Visual Distortion Detection in UGC Images Using Large Multimodal Models

Ziheng Jia[1], **Yingji Liang**[2], **Jiaying Qian**[1], **Xiongkuo Min**[1*]
[1]Shanghai Jiao Tong University
[2]East China Normal University
jzhws1@sjtu.edu.cn

Abstract

The localized depiction of perceptual quality has long been a crucial, yet underexplored, challenge in image quality assessment (IQA). Existing approaches based on large multimodal models (LMMs) predominantly rely on text-driven supervised fine-tuning (SFT). However, this training paradigm exhibits notable limitations in detection accuracy. Moreover, synthetically distorted images, which are often used as the primary training data source, show a significant generalization gap when deployed in real-world scenarios; thus, the **synthetic-to-authentic (*S2A*)** problem represents a critical challenge. Motivated by these issues, we propose ***VIGIL***, which leverages the LMM architecture for precise visual distortion detection. From a candidate pool of over $1000K$ samples, we construct the ***VIGIL-140K*** training set, which consists of over $140K$ distorted images. These images are obtained through rigorous quality filtering and carefully crafted distortion injection, covering 8 major synthetic distortion categories. Our model leverages different layers of the large language model (LLM) decoder, treating them as *multiple detectors* that perform synchronous distortion detection using multi-level features. Additionally, we retain distortion cues from predictions assigned to the non-distortion class, which helps mitigate the ambiguous foreground-background (*FG-BG*) separation commonly encountered in the *S2A* problem. After post-processing, our model consistently outperforms strong baselines on both in-domain synthetic distortion detection and *S2A* tasks.

Introduction

Image quality assessment (IQA) is one of the most extensively studied topics in computer vision. Classic IQA research mostly concentrates on perceptual quality rating, where the model's output is aligned with the mean opinion score (MOS) derived from human subjective experiments (Zhai and Min 2020). With the recent proliferation of large multimodal models (LMMs), IQA research on user-generated content (UGC) images has increasingly leveraged the versatility of LMMs to enable comprehensive quality analysis (Zhang et al. 2025b,a). However, most existing works still focus on **global** quality assessment, such as providing an overall quality score (first impression) or global descriptions aligned with specific quality factors and attributes. In contrast, for images with significant **localized** artifacts, systematic detection methods are still underdeveloped. This naturally leads to the question: *Why is it necessary to perform localized distortion detection and analysis beyond global assessment for UGC images?*

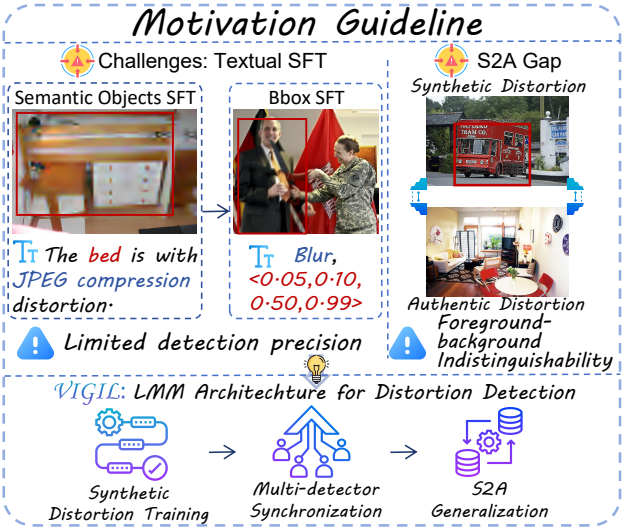

Figure 1: The limited detection precision caused by textual SFT and the *FG-BG* indistinguishability in *S2A* generalization tasks are two major challenges for existing models. To address these issues, we propose *VIGIL*. It leverages synthetic distortion training for scalable data expansion, employs multiple detectors for synchronous detection, and retains background class prediction information to improve *S2A* generalization.

First, in-the-wild UGC images are more likely to be affected by spatial local distortions caused by hardware limitations or specific capture conditions compared to other image types like AI-generated content (AIGC) or professionally generated content (PGC). Accurately identifying and localizing such distortions holds significant **practical value**. Secondly, with the continuous improvement in the perceptual quality of UGC images

*Corresponding author.

in recent years, meeting "high-quality" requirements is not merely equivalent to having a favorable global subjective impression; the focus of enhancement and post-processing has increasingly shifted towards fine-grained local details. As a result, it is becoming essential to evolve from global assessments to localized distortion detection, thereby enabling more effective **feedback-driven optimization**. Third, many UGC images deliberately introduce effects (e.g., defocus blur, color shifts) in specific regions to improve aesthetics. Detecting these localized effects facilitates in-depth annotation, which, in turn, enhances the LMM's overall **quality understanding**.

Some existing LMM-based approaches for distortion detection primarily rely on **text generation** (Chen et al. 2024c): they use supervised fine-tuning (SFT) with data labeled with regional distortion details, enabling LMMs to generate localized distortion cues. This training paradigm typically takes two forms: **semantic-object-level** descriptions, where the training label includes quality descriptions linked to local semantic objects with injected distortions (Jia et al. 2026a,b), and **bounding-box (bbox)-level** (Chen et al. 2024c), where coordinates are used directly as the SFT labels. However, in real-world scenarios, the locally distorted regions are challenging to define using explicit semantic objects. Furthermore, directly generating distortion coordinates without a clear matching and classification process can significantly reduce **detection accuracy**. These factors collectively constrain the localized distortion detection capability in existing models.

Another significant challenge is **generalization in real-world scenarios**. While synthesizing distortions efficiently scales training datasets, the injected distortions, which typically follow regular shapes and uniform spatial intensity, differ significantly from authentic distortions in *foreground (FG)-background (BG)* distinguishability. Consequently, the issue of generalizing from synthetic to authentic distortions (***S2A***) requires careful consideration.

In response to the challenges, we propose **VIGIL**, which enables accurate detection of localized **vi**sual distortions in U**G**C **i**mages using the **L**MM architecture. Rather than relying on SFT, we revert to the classical object detection paradigm, leveraging LMM's powerful representation learning and regression capabilities. To accelerate convergence and improve the model's detection capability, we utilize multiple layers of the large language model (LLM) part for synchronous detection. To mitigate the *FG-BG indistinguishability* in *S2A* tasks, we propose a simple yet effective technique that reuses detection cues from boxes predicted as non-distortion. The overview of our work is shown in Fig. 1.

Our main contributions are summarized as follows:

1. We construct the **VIGIL-140K** dataset. From the source pool of over $1000K$ images, we filter those suitable for distortion injection, perform region-level distortion combination, and finally obtain a large-scale training set with over $200K$ distortion area labels.
2. We introduce **VIGIL-8B**, an LMM-based model designed for perceiving and detecting local visual distortions. The model employs multiple detector synchronization, retains informative signals from boxes classified as background, and incorporates post-processing to refine the final outputs.
3. Our model achieves state-of-the-art (SOTA) performance in both in-domain synthetic distortion and **out-of-domain (OOD) authentic distortion detection** tasks.

Source	# Images	# AF	# AD	# Dist. Areas
Online	$600K$	28,369	8,600	10,353
CLIP-Pretrain	$300K$	68,132	54,940	75,559
COCO	$110K$	95,768	80,700	117,706
Unsplash	$10K$	9,783	1,100	2,254
Total	$1028K$	202,052	145,340	205,872

Table 1: Statistic summary of *VIGIL-140K*. "AF" means after the quality filtering. "AD" indicates after the distortion combination process.

Related Works

Object Detection Models

Object detection is a classic problem in computer vision. *Fast R-CNN* (Girshick 2015) computes convolutional features and applies *RoI pooling* to obtain region features for classification and bbox regression. *Faster R-CNN* (Ren et al. 2016) introduces the *region proposal network (RPN)*, which shares backbone features with the detection model, enabling two-stage detection without the need for external anchors. *Detection Transformer (DETR)* (Carion et al. 2020) reformulates detection as a set prediction bipartite matching problem, removing the need for post-processing. *Deformable DETR* (Zhu et al. 2021) adopts multi-scale *deformable attention* that samples points around reference locations, accelerating convergence. *YOLO-series* (Redmon et al. 2016) is a real-time object detection model series that processes the image in a single pass. *DINO* (Zhang et al. 2023a), and *Grounding-DINO* (Liu et al. 2024) enhance *DETR*-style detectors through improved query design and denoising-based training supervision.

Image Local Distortion Detection

Q-Ground (Chen et al. 2024b) leverages LMMs with the grounding-training paradigm for image distortion segmentation. *Grounding-IQA* (Chen et al. 2024c) distills location-relevant information from existing human-annotated SFT data (Wu et al. 2024a). During detection, it adopts a training-inference scheme where the model directly outputs bbox coordinates. *Refine-IQA-S1* (Jia et al. 2026a) employs the rule-based reinforcement learning strategy, using the *Intersection over Union (IOU)* between the target boxes and the predicted

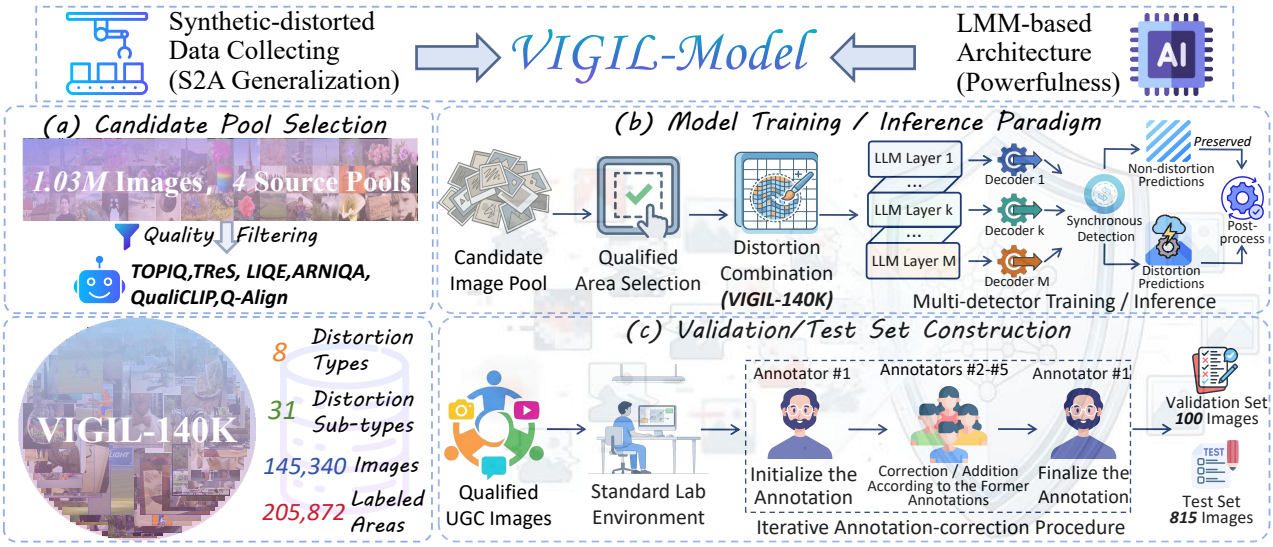

Figure 2: The main workflow of *VIGIL*.

boxes as the reward, while still relying on textual coordinate output during inference. *ViDA-UGC* (Liao et al. 2025) constructs a human-annotated dataset covering distortion detection, grounding, and description, and conducts *chain-of-thought (CoT)*-based SFT. However, these methods still exhibit limitations. First, they rely heavily on labor-intensive human annotation, with a relatively narrow range of distortion categories, and fail to fully achieve data scaling and explore the *S2A* generalization. Second, the majority of these approaches depend on text generation, which limits both the accuracy and robustness of the localization process.

The VIGIL

In response to the above issues, the *VIGIL* is trained entirely using synthetic distorted images. Its goal is to achieve high reliability in synthetic distortion detection while maximizing its *S2A generalization* capability in UGC scenarios. The overall workflow of *VIGIL* is demonstrated in Fig. 2.

Training Data Preparation

Synthetic distortions are applied to selected regions of high-quality images, maximizing diversity and randomness in both their locations and types.

Candidate Image Pool Selection We select diverse UGC images as the source for our training dataset. Specifically, we choose *COCO-2017-train* (Lin et al. 2014)(containing $110K$ samples), a subset of *CLIP* (Radford et al. 2021) pretraining data (approximately $300K$ images), short video screenshots crawled from streaming platforms ($600K$ samples), and $10K$ high-quality aesthetic images from *Unsplash*. These datasets span a wide range of semantic content, making them suitable for robust model training and *S2A* generalization.

To prevent the influence of pre-existing distortions, we apply quality filtering. We use 6 no-reference IQA models (*TOPIQ-NR* (Chen et al. 2024a), *TReS* (Golestaneh, Dadsetan, and Kitani 2022), *LIQE* (Zhang et al. 2023b), *ARNIQA* (Agnolucci et al. 2024), *QualiCLIP* (Agnolucci, Galteri, and Bertini 2024), and *Q-Align* (Wu et al. 2024b)) (pretrained on *KonIQ-10K* (Hosu et al. 2020)) to score the entire source dataset. After normalizing them to a $[0, 100)$ scale, we select data with scores above 85. As these IQA methods for regular UGC images do not apply to the ultra-high-definition *Unsplash* images, we manually check this part. We use a 4K resolution display device to present these images at their original resolution. Images with low quality or those that have undergone noticeable aesthetic post-processing (e.g., background defocusing) are excluded.

Distortion Combination To ensure local distortions are perceptible, we further filter randomly picked local regions by **sharpness**, **colorfulness**, and **brightness**, retaining only those that satisfy the criteria (see supplementary materials (*Supp.*)). Following the commonly used *KADIS-700K* (Lin, Hosu, and Saupe 2019), we consider 10 common distortion types: "BLUR", "NOISE", "COMPRESSION", "OVEREXPOSURE", "CONTRAST STRENGTHEN", "UNDEREXPOSURE", "CONTRAST WEAKEN", "SATURATE STRENGTHEN", "SATURATE WEAKEN", and "OVERSHARPEN". Based on perceptual similarity, we merge "OVEREXPOSURE" with "CONTRAST STRENGTHEN" and "UNDEREXPOSURE" with "CONTRAST WEAKEN", yielding 8 major categories (with 31 sub-types detailed in *Supp.*). For each category, we apply 2 severity levels, "NOTICEABLE" and "SEVERE". Details of the distortion combination strategy are provided in Algorithm 1 and *Supp.*. For each image where local distortions are successfully added, we record the distortion category and the bbox of each modified region.

The statistical information of the *VIGIL-140K* is shown in Tab. 1 and also detailed in *Supp.*.

Algorithm 1: Distortion Combination Process (per image)

$K \leftarrow$ random integer between 1 and 3
$count \leftarrow 0$
while $count < K$ **do**
    $rect\_area \leftarrow$ randomly select rectangle area between $\frac{1}{20}$ and 1 of the image
    $distortion \leftarrow$ randomly select one distortion
    $metrics \leftarrow$ calculate metrics for the selected region
    if *rect_area*: *clarity*, *brightness*, and *colorfulness metrics* are qualified **then**
        Add *distortion*, record *type / bbox*
        $count \leftarrow count + 1$
    else
        Skip this region
        $count \leftarrow count + 1$
    end if
end while

Model Structure Design

LMMs typically exhibit strong regression capacity. Building on this property, we repurpose the LMM architecture as a powerful encoder to comprehensively capture low-level perceptual image features. We further exploit multiple layers from the LLM part as parallel detectors for synchronous detection. Finally, the detections for each image are produced after post-processing during the inference stage. The detailed model structure is shown in Fig. 3.

Following *DETR* (Carion et al. 2020), we first encode the image to obtain a sequence of image tokens, and then append N **learnable area query tokens**. In designing the attention mask, we apply causal attention to each image token while using global attention for all area query tokens. We choose M layers from the LLM part, each serving as an **independent** detector. We attach two linear heads for classification (9 classes, including the non-distortion class) and box regression after each selected layer, enabling up to $M \times N$ predictions. As these detectors operate independently, we compute the matching cost for each detector's predictions and apply *Hungarian Matching* (Kuhn 1955) separately. This design is motivated by two considerations. First, visual distortion detection largely relies on **low-level cues**, which makes representations from the early stages of the LLM decoder potentially more informative. Secondly, leveraging multiple detectors for synchronous detection can be viewed as integrating complementary **perceptual perspectives**. As a result, optimizing detection at earlier stages expands the pool of candidate predictions, potentially reducing missed detections. Compared to simply increasing the number of learnable area query tokens, this approach also accelerates convergence.

Specifically, let y denote the ground-truth set of distorted areas and $\hat{y}_k = \{\hat{y}_{ki}\}_{i=1}^{N}$ the set of N predictions of the *k-th* detector. We treat y as a set of size N, padded with $\varnothing$ (representing the non-distortion class). For the *k-th* detector, the *Hungarian Matching* is represented as:

$$\hat{\sigma_k} = \underset{\sigma \in \mathfrak{S}_N}{\arg\min} \sum_{i=1}^{N} \mathcal{L}_{\text{match}} \left(y_i, \hat{y}_{k\sigma(i)}\right), \tag{1}$$

where $\mathcal{L}_{\text{match}} \left(y_i, \hat{y}_{k\sigma(i)}\right)$ is a pair-wise matching cost between ground truth y_i and a prediction with its index (mapping) $\sigma(i)$ in the *k-th* detector's predicted set. The matching cost accounts for both classification confidence and the alignment between the predicted bbox and the ground truth. Each ground-truth element i is represented as $y_i = (c_i, b_i)$, where c_i denotes the target class label (possibly $\varnothing$) and $b_i \in [0,1]^4$ specifies the ground-truth bbox in "XYXY" format, normalized by the image size. For the prediction indexed by $\sigma_k(i)$, we denote the predicted probability of class c_i as $\hat{p}_{\sigma_k(i)}(c_i)$ and the predicted box as $\hat{b}_{k\sigma_k(i)}$. Under these definitions, the cost associated with matching y_i to $\sigma_k(i)$ (which may not be the optimal one) is given by:

$$\begin{aligned} \mathcal{L}_{\text{match}} \left(y_i, \hat{y}_{k\sigma_k(i)}\right) = & -\mathbb{1}_{\{c_i \neq \varnothing\}} \hat{p}_{k\sigma_k(i)}(c_i) \\ & + \mathbb{1}_{\{c_i \neq \varnothing\}} \mathcal{L}_{\text{box}} \left(b_i, \hat{b}_{k\sigma_k(i)}\right). \end{aligned} \tag{2}$$

while $\mathcal{L}_{\text{box}}$ is denoted as:

$$\mathcal{L}_{\text{box}} = \lambda_{\text{giou}} \mathcal{L}_{\text{giou}} \left(b_i, \hat{b}_{k\sigma_k(i)}\right) + \lambda_{\text{L1}} \left\| b_i - \hat{b}_{k\sigma_k(i)} \right\|_1. \tag{3}$$

Here we set $\lambda_{\text{giou}} = 2$ and $\lambda_{\text{L1}} = 5$. The final detection loss is the average of all detector losses after bipartite matching :

$$\begin{aligned} \mathcal{L}(y, \hat{y}) = \frac{1}{M} \frac{1}{N} \sum_{k=1}^{M} \sum_{i=1}^{N} \Big[& -\log \hat{p}_{k\hat{\sigma}_k(i)}(c_i) \\ & + \mathbb{1}_{\{c_i \neq \varnothing\}} \mathcal{L}_{\text{box}} \left(b_i, \hat{b}_{k\hat{\sigma}_k(i)}\right) \Big]. \end{aligned} \tag{4}$$

In practice, when $c_i = \varnothing$, we down-weight the corresponding log-probability term by a factor of 0.05 for data balancing.

S2A Generalization Trick and Post-processing

Compared with synthetic distortions, manually localizing visual distortions in real-world UGC images is substantially more costly and less standardized. In particular, **local distortion regions are often irregular and non-uniform in shape and intensity**, which blurs the boundary between *FG* and *BG*. Under such conditions, detection models tend to incur elevated false negatives. To alleviate this issue, we apply an *S2A* generalization trick, avoiding discarding predictions assigned to the non-distorted class. Instead, we take the distortion category with the **second-highest classification probability** and its associated bbox as a valid predicted

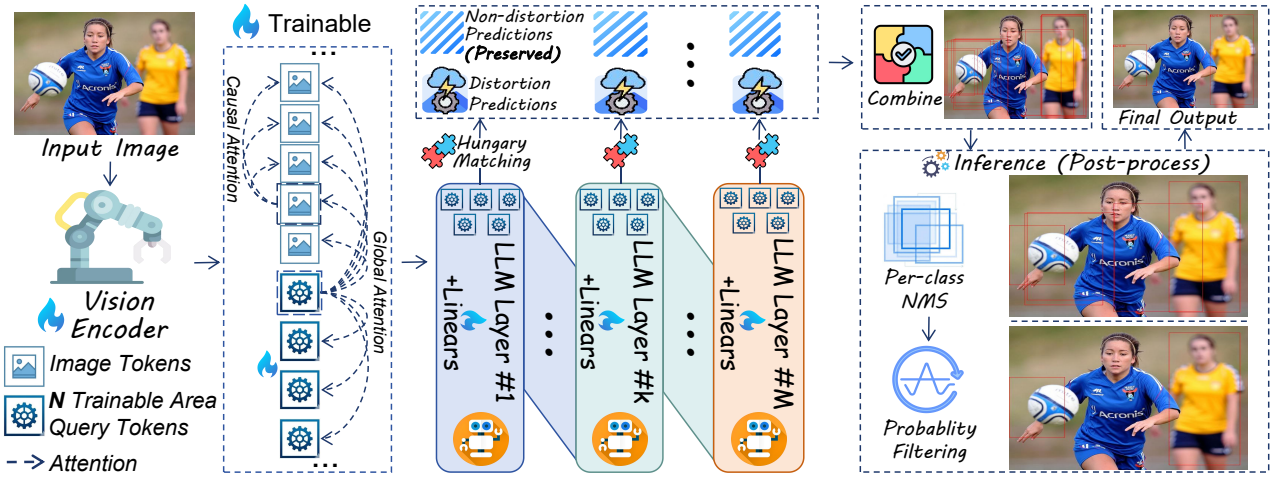

Figure 3: The model structure of *VIGIL*.

candidate. This strategy enlarges the candidate set and reduces errors caused by ambiguous *FG-BG* separation.

In our model, effective detection further relies on post-processing. For all candidate boxes produced, we assume equal opinion weights; this allows **probabilities from different detectors to be compared directly**. We first apply class-wise *non-maximum suppression (NMS)* with the *IoU* threshold T_{NMS}. We then discard candidates with probability below T_{Prob}. The remaining boxes are taken as the outputs.

Validation / Test Set Construction

We evaluate the model on two primary tasks: the in-domain synthetic distortion detection task and, **more critically**, the OOD generalization task on authentic distortion localization.

For in-domain evaluation, we generate 3000 synthetically distorted images from the undistorted images in *KADIS-700K*. We further include 1600 images from *COCO-2017-test* and *Unsplash* that satisfy the quality criteria, using the same distortion synthesis pipeline.

For the OOD task, we pick images from *COCO-2017-val* and *COCO-2017-test*. Since consistent annotation of authentic distortions among multiple annotators is challenging without reference opinions, we implement an **iterative annotation-correction procedure**: The first (leader) annotator marks perceptually obvious distortion regions in the image, each belonging to one of the 8 major categories. Annotators *2-5* then review the marked regions in turn; they may *revise* or *remove* regions that are not perceptually valid and *add regions* they consider missed. Finally, the leader annotator performs a last-pass check, during which only deletions are allowed.

All annotations are produced using *LabelMe*. A region is annotated only if the distortion is perceptually salient, and the four-connected distortion area is expected to align reasonably with the annotated bbox. Each image contains $1-3$ distorted regions, allowing overlaps. The final dataset consists of 915 human-labeled UGC images. We reserve 100 images for validation, and use the remaining 815 for testing.

Training Details

We use *InternVL-3-8B-Instruct* (Zhu et al. 2025) (LLM: *Qwen2.5-7B* (Qwen Team 2024)) (with added learnable area query tokens and the linear classifiers and regressors) as the base model. The number of epochs is set to 1, with checkpoint saving every 500 step. Each saved checkpoint is evaluated on the validation set, and the *AP50* metric of all labeled regions in the validation set is recorded. The checkpoint with the highest *AP50* is chosen as the final model for the selected hyperparameter setting. More detailed model structure and hyperparameter settings are presented in the *Supp.*.

Experiments

To thoroughly evaluate the performance of our model, we conduct comparative experiments against multiple strong baselines on both in-domain synthetic distortion detection and OOD authentic distortion detection tasks. In addition, to analyze the impact of key attributes on model performance, we perform comprehensive ablation studies accompanied by detailed discussions and analyses.

Experiments Settings and Main Results

Both synthetic and authentic distortion detection are evaluated on the respective test sets mentioned in *Validation / Test Set Construction* above. After a rigourous ablation study, we set $N = 3$ and $M = 5$ (using the $10th$, $15th$, $20th$, $25th$, and the final ($28th$) layers of the LLM). In the post-processing stage, we set $T_{\text{NMS}} = 0.3$, $T_{\text{Prob}} = 0.1$ for authentic distortion detection, and $T_{\text{NMS}} = 0.7$,$T_{\text{Prob}} = 0.8$ for synthetic distortion localization. In both experiments, we compare our method against comprehensive strong baselines. Since many general LMMs already support direct region localization, we

Dist. Type	Blur		Noise		Comp.		OE		UE		OS		US		OSH			
# of GT Areas	1,619		1,768		1,067		1,640		1,647		289		315		818		*mAP*↑	*AP50*↑
Models	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑		
General LMMs																		
InternVL3.5-8B	0.316	0.194	0.336	0.228	0.332	0.190	0.323	0.206	0.344	0.249	0.372	0.230	0.342	0.236	0.329	0.206	0.274	0.326
InternVL3.5-14B	0.381	0.220	0.412	0.263	0.364	0.224	0.400	0.256	0.419	0.281	0.366	0.284	0.484	0.188	0.393	0.249	0.322	0.389
InternVL3.5-38B	0.461	0.237	0.462	0.315	0.469	0.254	0.448	0.270	0.478	0.335	0.520	0.269	0.520	0.312	0.475	0.292	0.369	0.461
Qwen3VL-8B	0.298	0.171	0.319	0.224	0.316	0.177	0.304	0.194	0.326	0.228	0.352	0.189	0.340	0.189	0.329	0.178	0.259	0.313
Qwen3VL-32B	0.448	0.256	0.463	0.291	0.449	0.243	0.411	0.264	0.447	0.321	0.457	0.293	0.462	0.293	0.430	0.253	0.350	0.437
LLaVA-OV-1.5-7B	0.301	0.159	0.305	0.203	0.306	0.191	0.266	0.186	0.320	0.231	0.318	0.213	0.304	0.170	0.319	0.195	0.249	0.297
GPT-4o (24-11-20)	0.340	0.174	0.359	0.220	0.319	0.207	0.346	0.231	0.363	0.248	0.358	0.216	0.314	0.222	0.325	0.231	0.284	0.340
GPT-5 (25-08-07)	0.312	0.170	0.313	0.197	0.309	0.171	0.306	0.217	0.316	0.214	0.309	0.228	0.299	0.204	0.336	0.196	0.258	0.308
Gemini-3.0-Pro	0.449	0.231	0.447	0.265	0.489	0.259	0.423	0.276	0.426	0.335	0.481	0.334	0.478	0.318	0.435	0.272	0.360	0.438
Detection Models																		
Faster-RCNN-R50	0.532	0.297	0.562	0.342	0.510	0.304	0.539	0.318	0.541	0.398	0.553	0.305	0.534	0.354	0.502	0.292	0.418	0.528
DETR-R50	0.706	0.390	0.747	0.484	0.734	0.397	0.692	0.479	0.737	0.511	0.748	0.471	0.731	0.459	0.723	0.456	0.601	0.719
Deformable-DETR-R50	0.675	0.383	0.717	0.472	0.715	0.421	0.693	0.439	0.721	0.519	0.685	0.489	0.680	0.426	0.671	0.428	0.574	0.690
YOLO-V11	**0.865**	0.486	0.902	0.581	**0.882**	0.502	**0.867**	0.546	0.906	0.628	0.882	0.553	0.909	0.560	0.853	0.526	0.708	0.878
Grounding-DINO-R50	0.829	0.488	0.874	0.558	0.855	0.492	0.848	0.535	0.855	0.603	0.854	0.554	0.881	0.558	0.812	0.512	0.687	0.851
In-domain LMMs																		
Refine-IQA-S1	0.636	0.374	0.694	0.443	0.634	0.377	0.635	0.420	0.688	0.442	0.651	0.424	0.733	0.459	0.626	0.385	0.540	0.652
VIGIL-8B	0.845	**0.501**	**0.910**	**0.603**	0.869	**0.524**	0.866	**0.578**	**0.915**	**0.657**	**0.888**	**0.583**	**0.931**	**0.597**	**0.868**	**0.555**	**0.736**	**0.879**

Table 2: Performance on the *synthetic distortion detection* task. *"Comp."*, *"OE"*, *"UE"*, *"OS"*, *"US"*, and *"OSH"* represent *"Compression"*, *"Overexposure"*, *"Underexposure"*, *"Oversaturate"*, *"Undersaturate"*, and *"Oversharpen"* respectively. Detection models have been trained on *VIGIL-140K*. "*R50*" denotes that the backbone of this model is *ResNet-50*. [Per column: highest values are in **bold**, and second-highest values are underlined.]

Dist. Type	Blur		Noise		Comp.		OE		UE		OS		US		OSH			
# of GT Areas	464		69		52		337		168		47		93		51		*mAP*↑	*AP50*↑
Models	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑	*AP50*↑	*AP75*↑		
General LMMs																		
InternVL3.5-8B	0.188	0.136	0.111	0.095	0.047	0.033	0.087	0.104	0.080	0.055	0.242	0.101	0.207	0.179	0.135	0.155	0.137	0.128
InternVL3.5-14B	0.223	0.181	0.124	0.112	0.099	0.108	0.209	0.107	0.062	0.066	0.267	0.253	0.256	0.220	0.232	0.291	0.184	0.182
InternVL3.5-38B	0.191	0.170	0.133	0.053	0.064	0.084	0.146	0.128	0.115	0.051	0.377	0.147	0.158	0.249	0.165	0.226	0.169	0.153
Qwen3VL-8B	0.153	0.134	0.095	0.046	0.075	0.048	0.110	0.098	0.067	0.079	0.090	0.115	0.154	0.184	0.197	0.125	0.118	0.115
Qwen3VL-32B	0.156	0.121	0.067	0.064	0.019	0.124	0.125	0.095	0.067	0.073	0.172	0.135	0.293	0.197	0.329	0.142	0.154	0.131
LLaVA-OV-1.5-7B	0.143	0.138	0.111	0.140	0.109	0.056	0.139	0.125	0.061	0.064	0.199	0.135	0.277	0.235	0.128	0.209	0.146	0.129
GPT-4o (24-11-20)	0.174	0.136	0.074	0.121	0.100	0.072	0.133	0.095	0.101	0.049	0.190	0.128	0.220	0.174	0.199	0.135	0.149	0.141
GPT-5 (25-08-07)	0.150	0.104	0.068	0.068	0.032	0.044	0.118	0.068	0.034	0.050	0.157	0.124	0.192	0.090	0.126	0.106	0.110	0.110
Gemini-3.0-Pro	0.262	0.279	0.160	0.183	0.168	0.143	0.190	0.187	0.150	0.133	0.299	0.218	0.408	0.352	0.268	0.239	0.238	0.220
Detection Models																		
Faster-RCNN-R50	0.230	0.191	0.160	0.124	0.142	0.076	0.179	0.188	0.124	0.102	0.214	0.233	0.338	0.280	0.190	0.287	0.197	0.192
DETR-R50	0.364	0.360	0.278	0.241	0.161	0.127	0.242	0.219	0.179	0.171	0.429	0.219	0.464	0.477	0.364	0.256	0.310	0.293
Deformable-DETR-R50	0.416	0.329	0.218	0.232	0.190	0.132	0.237	0.235	0.153	0.139	0.330	0.322	0.497	0.461	0.360	0.458	0.300	0.302
YOLO-V11	0.474	0.389	0.305	0.230	0.177	0.178	0.328	0.271	0.197	0.180	0.434	0.418	0.540	0.602	0.334	0.449	0.349	0.366
Grounding-DINO-R50	0.469	0.428	0.335	0.287	0.215	0.140	0.310	0.311	0.172	0.188	0.450	0.320	0.538	0.549	0.411	0.466	0.362	0.362
In-domain LMMs																		
Refine-IQA-S1	0.321	0.270	0.131	0.117	0.113	0.115	0.231	0.200	0.133	0.131	0.389	0.258	0.452	0.326	0.260	0.317	0.254	0.250
VIGIL-8B	**0.617**	**0.545**	**0.420**	**0.323**	**0.237**	**0.228**	**0.444**	**0.374**	**0.275**	**0.241**	**0.596**	**0.483**	**0.800**	**0.742**	**0.597**	**0.594**	**0.482**	**0.502**

Table 3: Performance on the *authentic distortion detection* task.

include representative models from the newest *Qwen3-VL* (Bai et al. 2025) series, *InternVL-3.5* (Wang et al. 2025) series, and *LLaVA-Onevision-1.5* (An et al. 2025), as well as the proprietary *GPT* (OpenAI 2024, 2025) and *Gemini* (Google DeepMind 2026). For these models, we formulate the input prompt as follows: *"Please specify the exact area of distortions in the image using coordinates in the* 'XYXY' *format (the top-left and bottom-right corners), normalized by the image height and width (*$[0, 1)$*). The distortion types include: [eight major distortion types]. The output format should be [category_coordinates / category_coordinates/...], and you may only provide up to* 1 *to* 3 *predicted distortion areas."* . During evaluation, we directly select all predicted types and regions (without post-processing) after the greedy search text generation process for reproduction.

Additionally, we compare our method against object detection models. These models are also trained on *VIGIL-140K* and evaluated using their default inference protocols. For in-domain distortion detection baselines, since most of these methods have certain limitations in terms of **task types** and **open-source availability**, which prevent a fair comparison with our model (we provide justification in *Supp.*), we include only *Refine-IQA-S1* for comparison. We report *AP50* and *AP75* for each major distortion type, along with their *mAP*. We also provide *AP50* and *AP75* on all the labeled areas.

From the results in Tabs. 2 (synthetic) and 3 (au-

thentic), it is evident that general LMMs, which have not been specifically trained, perform poorly in both synthetic and *S2A* tasks, especially in the latter. This indicates that text-generation-based methods may lack the accuracy and generalizability required for pixel-level perception-based localization tasks. When compared to other object detection models, **VIGIL-8B** demonstrates a clear performance advantage, particularly in the *S2A* task. This underscores the strengths of LMM-based detection in terms of accuracy and reliability, as well as the improved *S2A* generalization capability achieved through the application of *S2A* generalization techniques.

Discussions

Ablation Study First, we compare our training setup with the text-generation setup. For the latter, we rephrase the distortion regions in each training sample as a sequence of "DISTORTION TYPE / x_1, y_1, x_2, y_2" format and implement SFT. We record the *mAP*, *AP50*, and *AP75* metrics on both synthetic and authentic tasks. Results are shown in Tab. 4. The performance of the text-generation-based approach is significantly inferior to our settings on both tasks. We propose several reasons: first, the lack of effective matching processes in text generation, and second, training the box regression task as a token classification problem, which compromises detection accuracy. Moreover, the number of detectable areas in the text generation approach is constrained by the training data, limiting the model's generalization capability for detecting multiple distortion regions.

Next, we perform the ablation on the multi-detector setups. First, we retain all other training settings unchanged and utilize only the last LLM layer, referring to this model version as "*Last*". Additionally, we expand $N = 15$ (equal to the areas predictable in our model) while still using only the last LLM layer, which we refer to as "*Last**". We also record the loss convergence and the *mAP* on the validation set during training for our setting and the "*Last**" setting (shown in Fig. 4). Furthermore, we retain the multi-detector setup, but unify the regression and classification linear modules attached to each detector with identical parameters, referred to as "*Unified*". Results are recorded in Tab. 5. The multi-detector setup significantly improves the detection accuracy of the *S2A* task compared to using only the final layer. Additionally, it converges faster during training and performs better than simply increasing the number of query tokens. Furthermore, using different decoder layers for each detector, paired with independent classification and regression heads, also positively impacts detection performance.

We also perform the *S2A* generalization technique ablation. We retain all hyperparameters and settings but discard non-distortion class predictions. The ablation results are recorded in Tab. 6. The *S2A* technique enhances the model's generalization ability in real-world scenarios.

Additionally, we perform detailed ablation studies on various hyperparameter settings. First, we explore different combinations of M and N with results shown in Tab. 7. We also conduct an ablation analysis on the selection of $M = 5$ detector positions. For $M = 5$, we test two configurations for comparison: early layers ("*Early*"), selecting layers 2, 4, 6, 8, and 10; last layers ("*Last*"), selecting the final 5 layers. The results are presented in Tab. 8. When the number of M and N is equal to or slightly greater than the maximum number of target regions in the training images, the detection model achieves better performance. Furthermore, selecting LLM layers in a uniformly distributed order by layer number (our setting) is most beneficial for improving performance.

Version	Synthetic			Authentic		
	mAP↑	*AP50*↑	*AP75*↑	*mAP*↑	*AP50*↑	*AP75*↑
Ours	**0.736**	**0.879**	**0.574**	**0.482**	**0.502**	**0.439**
Text	0.457	0.432	0.263	0.231	0.225	0.102

Table 4: Training strategies ablation. [Per column: highest values are in **bold**.]

Version	Synthetic			Authentic		
	mAP↑	*AP50*↑	*AP75*↑	*mAP*↑	*AP50*↑	*AP75*↑
Ours	0.736	0.879	0.574	**0.482**	**0.502**	**0.439**
Last	**0.757**	**0.913**	**0.582**	0.397	0.408	0.303
*Last**	0.743	0.885	0.569	0.413	0.420	0.365
Unified	0.735	0.871	0.562	0.440	0.451	0.373

Table 5: Ablation study on different multi-detector settings.

Version	Synthetic			Authentic		
	mAP↑	*AP50*↑	*AP75*↑	*mAP*↑	*AP50*↑	*AP75*↑
Ours	**0.736**	0.879	**0.574**	**0.482**	**0.502**	**0.439**
w / o S2A	0.727	**0.882**	0.571	0.407	0.416	0.351

Table 6: Ablation study on *S2A* generalization tricks.

Conclusion

We propose the **VIGIL**, an LMM-based framework for spatial visual distortion detection. We construct the **VIGIL-140K** through rigorous visual quality filtering and spatial distortion combination strategies within a source image pool of over $1000K$ samples. During training, we utilize different LLM layers synchronous detection, thereby fully leveraging the hierarchically enhanced image features. In the inference stage, we retain the features of areas predicted as non-distortion classes to alleviate the challenge of *FG-BG* ambiguity in *S2A* generalization. The *VIGIL-8B* demonstrates excellent performance in both synthetic distortion detection and OOD *S2A* tasks. Our work provides compelling insights into fine-grained local characterization of perceptual visual quality.

References

Agnolucci, L.; Galteri, L.; and Bertini, M. 2024. Quality-aware image-text alignment for real-world image quality

Setting	Synthetic			Authentic		
	mAP↑	*AP50*↑	*AP75*↑	*mAP*↑	*AP50*↑	*AP75*↑
M=1,N=3	**0.757**	**0.913**	**0.582**	0.397	0.408	0.303
M=3,N=3	0.728	0.870	0.567	0.465	0.491	0.425
M=5,N=1	0.745	0.752	0.501	0.352	0.369	0.315
M=5,N=3	0.736	0.879	0.574	**0.482**	**0.502**	**0.439**
M=5,N=5	0.734	0.868	0.566	0.474	0.496	0.413
M=5,N=10	0.732	0.868	0.560	0.465	0.493	0.422

Table 7: Ablation on different settings on M, N pairs. The setting in **bold** is our primary setting. When $M = 3$, we select the $10th$, $20th$, and the final layers; when $M = 1$, we select the final layer.

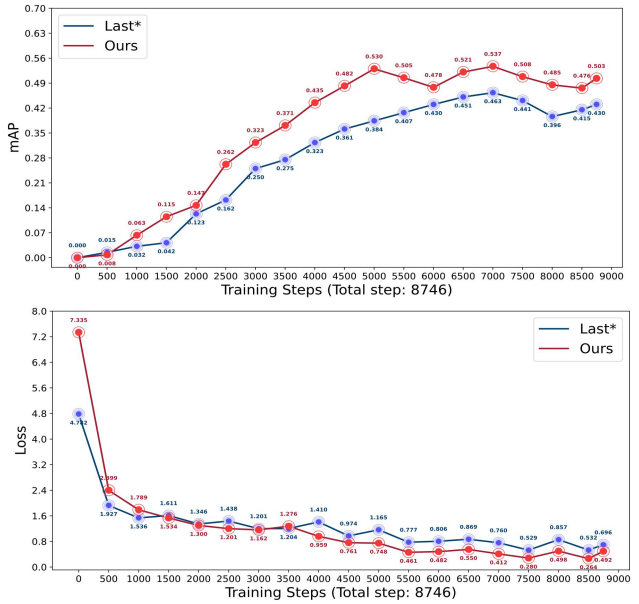

Figure 4: Curves of the variation of validation *mAP* and training loss over the course of training.

Version	Synthetic			Authentic		
	mAP↑	*AP50*↑	*AP75*↑	*mAP*↑	*AP50*↑	*AP75*↑
Ours	0.736	0.879	0.574	**0.482**	**0.502**	**0.439**
Early	0.732	0.874	0.580	0.368	0.374	0.301
Last	**0.743**	**0.904**	**0.590**	0.421	0.417	0.349

Table 8: Ablation on different layer selection strategies.

assessment. *arXiv e-prints*.

Agnolucci, L.; Galteri, L.; Bertini, M.; and Del Bimbo, A. 2024. Arniqa: Learning distortion manifold for image quality assessment. In *WACV*, 189–198.

An, X.; Xie, Y.; Yang, K.; Zhang, W.; Zhao, X.; Cheng, Z.; Wang, Y.; Xu, S.; Chen, C.; Zhu, D.; et al. 2025. Llava-onevision-1.5: Fully open framework for democratized multimodal training. *arXiv preprint arXiv:2509.23661*.

Bai, S.; Cai, Y.; Chen, R.; Chen, K.; Chen, X.; Cheng, Z.; Deng, L.; Ding, W.; Gao, C.; et al. 2025. Qwen3-VL Technical Report. *arXiv preprint arXiv:2511.21631*.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *ECCV*, 213–229.

Chen, C.; Mo, J.; Hou, J.; Wu, H.; Liao, L.; Sun, W.; Yan, Q.; and Lin, W. 2024a. Topiq: A top-down approach from semantics to distortions for image quality assessment. *IEEE Transactions on Image Processing*, 33: 2404–2418.

Chen, C.; Yang, S.; Wu, H.; Liao, L.; Zhang, Z.; Wang, A.; Sun, W.; Yan, Q.; and Lin, W. 2024b. Q-ground: Image quality grounding with large multi-modality models. In *ACM MM*, 486–495.

Chen, Z.; Zhang, X.; Li, W.; Pei, R.; Song, F.; Min, X.; Liu, X.; Yuan, X.; Guo, Y.; and Zhang, Y. 2024c. Grounding-iqa: Multimodal language grounding model for image quality assessment. *arXiv preprint arXiv:2411.17237*.

Girshick, R. 2015. Fast r-cnn. In *ICCV*, 1440–1448.

Golestaneh, S. A.; Dadsetan, S.; and Kitani, K. M. 2022. No-reference image quality assessment via transformers, relative ranking, and self-consistency. In *WACV*, 1220–1230.

Google DeepMind. 2026. Gemini 3 Pro Model Card. Google DeepMind. Model released in November 2025; last updated in May 2026.

Hosu, V.; Lin, H.; Sziranyi, T.; and Saupe, D. 2020. KonIQ-10k: An ecologically valid database for deep learning of blind image quality assessment. *IEEE TIP*, 29: 4041–4056.

Jia, Z.; Qian, J.; Zhang, Z.; Chen, Z.; and Min, X. 2026a. Refine-IQA: Multi-Stage Reinforcement Finetuning for Perceptual Image Quality Assessment. In *AAAI*.

Jia, Z.; Zhang, Z.; Zhu, X.; Li, C.; Han, J.; Liu, X.; Zhai, G.; and Min, X. 2026b. Scaling-up Perceptual Video Quality Assessment. In *AAAI*, volume 40, 22292–22300.

Kuhn, H. W. 1955. The Hungarian method for the assignment problem. *NRLQ*, 2(1-2): 83–97.

Liao, W.; Yuan, J.; Xu, Y.; Guo, C.; Zhang, Z.; Li, J.; Fu, J.; Fan, H.; Li, T.; Cui, J.; et al. 2025. ViDA-UGC: Detailed Image Quality Analysis via Visual Distortion Assessment for UGC Images. *arXiv preprint arXiv:2508.12605*.

Lin, H.; Hosu, V.; and Saupe, D. 2019. KADID-10k: A large-scale artificially distorted IQA database. In *QoMEX*, 1–3.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *ECCV*, 740–755.

Liu, S.; Zeng, Z.; Ren, T.; Li, F.; Zhang, H.; Yang, J.; Jiang, Q.; Li, C.; Yang, J.; Su, H.; et al. 2024. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *ECCV*, 38–55. Springer.

OpenAI. 2024. GPT-4o System Card. OpenAI.

OpenAI. 2025. GPT-5 System Card. OpenAI.

Qwen Team. 2024. Qwen2.5 Technical Report. *arXiv preprint arXiv:2412.15115*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*, 8748–8763.

Redmon, J.; Divvala, S.; Girshick, R.; and Farhadi, A. 2016. You only look once: Unified, real-time object detection. In *CVPR*, 779–788.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2016. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE TPAMI*, 39(6): 1137–1149.

Wang, W.; Gao, Z.; Gu, L.; Pu, H.; Cui, L.; Wei, X.; Liu, Z.; Jing, L.; Ye, S.; Shao, J.; et al. 2025. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*.

Wu, H.; Zhang, Z.; Zhang, E.; Chen, C.; Liao, L.; Wang, A.; Xu, K.; Li, C.; Hou, J.; Zhai, G.; et al. 2024a. Q-instruct: Improving low-level visual abilities for multi-modality foundation models. In *CVPR*, 25490–25500.

Wu, H.; Zhang, Z.; Zhang, W.; Chen, C.; Liao, L.; Li, C.; Gao, Y.; Wang, A.; Zhang, E.; Sun, W.; et al. 2024b. Q-ALIGN: teaching LMMs for visual scoring via discrete text-defined levels. In *ICML*, 54015–54029.

Zhai, G.; and Min, X. 2020. Perceptual image quality assessment: a survey. *SCIS*, 63(11): 211301.

Zhang, H.; Li, F.; Liu, S.; Zhang, L.; Su, H.; Zhu, J.; Ni, L.; and Shum, H.-Y. 2023a. DINO: DETR with Improved DeNoising Anchor Boxes for End-to-End Object Detection. In *The Eleventh International Conference on Learning Representations*.

Zhang, W.; Zhai, G.; Wei, Y.; Yang, X.; and Ma, K. 2023b. Blind image quality assessment via vision-language correspondence: A multitask learning perspective. In *CVPR*, 14071–14081.

Zhang, Z.; Wang, J.; Guo, Y.; Wen, F.; Chen, Z.; Wang, H.; Li, W.; Sun, L.; Zhou, Y.; Zhang, J.; Yan, B.; Jia, Z.; Xiao, J.; Tian, Y.; Zhu, X.; Zhang, K.; Li, C.; Liu, X.; Min, X.; Jia, Q.; and Zhai, G. 2025a. AIBench: Towards trustworthy evaluation under the 45° law. *Displays*, 103255.

Zhang, Z.; Wang, J.; Wen, F.; Guo, Y.; Zhao, X.; Fang, X.; Ding, S.; Jia, Z.; Xiao, J.; Shen, Y.; Zheng, Y.; Zhu, X.; Wu, Y.; Jiao, Z.; Sun, W.; Chen, Z.; Zhang, K.; Fu, K.; Cao, Y.; Hu, M.; Zhou, Y.; Zhou, X.; Cao, J.; Zhou, W.; Cao, J.; Li, R.; Zhou, D.; Tian, Y.; Zhu, X.; Li, C.; Wu, H.; Liu, X.; He, J.; Zhou, Y.; Liu, H.; Zhang, L.; Wang, Z.; Duan, H.; Zhou, Y.; Min, X.; Jia, Q.; Zhou, D.; Zhang, W.; Cao, J.; Yang, X.; Yu, J.; Zhang, S.; Duan, H.; and Zhai, G. 2025b. Large Multimodal Models Evaluation: A Survey. *SCIS*.

Zhu, J.; Wang, W.; Chen, Z.; Liu, Z.; Ye, S.; Gu, L.; Tian, H.; Duan, Y.; Su, W.; Shao, J.; et al. 2025. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*.

Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; and Dai, J. 2021. Deformable DETR: Deformable Transformers for End-to-End Object Detection. In *ICLR*.

Metrics Explanation

We provide the definitions and formulas for commonly used metrics in distorted area detection: mean Average Precision (mAP), Average Precision at specific Intersection-over-Union (IoU) thresholds ($AP50$, $AP75$), and Generalized Intersection over Union ($GIoU$).

Average Precision at IoU Threshold 50 (AP50)

AP50 is the Average Precision calculated at an IoU threshold of 50%. It measures the accuracy of predicted bounding boxes at this fixed threshold.

$$\text{AP50} = \frac{\sum_t \text{Precision}(t) \cdot \text{Recall}(t)}{\text{Total number of samples at IoU} \geq 0.5}$$

where t represents the threshold at which the predictions are considered correct.

Average Precision at IoU Threshold 75 (AP75)

AP75 is similar to AP50, but with a stricter IoU threshold of 75%. It focuses on evaluating the precision of predicted boxes that closely match the ground truth.

$$\text{AP75} = \frac{\sum_t \text{Precision}(t) \cdot \text{Recall}(t)}{\text{Total number of samples at IoU} \geq 0.75}$$

Mean Average Precision (mAP)

The mean Average Precision (mAP) is the average of the Average Precision (AP) across all object classes (with thresholds: $[0.05, 0.95]$ with the interval 0.05). It provides a general performance evaluation of how well an object detection model predicts the class and location of objects in images.

$$\text{mAP} = \frac{1}{C} \sum_{c=1}^{C} \text{AP}_c$$

where C is the number of classes, and AP_c is the average precision for class c.

Generalized Intersection over Union (GIoU)

The Generalized Intersection over Union (GIoU) extends IoU by accounting for cases where the predicted bounding box is far from the ground truth box. It is calculated as:

$$\text{GIoU} = \text{IoU} - \frac{|C - (A \cup B)|}{|C|}$$

where A and B are the predicted and ground truth boxes, and C is the smallest enclosing box containing both A and B.

Region Quality Filtering Metric: Brightness

The brightness is the average value of the V component in the HSV color space:

$$B = \frac{1}{N} \sum_{i=1}^{N} V_i$$

where B is the brightness, V_i is the brightness value (V component) of each pixel, and N is the total number of pixels in the image.

Region Quality Filtering Metric: Colorfulness

The colorfulness is calculated using the method by Hasler & Süsstrunk. The formula is:

$$C_{\text{std}} = \sqrt{\text{std}(R-G)^2 + \text{std}(0.5(R+G)-B)^2}$$

$$C_{\text{mean}} = \sqrt{\text{mean}(R-G)^2 + \text{mean}(0.5(R+G)-B)^2}$$

$$C = \frac{C_{\text{std}} + 0.3 \times C_{\text{mean}}}{C_{\text{MAX}}}$$

where C is the colorfulness, R, G, and B are the red, green, and blue components of the image, std and mean are the standard deviation and mean respectively, and $C_{\text{MAX}} = 1.3 \times \sqrt{2}$ is the maximum colorfulness value used for normalization.

Region Quality Filtering Metric: Sharpness

The sharpness is computed using the Laplacian variance of the image:

$$s_{\text{raw}} = \text{Var}\left(\nabla^2 I\right)$$

where s_{raw} is the Laplacian variance and $\nabla^2 I$ is the Laplacian operator applied to the image I. The sharpness is then mapped to the range [0, 1] using a logarithmic function:

$$s_{01} = \frac{\log(1 + s_{\text{raw}})}{\log(1 + s_{\max})}$$

$$s_{\max} = \left\lfloor \frac{12000}{\left(\frac{W}{640}\right) \times \left(\frac{H}{640}\right)} \right\rfloor$$

where H and W are the *height* and *width* of **the original image where the tested rectangle region comes from**, s_{01} is the log-compressed sharpness value, s_{raw} is the Laplacian variance, and $s_{\max}$ is a constant representing the maximum sharpness value.

Region Filtering Thresholds

To ensure all distorted regions are perceptually meaningful and avoid generating invalid samples, we apply quality control metrics to filter candidate regions before distortion application. Each region must satisfy the following criteria:

- **Brightness constraint:** $0.25 \leq B \leq 0.75$, where B is the mean value of the HSV V-channel, preventing over-dark or over-bright regions.
- **Sharpness constraint:** $S_{\text{norm}} \geq 0.9$, where $S_{\text{norm}} = \text{clip}(s_{01}, 0, 1)$ measures image sharpness via Laplacian variance, filtering out already blurry or out-of-focus regions.

Table 9: Details of the model structure and hyperparameters for the *VIGIL* model.

Model Structure/Training Hyper-Parameters	Name/Value	More Information
Vision encoder *init.*	*InternViT-300M-448px*	*Parameter size*=304.01M
Vision projector *init.*	*2-layers MLP+GeLU*	*Parameter size*=27.54M *(Layernorm+Linear(1024,3584)+GELU+Linear(3584,3584))*
LLM *init.*	*Qwen-2.5-7B*	*parameter size*=7612.82M,Decoder-only model
Classifier (for each detector)	*Linear* $(3584, 9)$	*parameter size*=0.03M
Regressor (for each detector)	*Linear* $(3584, 4)$	*parameter size*=0.01M
Image Token Feature Dimension (hidden size)	3584	/
Batch Size	16	*Per device train batch size*=2 *(for pair-wise training, this is set to* 1*)*
LR Max	2e-5	/.
Gradient Accumulation Steps	2	/
Numerical Precision	`bfloat16`	/
Epoch	1	/
Validation Steps	500	/
Optimizer	AdamW	/
Activation Checkpointing	✓	/
Deepspeed Stage	2	/

Table 10: Area number generalization test. *Avg Preds* denotes the average qualified predictions on all the images.[Per column: highest in **bold**.]

Version	*Authentic-Supp*		
	AP50↑	*AP75*↑	*Avg Preds*↑
Ours	**0.384**	**0.211**	**4.22**
Text	0.142	0.061	1.78

Table 11: Different parameter frozen strategies ablation.

Version	Synthetic			Authentic		
	mAP↑	*AP50*↑	*AP75*↑	*mAP*↑	*AP50*↑	*AP75*↑
Ours (all trainable)	**0.736**	**0.879**	0.574	**0.482**	**0.502**	**0.439**
F-LLM	0.721	0.874	0.568	0.461	0.485	0.419
F-LMM	0.685	0.832	0.475	0.352	0.403	0.342
LoRA	0.730	0.871	**0.579**	0.457	0.483	0.415

- **Colorfullness constraint:** For all evaluated region candidates $C > 0.2$, avoiding overly monotone regions. If the randomly picked distortion type of the region is related to "saturation", we additionally require $C < 0.8$,

Model Structure Supplementary Information

The model structure and key hyperparameter settings of *VIGIL-8B* are depicted in Tab. 9.

Experiments Supplementary Information

We also conduct several supplementary experiments on specific points of concern.

Justification on Some Key Issues

Under the loss design of Eq. 4 in the main paper, expanding the number of area query tokens to 15 should result in a lower loss-lower-bound compared to our multi-detector setup (because the regression loss and the classification loss for positive samples with higher assigned weights are matched multiple times in our setup). Therefore, it is valid that the training loss for the multi-detector setup decreases more rapidly in Fig. 4 in the main paper.

Implementation Details of the Object Detection Baselines

For all object detection models, during training on the *VIGIL-140K*, we set the training epochs for all models to 100. We use the same validation set and the early-stop strategy. The training hyperparameters follow the settings specified in the respective original model configuration files.

Justification on the Use of In-domain Baselines

Because most of the in-domain related works mentioned in *Sec.* are either not open-source or not directly applicable to our primary tasks, the number of available comparison models is limited. Specifically, the situation is as follows:

- *Q-Ground* is a segmentation-based model, so our data is not suitable for training/evaluation on it.

- *Grounding-IQA* only has the paper available for reference, as the model weights and training/testing code are not open-source.

- *ViDA-UGC*'s main contribution lies in its training data and paradigm. However, its training and testing methods still rely on text-based outputs (which we have already compared in our ablation studies), and the tasks corresponding to this model are substantially different from ours, making it unsuitable for direct comparison.

Predicted Area Number Generalization

To explore whether the number of distortion regions in a single training image sample affects the model's ability to predict the number of regions during testing, we annotated an additional 100 test images, each containing $4-5$ distortion regions (exceeding the maximum number of labeled regions in the training set). Consistent with the ablation study in *Sec.* , we compare the performance of the *VIGIL-8B* and the text-based SFT variant on this extended test set. We also recorded the average number of valid predicted regions. Results are shown in Tab. 10. The results show that due to the **overfitting** influence of SFT on output contexts, the number of the model's predicted valid regions is far less than our setting, thus hindering the area number generalization. Our model effectively alleviates this issue, leading to a significant improvement in performance.

Ablation Study on Model Parameter Freezing Strategies

We explore the model performance under different parameter freezing settings. Keeping all other settings the same, we compare three variants of the model:

- Freezing only the LLM part (denoted as *F-LLM*),
- Freezing the entire LMM part, retaining only the learnable area query tokens and linear classifiers and regressors (denoted as *F-LMM*),
- Performing LoRA fine-tuning on the LLM and linear classifiers and regressors (denoted as *LoRA*).

Results are shown in Tab. 11. The results show that even when all LMM parameters are frozen, the model's performance on in-domain tasks does not significantly decrease, and its performance on the OOD *S2A* task remains at an acceptable level. The performance drop for other model versions was even less noticeable. This demonstrates that our model has a feasible application scenario: it can be trained combined with the proprietary visual quality assessment LMMs to add distortion detection capabilities without significantly affecting its original performance on other tasks. For example, the base model can still use text-generation capabilities for image quality scoring and textual description tasks, and the training methods described above allow the model's functional scope to be expanded without impairing its original capabilities. This enables the development of a more versatile visual quality evaluation model.

Additional Justification of Key Points

S2A Test Set. The *COCO* dataset was selected because it was originally designed for object detection and covers a wide variety of real-world scenes. These scenes also contain diverse and naturally occurring visual degradations, making them suitable for evaluating distortion detection under challenging conditions. Owing to the substantial cost of manual annotation, together with the strict annotation verification and filtering protocol adopted in this work, the resulting test set contains fewer than 1K samples.

Evaluation Metrics. The primary evaluation metrics are standard average-precision metrics, including *mAP*, *AP50*, and *AP75*, as detailed in **Supp. Sec. A**. These metrics jointly characterize precision and recall across different intersection-over-union thresholds and therefore account for both missed detections and false-positive predictions.

Hyperparameter Selection. The post-processing thresholds (T_{nms}) and (T_{prob}) have limited influence in synthetic scenarios, where the confidence scores of predicted boxes are typically high and often exceed 0.95. In real-world scenarios, performance is more sensitive to these thresholds. To determine them without using the test set, a combinatorial search was conducted on a validation set of 100 images. Both thresholds were varied from 1 to 0 with a step size of 0.05, and all possible combinations were evaluated. The combination achieving the highest **mAP** on the validation set was then used for evaluation. This procedure provides a systematic threshold-selection strategy that can be readily applied to common real-world scenarios.

Evaluation of LMM Baselines. For general LMMs, relaxing greedy decoding produces only limited changes in detection performance. Object detection requires identifying the region with the highest prediction confidence, which is conceptually consistent with greedy decoding in LMMs, where the highest-probability token is selected for each coordinate prediction. Accordingly, non-greedy decoding is not necessarily better aligned with the conventional detection formulation.

To assess the influence of prompt wording, five semantically equivalent prompts with different phrasings were evaluated. For each image, the prediction with the highest *mIoU* among the five outputs was retained. The resulting performance was close to that obtained with the default prompt, indicating that the evaluation results are not strongly dependent on a particular prompt formulation.

Text-Generation Ability. The proposed training scheme remains compatible with conventional text-generation-based LMM supervised fine-tuning. During inference, the distortion-detection module uses only the **prefill** stage and does not modify the **decode** stage used for text generation. To examine this compatibility, the model was jointly trained with the quality-assessment instruction-tuning dataset *Q-Instruct-200K*. The jointly trained model retained both distortion-detection capability and text-generation ability, achieving an accuracy of 64.5% on *Q-Bench-Test*, which is comparable to mainstream IQA-LMMs.

As the present work focuses on distortion detection, a comprehensive evaluation of text-generation tasks is outside its scope. Nevertheless, these results indicate that, with an appropriate mixed-training strategy, the proposed detection capability can serve as a complementary enhancement to IQA-LMMs without substantially limiting their general-purpose functionality.

Rationale for the S2A Strategy and Post-Processing. The S2A strategy, which uses the second-highest classification probability, and the post-processing thresholds (T_{nms}) and (T_{prob}) are designed to exploit latent information in synthetic data without requiring human annotations. Together, they balance cross-domain information extraction and pseudo-label reliability under a resource-efficient setting.

When training relies exclusively on synthetic distortions, domain generalization becomes a central challenge. The S2A strategy adopts a relatively aggressive information-mining mechanism to extract useful signals from cross-domain data. Such a strategy may also introduce additional noise and false-positive predictions. The subsequent post-processing stage mitigates these effects by filtering predictions according to (T_{nms}) and (T_{prob}). This combination supports domain generalization while

avoiding the cost of large-scale manual annotation.

Base Model Selection. *InternVL-3-8B* was selected primarily because its fine-tuning pipeline is convenient and can be readily transferred within the InternVL model family. Preliminary experiments compared 8B-scale models from the InternVL-2, InternVL-2.5, InternVL-3, and InternVL-3.5 series and showed comparable performance across these variants. These observations suggest that base-model selection is largely dependent on empirical performance under the specific task and training configuration, rather than solely on model release recency.

Distortion Combination Details

This section provides detailed mathematical formulations for all 31 distortion sub-types across 8 major categories used in our training and in-domain test datasets. As described in the main text, following **KADIS-700K**, we consider 10 common distortion types which are merged into 8 major categories based on perceptual similarity: overexposure is merged with contrast strengthen, and underexposure is merged with contrast weaken. Each distortion is applied at two severity levels: NOTICEABLE" and SEVERE".

To simulate the irregular nature of real-world distortions as closely as possible, we randomly select an irregular region within 1/2 to 1 of the ratio of each qualified rectangular distortion area. We ensure that the four-connected region of the distortion overlaps with the four edges of the rectangle. Finally, we apply a closing morphological operation at the edges of the selected distortion area to ensure the integrity of the region.

In the formulations below, I represents the input image and I_{LQ} represents the distorted output. Unless otherwise specified, images are represented as floating-point arrays with pixel values normalized to the range $[0, 1]$ (i.e., $I \in [0, 1]^{H\times W\times C}$). The oversharpen distortion is an exception that operates directly on uint8 images in the range $[0, 255]$.

Blur (6 sub-types)

Motion Blur:

$$I_{LQ} = I * K_{\text{motion}}(r, \sigma, \theta)$$

where $(r, \sigma) \in \{(15, 7), (20, 12)\}$, $\theta \sim \mathcal{U}(-90, 90)$, and $*$ denotes convolution.

Gaussian Blur:

$$I_{LQ} = \mathcal{G}_\sigma(I), \quad \sigma \in \{3.6, 6.0\}$$

where $\mathcal{G}_\sigma$ denotes Gaussian filtering with standard deviation σ.

Glass Blur:

$$I_{LQ} = \mathcal{G}_\sigma(\text{ShufflePixels}(\mathcal{G}_\sigma(I), s, n))$$

where $(\sigma, s, n) \in \{(1.2, 2, 2), (1.6, 4, 2)\}$, s is shift range, and n is iteration count.

Lens Blur:

$$I_{LQ}^{(c)} = I^{(c)} * K_{\text{disk}}(r), \quad c \in \{R, G, B\}$$

where $r \in \{4, 8\}$ is the disk kernel radius.

Zoom Blur:

$$I_{LQ} = \frac{1}{|\mathcal{Z}| + 1}\left(I + \sum_{z\in\mathcal{Z}} \text{Zoom}(I, z)\right)$$

where $\mathcal{Z} \in \{\text{arange}(1, 1.10, 0.02), \text{arange}(1, 1.21, 0.02)\}$.

Jitter Blur:

$$I_{LQ} = \text{ShufflePixels}(I, s, 1), \quad s \in \{3, 5\}$$

Noise (6 sub-types)

Gaussian Noise (RGB):

$$I_{LQ} = \text{clip}(I + \mathcal{N}(0, \sigma^2), 0, 1), \quad \sigma \in \{0.15, 0.25\}$$

Gaussian Noise (YCrCb):

$$I_{LQ} = \text{YCrCb2RGB}(Y + \mathcal{N}_Y, Cr + \mathcal{N}_{Cr}, Cb + \mathcal{N}_{Cb})$$

where $(Y, Cr, Cb) = \text{RGB2YCrCb}(I)$, with $(\sigma_Y, \sigma_{Cr}, \sigma_{Cb}) \in \{(0.07, 0.133, 0.133), (0.09, 0.252, 0.252)\}$.

Speckle Noise:

$$I_{LQ} = \text{clip}(I + I \odot \mathcal{N}(0, s^2), 0, 1), \quad s \in \{0.28, 0.42\}$$

Spatially Correlated Noise:

$$I_{LQ} = \text{Blur}_{3\times3}(I + \mathcal{N}(0, \sigma^2)), \quad \sigma \in \{0.14, 0.22\}$$

Poisson Noise:

$$I_{LQ} = \frac{\text{Poisson}(\lambda I)}{\lambda}, \quad \lambda \in \{40, 15\}$$

Impulse Noise (Salt & Pepper):

$$I_{LQ} = \text{SaltPepper}(I, p), \quad p \in \{0.05, 0.10\}$$

where p is the fraction of affected pixels.

Compression (2 sub-types)

JPEG Compression:

$$I_{LQ} = \text{JPEG}^{-1}(\text{JPEG}(I, q)), \quad q \in \{10, 3\}$$

where q is the quality parameter.

JPEG2000 Compression:

$$I_{LQ} = \text{JP2K}^{-1}(\text{JP2K}(I, q_{\text{dB}})), \quad q_{\text{dB}} \in \{26, 23\}$$

where q_{dB} is the quality in decibels.

Overexposure & Contrast Strengthen (6 sub-types)

Brighten Shift (HSV):

$$I_{LQ} = \text{HSV2RGB}(H, S, V + \Delta V)$$

where $(H, S, V) = \text{RGB2HSV}(I)$, $\Delta V \in \{0.39, 0.65\}$.

Brighten Shift (RGB):

$$I_{LQ} = \text{clip}(I + \Delta I, 0, 1), \quad \Delta I \in \{0.325, 0.52\}$$

Brighten Gamma (HSV):

$$I_{LQ} = \text{HSV2RGB}(H, S, V^\gamma)$$

where $\gamma \in \{0.3375, 0.165\}$.

Brighten Gamma (RGB):

$$I_{LQ} = I^{\gamma}, \quad \gamma \in \{0.45, 0.30\}$$

Contrast Strengthen (Scale):

$$I_{LQ} = \text{ContrastEnhance}(I, f), \quad f \in \{5.2, 8.0\}$$

where $f > 1$ increases contrast.

Contrast Strengthen (Stretch):

$$I_{LQ} = \frac{1}{1 + \left(\frac{\bar{I}}{I+\epsilon}\right)^{f}}, \quad f \in \{12.0, 20.0\}$$

where $\bar{I} = \text{mean}(I)$ and $\epsilon = 10^{-12}$.

Underexposure & Contrast Weaken (6 sub-types)

Darken Shift (HSV):

$$I_{LQ} = \text{HSV2RGB}(H, S, V - \Delta V), \quad \Delta V \in \{0.30, 0.45\}$$

Darken Shift (RGB):

$$I_{LQ} = \text{clip}(I - \Delta I, 0, 1), \quad \Delta I \in \{0.25, 0.40\}$$

Darken Gamma (HSV):

$$I_{LQ} = \text{HSV2RGB}(H, S, V^{\gamma}), \quad \gamma \in \{3.51, 5.2\}$$

where $(H, S, V) = \text{RGB2HSV}(I)$.

Darken Gamma (RGB):

$$I_{LQ} = I^{\gamma}, \quad \gamma \in \{3.38, 4.68\}$$

Contrast Weaken (Scale):

$$I_{LQ} = \text{ContrastEnhance}(I, f), \quad f \in \{0.27, 0.09\}$$

where $f < 1$ reduces contrast.

Contrast Weaken (Stretch):

$$I_{LQ} = \frac{1}{1 + \left(\frac{\bar{I}}{I+\epsilon}\right)^{f}}, \quad f \in \{0.42, 0.30\}$$

where $\bar{I} = \text{mean}(I)$ and $\epsilon = 10^{-12}$.

Saturate Strengthen (2 sub-types)

Strengthen Saturation (HSV):

$$I_{LQ} = \text{HSV2RGB}(H, \text{clip}(f \cdot S, 0, 255), V), \quad f \in \{18.0, 96.0\}$$

where $(H, S, V) = \text{RGB2HSV}(I)$.

Strengthen Saturation (YCrCb):

$$I_{LQ} = \text{YCrCb2RGB}(Y, 128 + f(Cr - 128), 128 + f(Cb - 128))$$

where $(Y, Cr, Cb) = \text{RGB2YCrCb}(I)$, $f \in \{12.0, 24.0\}$.

Saturate Weaken (2 sub-types)

Weaken Saturation (HSV):

$$I_{LQ} = \text{HSV2RGB}(H, f \cdot S, V), \quad f \in \{0.28, 0.0\}$$

where $(H, S, V) = \text{RGB2HSV}(I)$.

Weaken Saturation (YCrCb):

$$I_{LQ} = \text{YCrCb2RGB}(Y, 128 + f(Cr - 128), 128 + f(Cb - 128))$$

where $(Y, Cr, Cb) = \text{RGB2YCrCb}(I)$, $f \in \{0.14, 0.0\}$.

Oversharpen (1 sub-type)

Oversharpen:

$$I_{LQ} = \text{clip}((1 + \alpha)I - \alpha \mathcal{G}_{\sigma}(I), 0, 255)$$

where $\alpha \in \{6.0, 12.0\}$, $\sigma = 5$, and input image is in $[0, 255]$ range.

Limitations

Due to the fact that most in-the-wild UGC image content cannot serve as an effective source for adding distortions, our dataset is still limited in scale, preventing us from verifying the data scaling law. Additionally, due to resource constraints, we have not yet included large-scale annotated real-world scene data in our training set. As a result, there is still room for improvement in the model's performance. These limitations represent key insights and directions for our future research.

Detailed Statistical Information

We present comprehensive statistical analyses of our datasets, including the training set (*VIGIL-140K*), the synthetic distortion test set, and the authentic distortion test set. We visualize five key statistical distributions for each dataset: image resolution distribution, bounding box resolution distribution, box-to-image area ratio distribution, number of boxes per image distribution, and distortion type distribution.

Training Set Statistics

The training set *VIGIL-140K* comprises over $145K$ distorted images with more than $205K$ distortion area annotations.

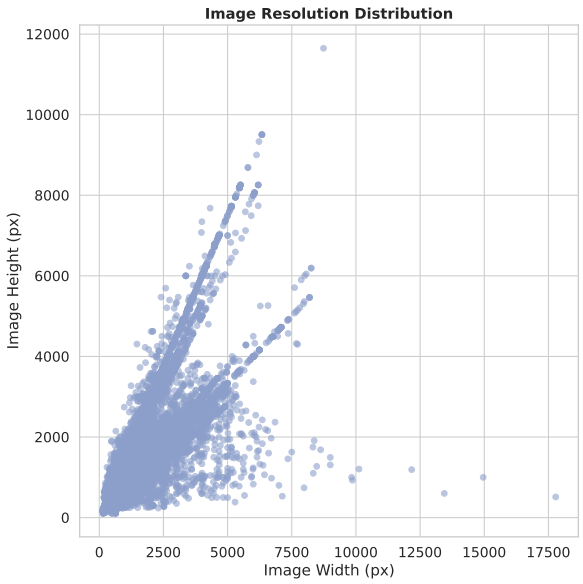

Figure 5: Image resolution distribution of the training set.

Fig. 5 shows the joint distribution of image width and height. The dataset exhibits high diversity, with individual dimensions reaching up to $17,500$ pixels in width and $12,000$ pixels in height. Most images are concentrated within the 500×500 to $5,000 \times 5,000$ pixel range. Notably, the clear linear patterns suggest that the images follow several standard aspect ratios.

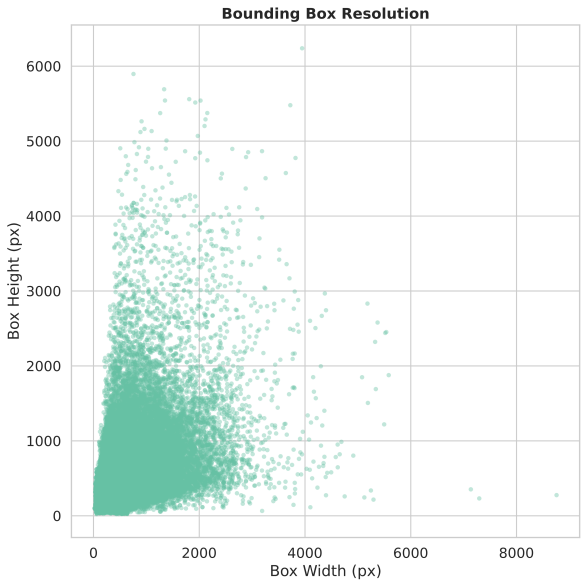

Figure 6: Bounding box resolution distribution of the training set.

Fig. 6 illustrates the distribution of bounding box resolutions. The distortion regions cover a wide range of spatial scales, contributing to robust model performance across varying region sizes. Most bounding boxes are small to medium (typically below 2,000 pixels), while a substantial number exceed $4{,}000 \times 4{,}000$ pixels.

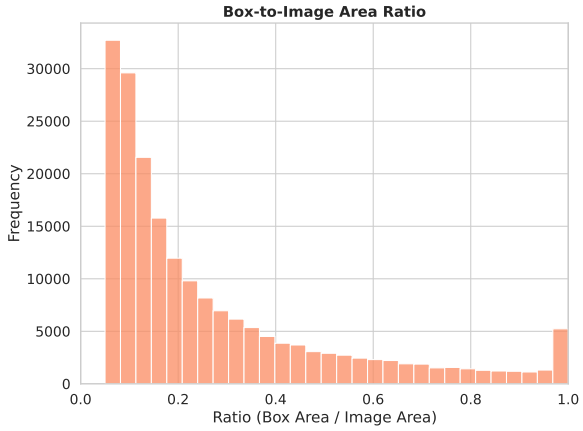

Figure 7: Box-to-image area ratio distribution of the training set.

The box-to-image area ratio distribution (Fig. 7) demonstrates that most distortion regions occupy between 5% and 30% of the total image area.

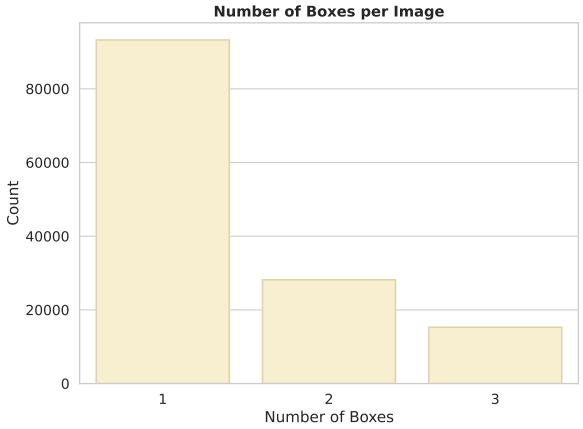

Figure 8: Number of boxes per image distribution of the training set.

Fig. 8 shows the distribution of the number of distortion boxes per image. Following our distortion combination algorithm (Algorithm 1), each image contains 1–3 distortion regions.

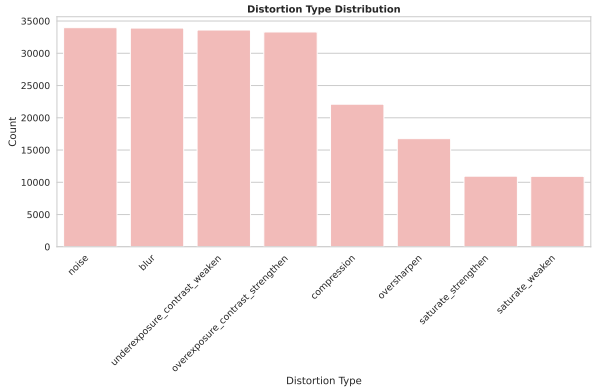

Figure 9: Distortion type distribution (number of labeled areas) of the training set.

Fig. 9 presents the distribution of distortion types in the training set, counted by the number of labeled distortion areas. Noise, blur, underexposure/contrast weaken, and overexposure/contrast strengthen are the most prevalent types, each with over 30K instances.

Synthetic Distortion Test Set Statistics

The synthetic distortion test set contains 4,600 images generated using the same distortion synthesis pipeline, applied to held-out source images. This test set is used to evaluate the model's in-domain detection performance.

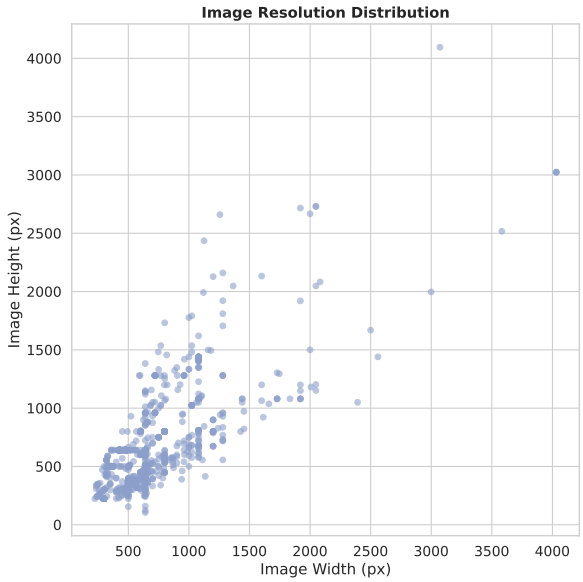

Figure 10: Image resolution distribution of the synthetic distortion test set.

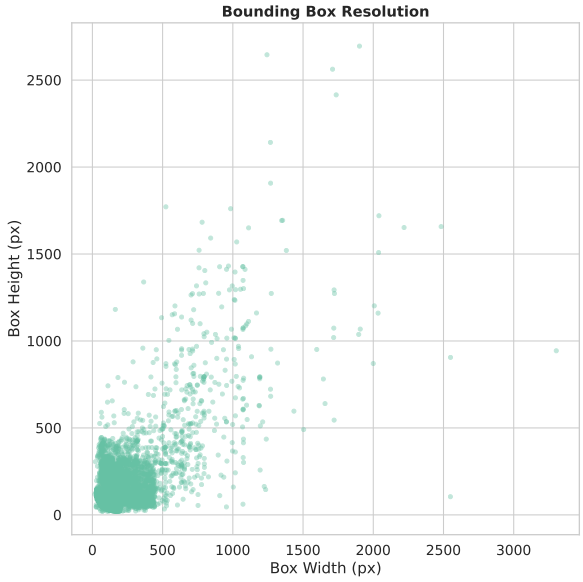

Figure 11: Bounding box resolution distribution of the synthetic distortion test set.

Figs. 10 and 11 show the image and bounding box resolution distributions, respectively. The synthetic test set exhibits similar patterns to the training set but with a slightly different distribution due to the different source image pools (*KADIS-700K*, *COCO-2017-test*, and *Unsplash*).

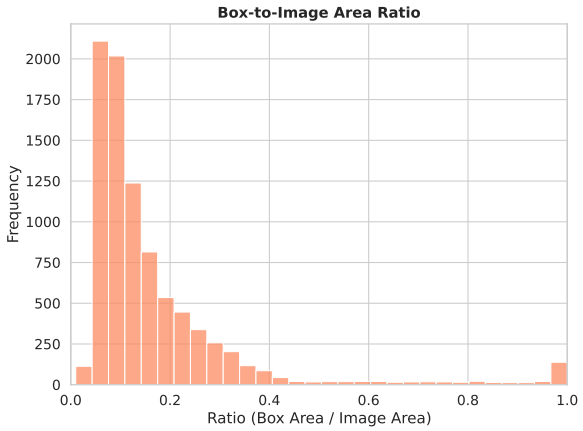

Figure 12: Box-to-image area ratio distribution of the synthetic distortion test set.

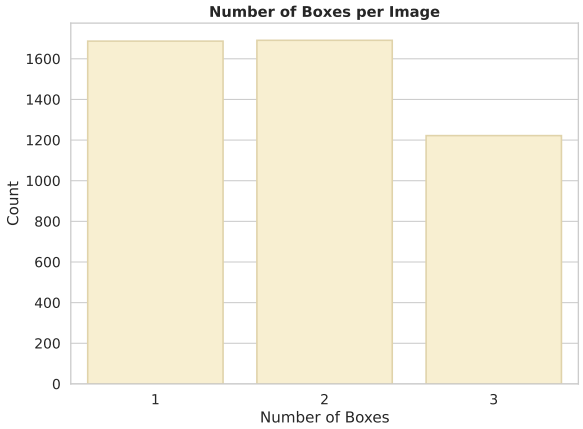

Figure 13: Number of boxes per image distribution of the synthetic distortion test set.

The box-to-image ratio (Fig. 12) and box count (Fig. 13) distributions show the size of the distortion areas relative to the image and the number of distorted regions found in each image within the synthetic test set.

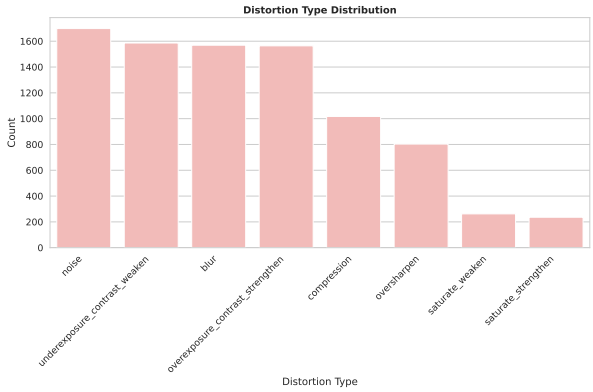

Figure 14: Distortion type distribution of the synthetic distortion test set.

The distortion type distribution (Fig. 14) shows a balanced representation across all eight major distortion categories.

Authentic Distortion Test Set Statistics

The authentic distortion test set consists of 815 human-annotated UGC images with real-world distortions and is used to evaluate out-of-domain (OOD) authentic distortion localization for *S2A* (synthetic-to-authentic) generalization.

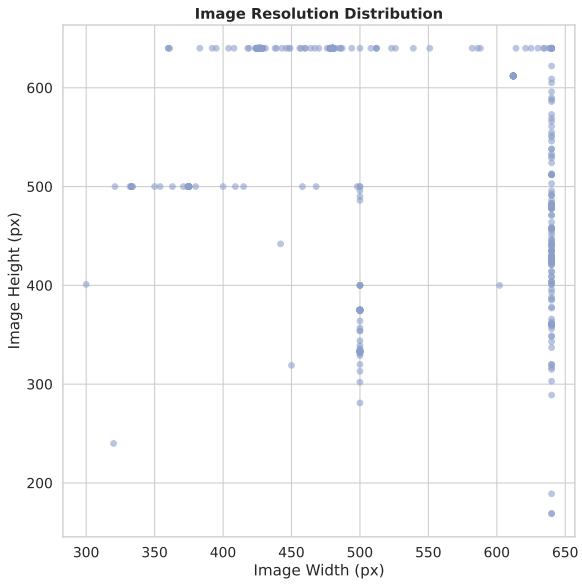

Figure 15: Image resolution distribution of the authentic distortion test set.

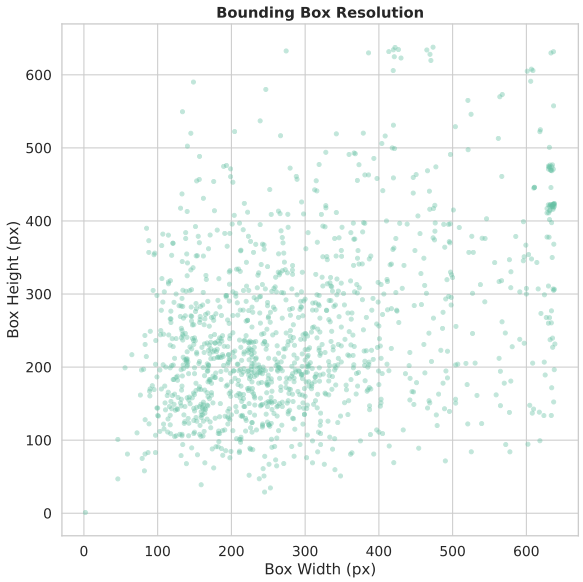

Figure 16: Bounding box resolution distribution of the authentic distortion test set.

Figs. 15 and 16 illustrate the resolution distributions for the authentic test set. The image resolutions are mainly concentrated around 500×500 and 640×640 pixels, reflecting common resolution scales of UGC images from the *COCO2017* dataset. The bounding box resolutions show a more uniform spread compared to synthetic data, reflecting the irregular and varied nature of authentic distortion regions.

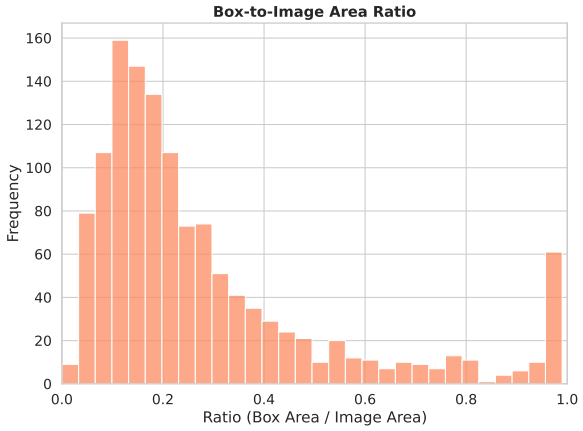

Figure 17: Box-to-image area ratio distribution of the authentic distortion test set.

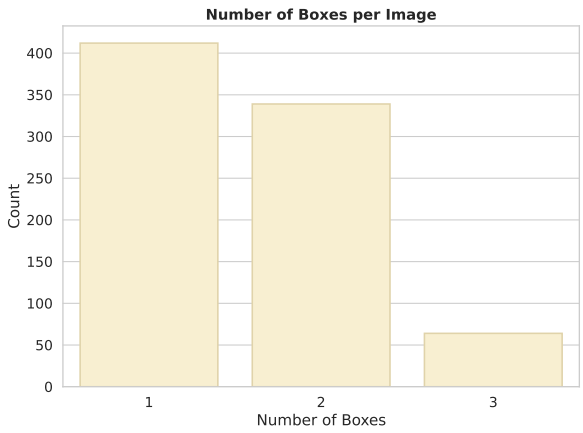

Figure 18: Number of boxes per image distribution of the authentic distortion test set.

The box-to-image area ratio distribution (Fig. 17) indicates that authentic distortions often occupy relatively large portions of the image, with a substantial number of regions exceeding 30% of the image area, while the distribution of distortion region counts per image is summarized in Fig. 18.

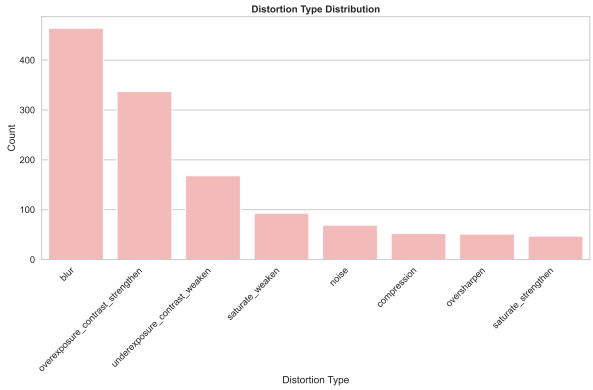

Figure 19: Distortion type distribution of the authentic distortion test set.

Fig. 19 shows the distribution of authentic distortion types.

Visualization

Training Set Samples

Fig. 20 shows 210 representative samples from our training set *VIGIL-140K*.

Authentic Distortion Test Set Samples

Figs 21 presents 210 samples from our authentic distortion test set, showcasing real-world distortions encountered in user-generated content for evaluating OOD generalization.

Model Predictions

In Fig. 22, we present the detection results for each detector of **VIGIL-8B** under the $M = N = 5$ setting (the choice of $N = 5$ instead of the main setting $N = 3$ from the main paper is made to more clearly show the detection differences among the detectors). In Fig. 23, we present the final output results of the model under the main setting ($M = 5, N = 3$) in the main paper.

Figure 20: 210 representative training samples from *VIGIL-140K*.

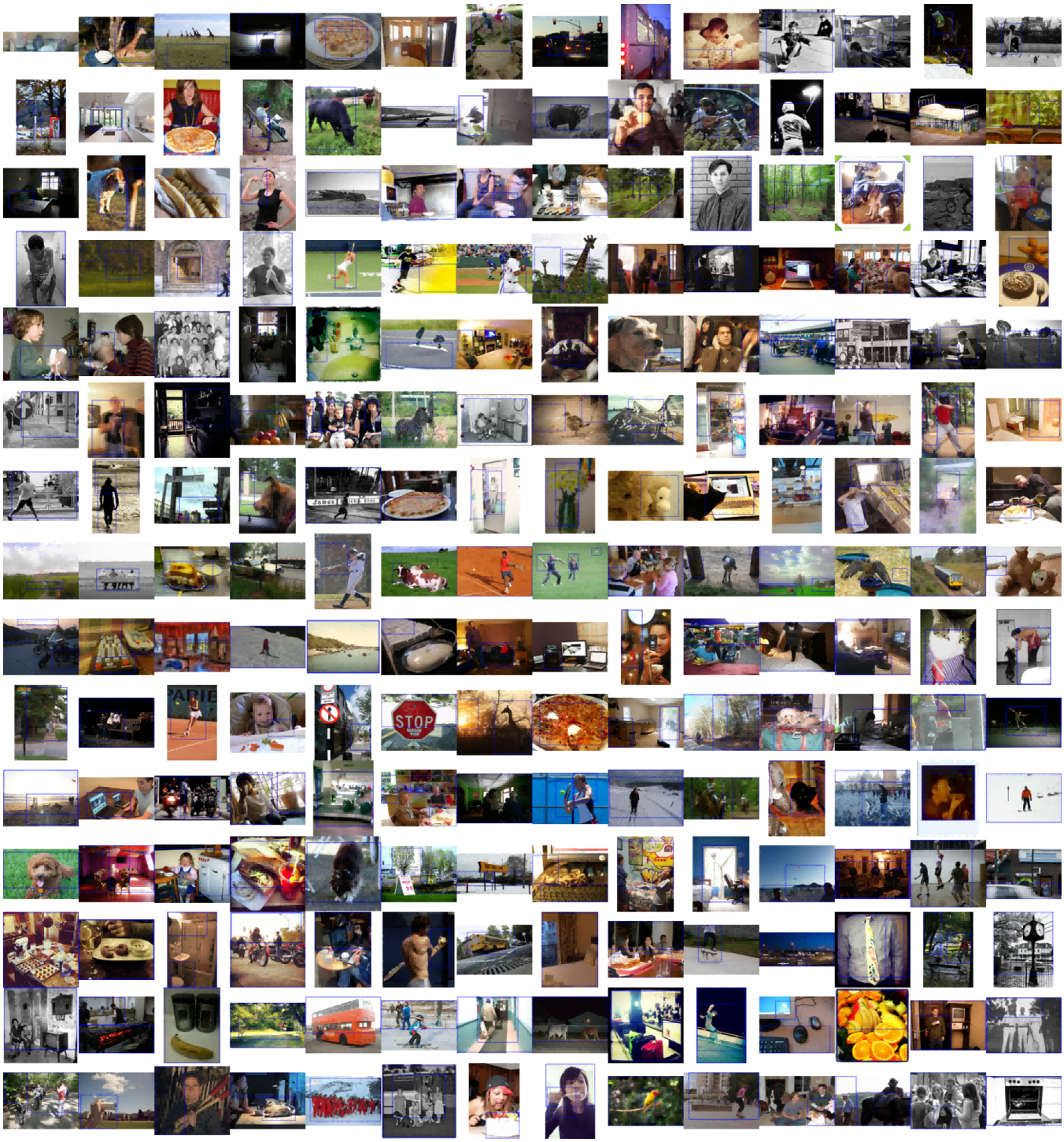

Figure 21: 210 samples from the authentic distortion test set.

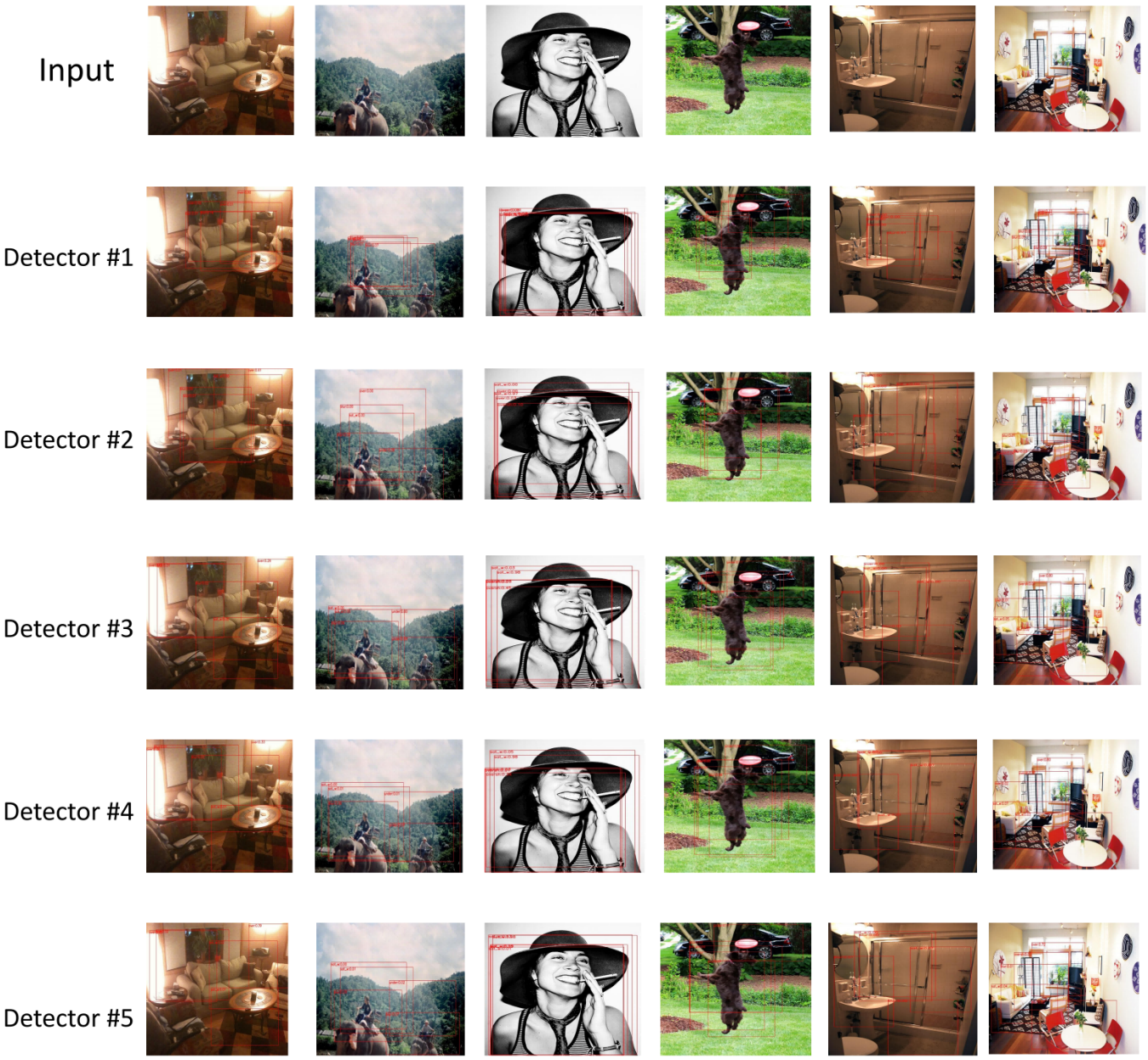

Figure 22: Detailed case study examples of the predicted results of each detector of *VIGIL-8B* under the setting $M = N = 5$. All images are resized to the same scale for better arrangement and visualization. The annotations in the images "comp", "over", "under", "sat-s", 'sat-w", and "over.sh" denote "compression", "overexposure-contrast-strengthen", "underexposure-contrast-weaken", "saturate-strength", "saturate-weaken", and "oversharpen", respectively.

Figure 23: Examples of the final predicted results after post-processing of *VIGIL-8B* under the setting $M = 5, N = 3$ (our setting in the main paper). All images are resized to the same scale for better arrangement and visualization.