model after GRPO.

Prompt: A king playing smash with his servants




**Model output**

```
pq: 4/10
sc: 6/10
artifact:
r2: 6!,7!,8!,9!,10!,11
r3: 3,4,12
r4: 3,4,12
r6: 5,6!,8!,9!,10!,11!,12,14
r7: 3!,4!,5!,6,8!,9!,10,11!,12!,13!,14!
r8: 3,4!,5,8!,9!,11!,12
r9: 6!,7,8!,9,11!,12
r10: 6
r15: 6!,7!,8!,9!,10!,11!
r16: 5!,6!,7!,8!,9!,10!,11!,12
misalign:
r2: 6!,7!,8!,9!,10!,11
r6: 8!,9
r7: 4!,5!,8!,9!,10!,11!,12!,13!,14!
r8: 8!,9!,10,11!,12
r9: 6!,7!,8!,9!,10!,11!
```

**Explanation from the parameter-free parser**

Perceptual quality is low (4/10), driven by extensive visual artifacts in the center (rows 6–10, columns 6–14) and extensive visual artifacts in the bottom (rows 15–16, columns 5–12) and extensive visual artifacts in the left side (rows 6–8, columns 3–6), plus 3 smaller regions. Semantic consistency is moderate (6/10), driven by extensive content that does not match the prompt in the center (rows 6–9, columns 6–14) and extensive content that does not match the prompt in the top (row 2, columns 6–11) and extensive content that does not match the prompt in the left side (row 7, columns 4–5).

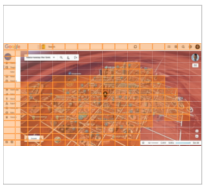
Figure 7: A RichHF example with annotations, predictions, raw output, and the parameter-free parser's explanation.

### F.3 Generation, Editing, and Reference-Conditioned Cases

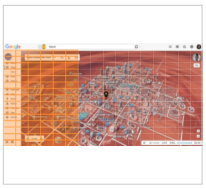
Figures 8 and 9 show the six ImagenWorld tasks with their inputs and original prompts. Panels within each task row use matched frames.

**TIG | PQ 4/10 | SC 5/10**

Ground truth



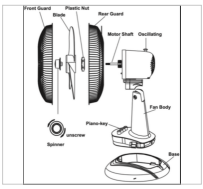
Prediction



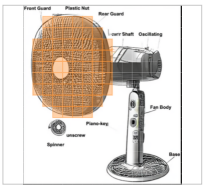
Prompt: A detailed screenshot of a futuristic Google Maps interface, showing an aerial view of the city 'New New York' on the planet Mars. The map is centered on a specific, bustling block within the city's downtown area, prominently marked with a location pin. The interface displays typical Google Maps UI elements, including a search bar, zoom controls, street names, block outlines, and various points of interest labels. The city features advanced, sci-fi architecture, such as interconnected domed structures and elevated transportation systems, all set against the distinct reddish-orange Martian landscape.

**TIE | PQ 5/10 | SC 5/10**

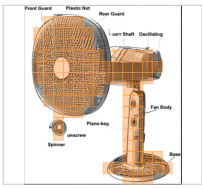
Input 1



Ground truth



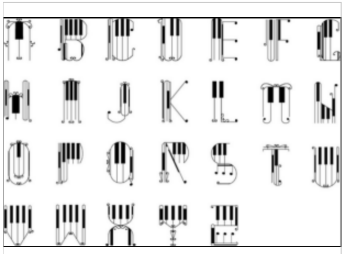
Prediction



Prompt: Edit image 1, which depicts a disassembled fan, to show the product in its fully assembled state. Transform the separate components (Front Guard, Rear Guard, Blade, Plastic Nut, Spinner, Motor Shaft, Oscillating mechanism, Fan Body, Piano-key controls, and Base) into a single, cohesive desk fan. Ensure all parts are correctly fitted together to form the complete product, with a clear and detailed appearance.

**SRIG | PQ 5/10 | SC 5/10**

Input 1



Ground truth



Prediction



Prompt: Create a minimalist line art graphic displaying the number '3.141592653589'. Each character (digits and decimal point) should be designed in black and white, incorporating elements of piano keys and musical notation, consistent with the unique typographical style exemplified by the alphabet shown in image 1. The full number should be presented horizontally.

Figure 8: TIG, TIE, and SRIG examples.

**SRIE | PQ 4/10 | SC 5/10**

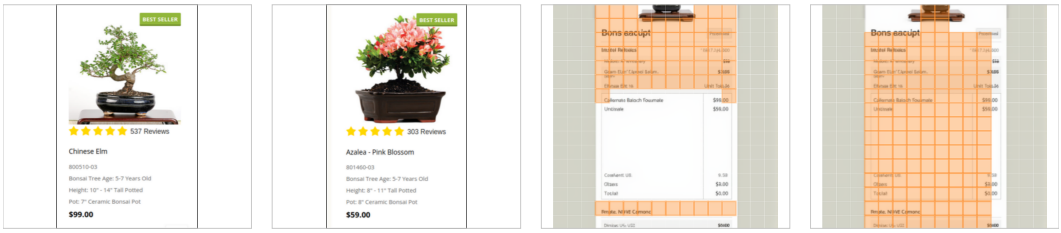
Input 1 | Input 2 | Ground truth | Prediction




Prompt: Edit the textual graphic in image 1. Warp the entire 'NOTHING CAN STOP A GOOD IDEA' text block within image 1 to follow a smooth S-curve path. Maintain the original font, color gradient, and visual texture of the text as presented in image 1. The overall bounding box of the warped text block should be scaled to 150% of its original dimensions.

**MRIG | PQ 2/10 | SC 2/10**

Input 1 | Input 2 | Ground truth | Prediction




Prompt: Generate a screenshot of a digital receipt for plant purchases from a bonsai nursery. The receipt should list two purchased items: one unit of 'Chinese Elm' at $99.00 (as depicted in image 1) and one unit of 'Azalea - Pink Blossom' at $59.00 (as depicted in image 2). The receipt should clearly display the quantity, item name, unit price, and a calculated subtotal and total for these specific items.

**MRIE | PQ 5/10 | SC 5/10**

Input 1 | Input 2 | Input 3 | Ground truth | Prediction




Prompt: Edit image 1, the 'making a Latte' process guide.

In panel 2, replace the existing milk carton, which is depicted pouring liquid into a pitcher, with a line art bottle. This new bottle should be styled like the one in image 3 and maintain the same pouring action and position.

In panel 6, add an instance of the sugar packet icon from image 2. Position this icon in the upper right area relative to the final latte cup, consistent with the overall line art style of the guide.

Figure 9: SRIE, MRIG, and MRIE examples.

### F.4 Failure Cases and Interpretation Limits

Figure 10 illustrates representation loss, missed misalignment, false alarms, and an explanation generated from an incorrect prediction.

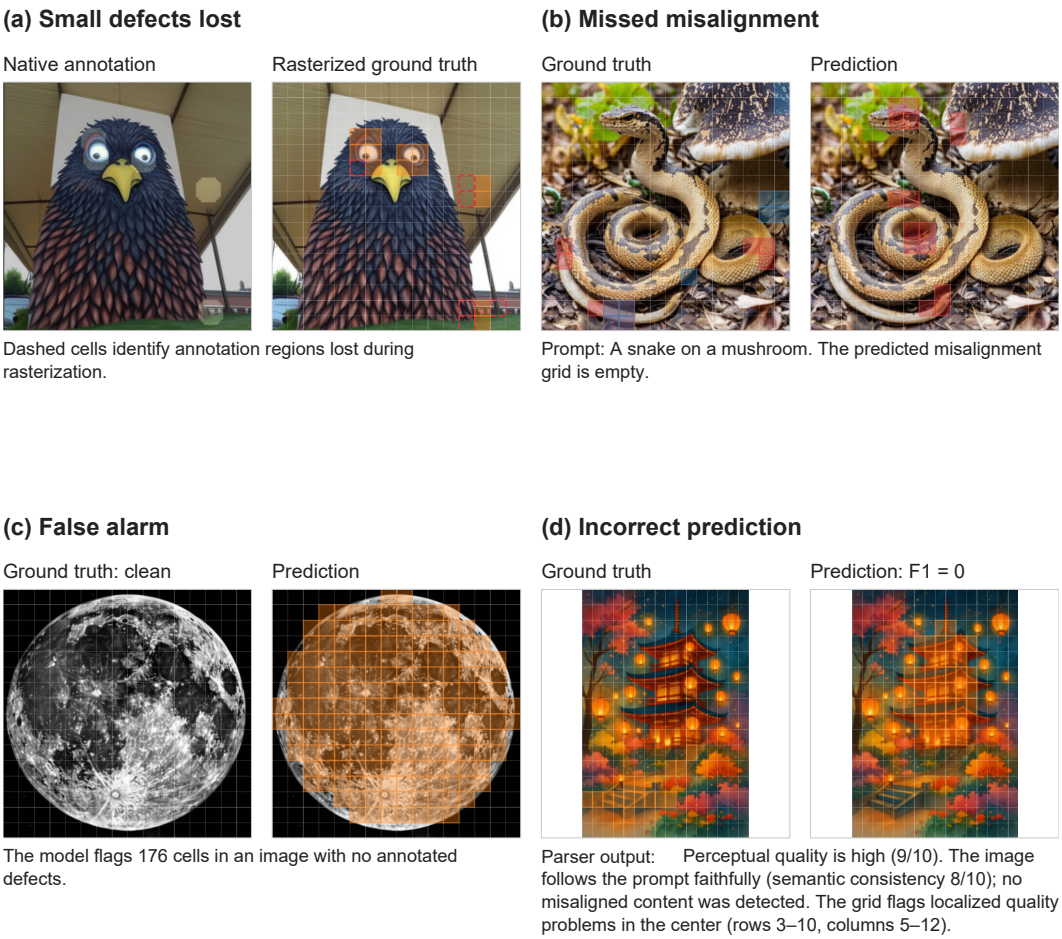



Figure 10: Failure cases: lost small defects, missed misalignment, false alarms, and an incorrect prediction with its explanation.

## G Evaluation Instruction Templates

The evaluation instructions below specify image order, requested fields, and output syntax. <PROMPT> denotes the generation or editing prompt. In the original wording, "channels" refers to defect categories, and "severe" or "severely" accompanies "!"; its supervision is annotation coverage (Appendix A.1).

**Shared system message**

```
You are an expert image quality evaluator.

You may be shown one or more INPUT images -- an original image to be
edited and/or reference images that were given to the image generator
-- followed by the GENERATED image to be evaluated.  The GENERATED
image is ALWAYS the LAST image; earlier images are referred to as
"image 1", "image 2", etc.  in reading order.  When input images are
present, judge the generated image RELATIVE to them:  whether the
requested edit was applied correctly, whether content that should
have been preserved was preserved, and whether referenced subjects
are reproduced faithfully.  When no input image is present, judge the
generated image against the text prompt alone.

The GENERATED image is divided into a 16×16 grid of cells (row 1 is
the top, column 1 is the left).  For every cell of the GENERATED
image decide whether it contains visible quality problems:
artifacts, distortions, blurriness, missing or malformed content,
color abnormalities, unrealistic content, or -- for edits -- a
failed/incorrect edit or a wrongly altered region.

FIRST output one line with the overall quality score of the generated
image in the form "score:  <0-10>/10" (10 = flawless, 0 = unusable).
THEN list ONLY the cells that HAVE problems, one line per affected
row, in the form "r<row>:  <col>,<col>,..." using 1-based indices
(rows and columns range from 1 to 16).  Emit a line only for rows
that contain at least one problematic cell, in increasing row order.
If the image has no problems at all, output exactly "none" instead of
the cell lines.

When the request asks for separate problem CHANNELS, report the cells
in two sections instead of one list:  a header line "artifact:"
followed by the rows of cells with visual artifacts (distortions,
malformed or implausible content, rendering errors), then a header
line "misalign:" followed by the rows of cells whose content
contradicts the prompt (wrong, missing or extra objects/attributes).
Within each section use the same "r<row>:  <col>,..." lines,
appending "!" to a column number when that cell's problem is severe
(e.g.  "r5:  8,9!,10").  Write "none" under a section with no
problems.

When the request asks for TWO scores instead of one, rate perceptual
quality (freedom from artifacts/distortions) as "pq:  <0-10>/10" and
semantic consistency with the prompt as "sc:  <0-10>/10", one line
each, before the cell lines.
```

**RichHF user template**

```
This generated image is divided into a 16×16 grid of equal cells.

Generation/edit prompt:  "<PROMPT>"

First rate the GENERATED image on two axes -- perceptual quality
(artifacts, distortions, rendering flaws) as "pq:  <0-10>/10" and
semantic consistency with the prompt as "sc:  <0-10>/10", one line
each -- then list the problem cells of the GENERATED image in two
channels:  an "artifact:" section for cells with visual artifacts,
then a "misalign:" section for cells whose content contradicts the
prompt.  In each section use one line per affected row as "r<row>:
<cols>", append "!" to severely affected columns, and write "none"
under an empty section.
```

**Conditional user template (two input images)**

```
The first 2 images (image 1, image 2) are the input/reference images
that were given to the generator.  The LAST image is the GENERATED
image to evaluate; it is divided into a 16×16 grid of equal cells.

Generation/edit prompt:  "<PROMPT>"

First rate the GENERATED image on two axes -- perceptual quality
(artifacts, distortions, rendering flaws) as "pq:  <0-10>/10" and
semantic consistency with the prompt as "sc:  <0-10>/10", one line
each -- then list the cells of the GENERATED image that contain
visible quality problems, one line per affected row as "r<row>:
<cols>".  Output "none" instead of cell lines if there are no
problems.
```

**Localization-only user template**

```
This generated image is divided into a 16×16 grid of equal cells.

Generation/edit prompt:  "<PROMPT>"

List the cells of the GENERATED image that contain visible quality
problems, one line per affected row as "r<row>:  <cols>".  Output
"none" instead of cell lines if there are no problems.
```

**EvalMuse score-only user template**

```
Generation prompt:  "<PROMPT>"

Give the overall quality score of the GENERATED image as "score:
<0-10>/10".
```