WHAT SHOULD WE GENERATE FOR UAV DETECTION?

Anonymous authors
Paper under double-blind review

ABSTRACT

Generative augmentation expands the environmental coverage for small unmanned aerial vehicle (UAV) detection, but semantically distinct conditions can lead to similar changes in the detector's features. We combine a two-stage decoupled generation method with detector-guided condition selection. To provide a shared target reference for comparing conditions, we first edit the background and then blend the source UAV back into the image using a soft mask. A frozen detector, trained on clean images, measures paired context responses to estimate condition similarities, and a backward greedy algorithm balances response coverage against pairwise overlap, selecting conditions for a user-specified weight without prescribing the subset size. The retained generated images are combined with clean images for detector training. With five candidate conditions, all selected two- to four-condition configurations achieve higher external mean AP than Clean-only and Full generation with both RT-DETRv2-S and YOLO11m. For RT-DETRv2-S, a selected three-condition subset improves external mean AP by 2.95 points over Clean-only, 0.80 over Full generation, and 2.11 over Full generation with a matched clean/generated sampling ratio. For YOLO11m, the selected subsets improve external mean AP over Full generation by 1.05-1.14 points.

1 INTRODUCTION

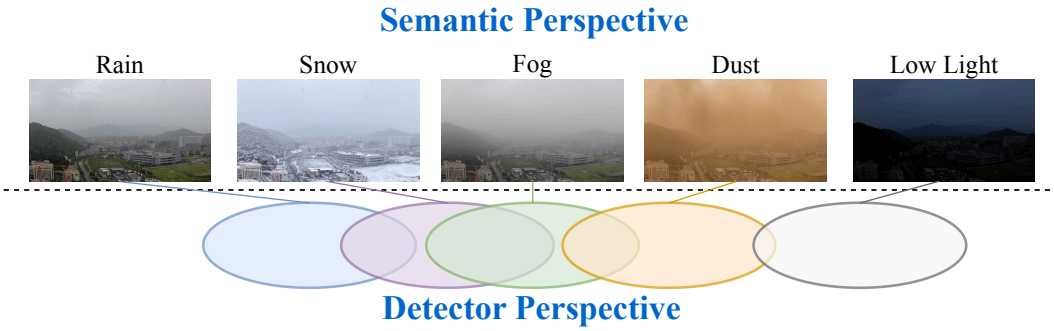


Figure 1: **Semantic perspective versus detector perspective.** The upper panel illustrates semantically distinct adverse conditions. The lower panel depicts selected response relationships: some conditions can cause similar changes in detector features. Greater overlap between ellipses indicates more similar detector responses (ellipse overlap is conceptual and is not fitted to measured similarities).

Generative augmentation can extend detector training beyond the conditions represented in annotated data. Layout-conditioned diffusion models can generate synthetic images for object detection (Chen et al., 2024), while adverse-condition synthesis expands environmental coverage for visual perception (Tremblay et al., 2021; Qian et al., 2025). This is particularly valuable for detecting unmanned aerial vehicles (UAVs), as they often occupy only a small portion of the image against a changing background (Zheng et al., 2021). Adverse conditions further reduce visibility and impair

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.

detection performance (Munir et al., 2024). To improve robustness, prior work has combined original images with images synthesized under predefined adverse conditions in fixed proportions for detector training (Gurbindo et al., 2025). However, it remains unclear which conditions should be combined to improve detection performance. We study which adverse conditions should be generated and combined for UAV detector training.

Since the selected conditions are used for detector training, their differences should also be examined in relation to the detection task. Semantically distinct conditions may affect the detector in similar ways (Figure 1). This motivates comparing conditions from the detector's perspective when constructing augmentation sets.

For small UAVs, editing the environment directly may also alter or erase the target, undermining the original annotations. Therefore, generative augmentation should change the environment while maintaining a consistent target reference across conditions.

We propose a detector-guided generative augmentation framework for UAV detection that combines two-stage decoupled generation with coverage-redundancy condition selection. The generation strategy separates environmental editing from target integration to provide a shared target reference across different generative conditions. The selection strategy uses paired context responses from a frozen, clean-trained detector to determine which conditions to retain. Images generated under the selected conditions are combined with clean images for detector training.

Our contributions are threefold:

- We introduce a two-stage decoupled generation strategy that separates environmental editing from source-UAV integration, providing a shared target reference across conditions for detector training and response comparison.
- We propose a detector-guided approach to condition selection based on the feature changes caused by candidate conditions. To the best of our knowledge, this is the first study to use detector features to select adverse-condition generation tasks in UAV detection.
- We show that selecting fewer conditions can improve UAV detection compared to using all five candidate conditions. This finding is supported by the results for RT-DETRv2-S and YOLO11m.

2 Related Work

2.1 UAV Detection under Adverse Conditions

Munir et al. (2024) analyze the degradation of detection performance under adverse weather conditions and in the presence of image distortions, then evaluate data augmentation with corrupted images. Beyond Clear Skies introduces SDV-W, with matched drone scenes across weather and seasons, and studies weather diversity by replacing generic synthetic training images with weather-varied images while maintaining a fixed dataset size (Lenhard et al., 2026). Our work focuses on which candidate adverse conditions should be retained together for detector training.

2.2 Generative Augmentation for Detection

GeoDiffusion and ODGEN control the layout of objects in diffusion-generated detection data (Chen et al., 2024; Zhu et al., 2024). Aerial image synthesis also addresses object-level control and ensures annotation consistency (Li et al., 2026). Weather editing methods improve scene fidelity through control in intrinsic space and structure-aware generation (Zhu et al., 2026; Ning et al., 2026). Our two-stage decoupled generation strategy maintains a common target reference across generated variants, allowing both detector training and paired response measurement to be carried out consistently.

Mishra et al. (2022) introduce Task2Sim, which adapts simulation parameters to downstream tasks to generate synthetic pre-training data, while prior work compares detectors and synthetic training configurations under adverse conditions (Teeti et al., 2022). Detector feedback can guide both generation and sample selection. DetDiffusion conditions generation on object difficulty estimated by a pretrained detector (Wang et al., 2024). Neptune-X selects generated maritime images using attribute-level detection difficulty and sample prediction errors (Guo et al., 2025). We select a set

of adverse conditions for augmentation, with each condition contributing generated variants of the source images.

2.3 Condition Relationships for Augmentation

Condition relationships can guide augmentation design. Mintun et al. (2021) study the similarity between augmentation and corruption in classification tasks by calculating the difference between the transformed features and the original features across the entire image dataset. Hsiang & Chen (2024) select representative corruptions using cross-corruption robustness gains after training condition-specific detectors. For condition selection, we estimate condition relationships without fine-tuning for specific conditions, using a frozen detector to measure context response. We compute cosine similarities within each source image before averaging across sources, then use the resulting relationships to select conditions through a coverage-redundancy objective.

3 Problem Formulation

Data and candidate conditions. Let $\mathcal{D}_{\text{clean}} = \{(x_i, y_i)\}_{i=1}^{N}$ be a clean training set for UAV detection, where y_i denotes the UAV bounding-box annotations for image x_i in normalized image coordinates. We consider a finite set of candidate adverse conditions $\mathcal{T} = \{t_1, \ldots, t_m\}$, including rain, snow, fog, dust, and low light in our experiments.

Paired conditional generation. Let G denote the conditional editing pipeline and $\mathcal{I} \subseteq \{1, \ldots, N\}$ a shared set of source-image indices. For each $t \in \mathcal{T}$, the generated image is $\widetilde{x}_i^t = G(x_i, y_i; t)$, yielding

$$\mathcal{D}_t = \left\{(\widetilde{x}_i^t, y_i) : i \in \mathcal{I}\right\}. \tag{1}$$

The generation process reintegrates source UAV content at the annotated locations through soft-mask blending. Shared sources establish clean-generated pairs and corresponding variants across conditions.

Condition-set design. We seek a nonempty subset $\mathcal{A} \subseteq \mathcal{T}$. Clean data are always retained, giving the training pool

$$\mathcal{D}_{\text{aug}}(\mathcal{A}) = \mathcal{D}_{\text{clean}} \cup \bigcup_{t \in \mathcal{A}} \mathcal{D}_t. \tag{2}$$

A detector f_0, trained on clean images and then frozen, provides the paired response measurements used to select $\mathcal{A}$. The final detector is $f_{\mathcal{A}} = \text{Train}(\mathcal{D}_{\text{aug}}(\mathcal{A}))$, where Train denotes the prescribed detector-training procedure.

4 Method

Figure 2 summarizes the three components: two-stage generation, context response measurement with the frozen detector f_0, and backward greedy condition selection for training $f_{\mathcal{A}}$.

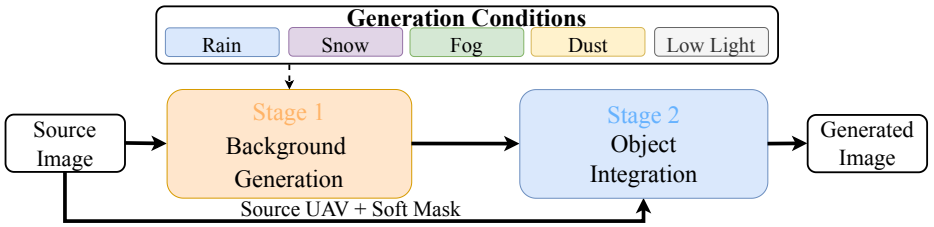
4.1 Two-Stage Decoupled Generation

We separate condition-dependent background generation from source-UAV integration (Figure 2(a)). For each source image x_i, we obtain target masks using Segment Anything (Kirillov et al., 2023), with the normalized boxes y_i converted to image coordinates as prompts. We dilate the binary source-UAV masks to form protection masks, then resize and Gaussian-smooth them to obtain $M_i \in [0, 1]^{H \times W}$ on the generation grid. The same M_i is used for input preparation and target integration across all conditions, keeping the blending settings fixed for downstream comparisons.

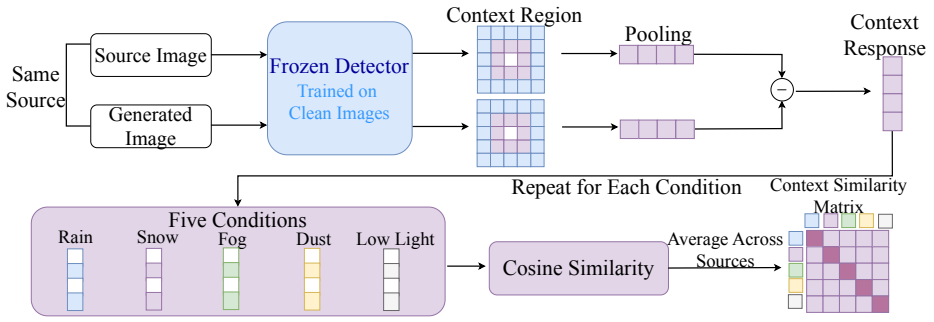
The first stage generates the background under condition t. We prepare the editing input by blending a blurred version of the source image into the target region to help it merge with the surrounding background:

$$x_i^{\text{pre}} = M_i \odot \text{Blur}(x_i) + (1 - M_i) \odot x_i, \tag{3}$$

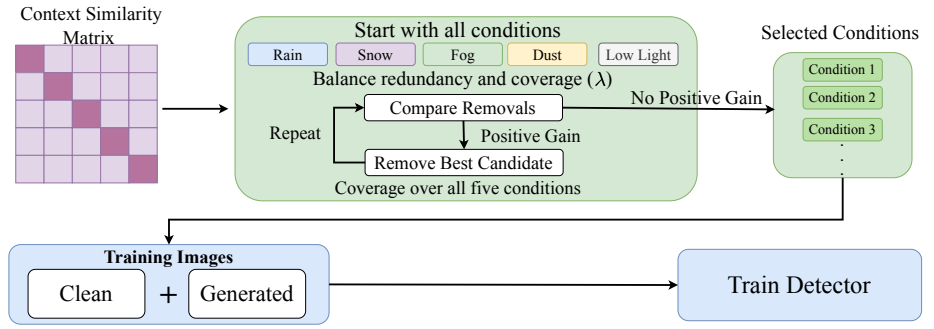
where $\odot$ denotes elementwise multiplication, with mask weights shared across color channels. The image editor G_{bg} receives the full preprocessed image and a condition-specific background-editing



(a) Two-stage decoupled generation



(b) Detector-response similarity



(c) Backward greedy condition selection

Figure 2: **Overview of the proposed framework.** (a) Condition-specific background generation followed by source-UAV integration through soft-mask blending. (b) A frozen clean-trained detector measures context responses to construct the condition similarity matrix. (c) Backward greedy selection balances coverage and redundancy. Generated images from the selected conditions are combined with clean images for detector training.

prompt, producing

$$b_i^t = G_{\text{bg}}(x_i^{\text{pre}}; t). \tag{4}$$

The prompt asks the editor to remove the UAV, introduce no additional flying objects, and apply condition t while retaining the source viewpoint and scene layout.

The second stage integrates the source UAV into b_i^t through soft-mask blending:

$$\widetilde{x}_i^t = G(x_i, y_i; t) = M_i \odot x_i + (1 - M_i) \odot b_i^t. \tag{5}$$

Source content is blended at its original coordinates without target-specific translation or rescaling. Feathered mask weights provide gradual target-background transitions. Each source-condition pair requires one image-editing call followed by deterministic compositing. The output retains annotations y_i and is paired with x_i for response measurement.

4.2 Paired Detector-Response Similarity

We characterize each condition through the changes it produces in the context features of a frozen clean-trained detector f_0 (Figure 2(b)). The same detector is used for all source images and condition

variants. Let $\phi_0(x) \in \mathbb{R}^{C\times H_f\times W_f}$ be a spatial feature map within a cross-scale fusion module of f_0, before prediction-specific heads; C and $H_f \times W_f$ denote its channel dimension and grid size. For RT-DETRv2-S, we use the highest-resolution HybridEncoder output (stride 8), which combines fine-scale features with information propagated from coarser levels (Zhao et al., 2024; Lv et al., 2024). This detection-trained representation remains on a dense grid before object-query selection, allowing source-defined regional measurements without matching predicted objects across variants. Its fine spatial sampling is suited to pooling context around small UAVs.

We pool this feature map within a square centered on the source bounding box, excluding the protection region after additional dilation on the detector-input grid. We map the remaining context region to the feature grid by area averaging, obtaining fractional weights $w_i(p) \in [0, 1]$ at locations $p \in \Omega$, where Ω is the set of feature-grid locations. The same source-derived weights are used for the clean image and all its condition variants. We pool the context features as

$$h_i(x) = \frac{\sum_{p\in\Omega} w_i(p)\,\phi_0(x)_p}{\sum_{p\in\Omega} w_i(p) + \epsilon}, \tag{6}$$

where $\epsilon = 10^{-12}$ is a numerical stabilizer.

For each clean-generated pair, we subtract the pooled clean features from the generated features and normalize the resulting difference:

$$\delta_i^t = h_i(\widetilde{x}_i^{\,t}) - h_i(x_i), \qquad \widehat{\delta}_i^{\,t} = \frac{\delta_i^t}{\|\delta_i^t\|_2 + \epsilon}. \tag{7}$$

Each response is defined relative to the same clean reference, using a shared detector and spatial support. Normalizing the difference emphasizes the direction of the context change.

Let $\mathcal{I}_{\text{meas}} \subseteq \mathcal{I}$ contain the shared source indices used for response measurement. These source images are disjoint from the clean images used to train f_0. Both sets are drawn exclusively from the internal training partition (Section 5.1). For each pair of conditions $t, u \in \mathcal{T}$, we compare their normalized changes within each source image and average across sources:

$$S_{tu} = \frac{1}{|\mathcal{I}_{\text{meas}}|} \sum_{i\in\mathcal{I}_{\text{meas}}} (\widehat{\delta}_i^{\,t})^\top \widehat{\delta}_i^{\,u}, \qquad t, u \in \mathcal{T}. \tag{8}$$

Each source contributes one stabilized cosine similarity. Larger values indicate more aligned context-response directions.

4.3 Coverage-Redundancy Condition Selection

We use the response relationships to select conditions that cover the candidate response patterns while limiting overlap within the selected set (Figure 2(c)). We retain the off-diagonal entries from Equation (8) and set $S_{tt} = 1$. For a nonempty $\mathcal{A} \subseteq \mathcal{T}$, we use facility-location coverage (Wei et al., 2015) and define pairwise response redundancy as

$$F(\mathcal{A}) = \sum_{t\in\mathcal{T}} \max_{a\in\mathcal{A}} S_{ta}, \qquad R(\mathcal{A}) = \sum_{\substack{\{a,b\}\subseteq\mathcal{A}\\ a\neq b}} S_{ab}. \tag{9}$$

Coverage represents every original candidate by its most similar retained condition, so removed conditions remain part of the coverage requirement. The pairwise term sums signed response similarities. We combine them as:

$$J_\lambda(\mathcal{A}) = F(\mathcal{A}) - \lambda R(\mathcal{A}), \qquad \lambda \geq 0. \tag{10}$$

Both terms are unnormalized sums. The user-specified weight λ controls the penalty on response overlap relative to coverage, indirectly adjusting the size and composition of the retained set.

Starting from $\mathcal{A} = \mathcal{T}$, we apply backward greedy deletion. For $|\mathcal{A}| > 1$, the selection-score gain from removing $a \in \mathcal{A}$ is

$$\begin{aligned}\Delta_\lambda(a \mid \mathcal{A}) &= J_\lambda(\mathcal{A}\setminus\{a\}) - J_\lambda(\mathcal{A}) \\ &= \lambda \sum_{b\in\mathcal{A}\setminus\{a\}} S_{ab} - \big[F(\mathcal{A}) - F(\mathcal{A}\setminus\{a\})\big].\end{aligned} \tag{11}$$

The two terms are the weighted redundancy removed and the coverage lost, respectively. Each iteration removes the condition with the largest strictly positive gain, then recomputes all gains on the updated set. Exact ties follow the fixed order rain, snow, fog, dust, and low light. Selection stops when no gain is positive or one condition remains. Algorithm 1 summarizes the procedure.

For each λ, deletion decisions use only the fixed matrix S. The stopping rule determines the subset size; all clean images and the retained generated datasets form the training pool in Equation (2). The practical utility of this response-based selection criterion is evaluated through the downstream subset comparisons in Section 5.4.

Algorithm 1 Backward greedy condition selection

Input: Candidate set $\mathcal{T}$, fixed symmetric matrix S with $S_{tt} = 1$, $\lambda \geq 0$, and a fixed tie-breaking order
Output: Nonempty retained set $\mathcal{A} \subseteq \mathcal{T}$

$\mathcal{A} \leftarrow \mathcal{T}$
while $|\mathcal{A}| > 1$ **do**
  for all $a \in \mathcal{A}$ **do**
    $g_a \leftarrow \Delta_\lambda(a \mid \mathcal{A})$ ▷ Equation (11)
  end for
  $a^\star \leftarrow \arg\max_{a \in \mathcal{A}} g_a$ ▷ Fixed-order exact ties
  if $g_{a^\star} \leq 0$ **then**
    break
  end if
  $\mathcal{A} \leftarrow \mathcal{A} \setminus \{a^\star\}$
end while
return $\mathcal{A}$

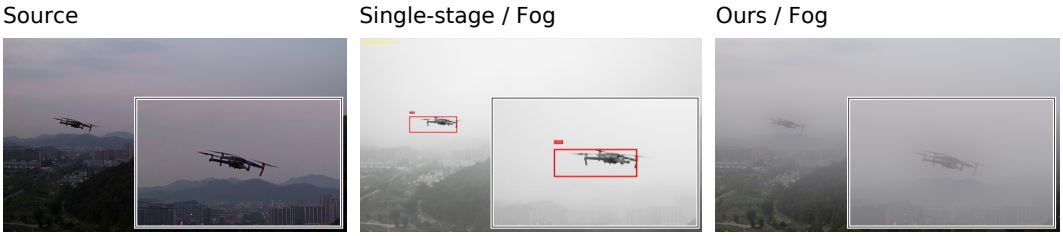
5 Experiments

5.1 Experimental Setup

Data and internal evaluation. We use 730 annotated clean images from Det-Fly (Zheng et al., 2021), including 146 source images for generation and 584 additional clean images. For the main experiments, we generate one variant per source image for each of five conditions, resulting in a total of 730 generated images. For internal evaluation, we split the original images into 511 training, 110 validation, and 109 test images. Within these partitions, the 146 sources images are allocated as 102/22/22 images, and the 584 additional clean images as 409/88/87 images, respectively. An original image and all its generated variants remain in the same partition. Training combines the 511 clean images with 102 generated images per selected condition. The fixed validation and test sets contain 220 and 219 images, respectively, and cover clean images and all five conditions, including those omitted from training. In main detection comparisons, internal validation is used only to monitor performance; all reported results use the final exponential-moving-average (EMA) checkpoint.

External evaluation. We train separate detectors using all 730 clean images and 146 generated images per retained condition. We evaluate on a real-world test set and a synthetic test set, comprising 247 images in total. The real-world set contains naturally captured images, combining images from Det-Fly with snowy images selected from DUT Anti-UAV (Zhao et al., 2022). Although some of the images also originate from Det-Fly, their scene backgrounds and camera viewpoints differ significantly from those represented in the training data. The synthetic set contains multi-UAV images with annotated instances from Simuletic (Simuletic Team, 2025).

Implementation. For the main experiments, we use FLUX.2 [dev] (Black Forest Labs, 2025) for background editing at a resolution of 1024×576, with 28 inference steps and a guidance scale of 3.0. The main experiments use RT-DETRv2-S with a ResNet-18 backbone (Lv et al., 2024) for both the frozen and downstream detectors, initialized from the same COCO-pretrained weights. The frozen detector and the main comparison models are fitted independently for 6,000 successful optimizer updates using AdamW, a batch size of 16, and a learning rate and weight decay of 10^{-4}. The learning rate is warmed up for 300 updates and reduced by a factor of 10 at update 4,500. Images



(a) Same-source fog comparison



(b) Additional two-stage outputs

Figure 3: **Target preservation during conditional generation.** (a) Source image and corresponding outputs for fog from single-stage editing and our two-stage method. (b) Two-stage outputs of snow, dust, and low-light from the same source image.

are resized to 960×960 without additional random image augmentation and are uniformly sampled from shuffled training images. We use mixed precision and evaluate the final exponential-moving-average weights.

Response measurement and comparisons. The shared RT-DETRv2-S detector is trained once on the 409 additional clean images assigned to the internal training partition and then frozen using its final EMA weights. We estimate the context-response similarity matrix using the 102 generation sources in that same training partition and their five generated variants. The frozen-detector training sources and response-measurement sources are disjoint, and neither includes internal validation or test sources. The same selected condition sets are used in both downstream evaluation protocols. Clean-only uses original images; Full generation combines clean images with generated images from all five candidate conditions. We report representative values of λ for all distinct terminal sets returned by the backward greedy algorithm as λ varies (Table 1). The full set returned at $\lambda = 0$ coincides with Full generation. For comparison, we evaluate all 30 subsets between one and four generated conditions and report their RT-DETRv2-S internal and external results in Appendix B. Within each protocol, the source images, generated images, training recipe, and update budget are shared. The retained conditions determine the size of the training pool and, under uniform image sampling, the clean/generated sampling proportions.

Evaluation metrics. We report COCO-style AP (Lin et al., 2014) averaged over IoU thresholds from 0.50 to 0.95 in steps of 0.05, on a 0-100 scale. Internal AP averages six AP scores, computed separately for clean images and each of the five generated conditions. External AP is the equal-weight mean calculated on the pooled real test set and the synthetic test set. For the main experiments, we first compute internal or external AP within each run and then average across training runs. Results and within-size external rankings appear in Appendix B; the YOLO and short-budget transfer protocols appear in Appendices A and C.

5.2 Target Preservation in Two-Stage Decoupled Generation

Figure 3(a) compares same-source fog outputs. Single-stage editing changes the UAV's tilt and body shape; additional examples show enlargement, displacement, or extra UAVs (Appendix D). These trial outputs were excluded from detector training because they will invalidate the source annotations.

The two-stage examples retain recognizable UAV outlines and positions while changing the surrounding appearance (Figure 3). Additional examples with FLUX.2-klein-base-9B illustrate the same procedure with a second editor (Appendix E). Response measurement and downstream training use the FLUX.2-dev outputs.

5.3 Detector-Response Relationships and Condition Selection

We apply the procedure in Section 4.2 to 102 training sources and their five variants (510 clean-generated pairs), using the frozen RT-DETRv2-S detector. Figure 4 shows the resulting matrix.

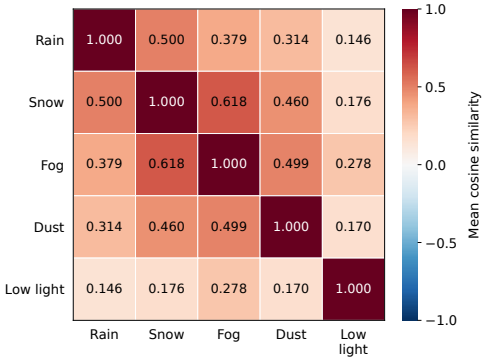


Figure 4: **RT-DETRv2-S response similarity.** Mean within-source cosine similarity of generated-minus-clean context responses over 102 training sources.

Snow and fog have the largest off-diagonal similarity (0.618), indicating the strongest average alignment of context-response directions among the candidate pairs. Low light is less aligned with rain (0.146) and dust (0.170).

Cross-condition transfer. We compare the context-response similarity matrix to single-condition transfer measurements. Each auxiliary RT-DETRv2-S model uses 102 clean images and their 102 variants within a single condition; the control uses two copies of the same clean images. The transfer gain corresponds to the AP difference from this control when evaluating under another condition. Across the ten unordered pairs of conditions, response similarity is positively associated with the mean of the two directional gains (Spearman $\rho = 0.879$; Pearson $r = 0.908$, using gains averaged across training runs). Conditions with more aligned response directions tend to provide larger transfer gains under this protocol. This association supports the task relevance of the measured response relationships. Appendix C reports the separate short-budget protocol, transfer visualizations, and results.

Table 1 reports the terminal sets obtained by restarting Algorithm 1 from all five conditions at each listed λ. Fog is removed first for every positive value shown. Increasing penalties yield three- and two-condition outputs, then rain alone.

Table 1: **Condition selection.** Representative λ values for all distinct terminal sets; each run restarts from the full set using the full-precision matrix. K counts retained conditions; R/S/F/D/L denote rain/snow/fog/dust/low light.

λ	Deletion order	Retained conditions	K
0	–	Rain, Snow, Fog, Dust, Low light	5
0.3	F	Rain, Snow, Dust, Low light	4
0.6	F $\rightarrow$ R	Snow, Dust, Low light	3
0.8	F $\rightarrow$ S	Rain, Dust, Low light	3
2	F $\rightarrow$ S $\rightarrow$ R	Dust, Low light	2
4	F $\rightarrow$ S $\rightarrow$ D	Rain, Low light	2
6	F $\rightarrow$ S $\rightarrow$ D $\rightarrow$ L	Rain	1

5.4 Main Detection Results

Table 2 compares the algorithm outputs with Clean-only and Full generation. All selected two- to four-condition configurations have higher mean AP than both baselines under both protocols. RSDL has the highest internal mean among these outputs (74.59), and SDL has the highest external mean (34.88), exceeding Full generation by 1.53 and 0.80 points, respectively. Rain alone exceeds Clean-only under both protocols but falls below Full generation externally, indicating that more aggressive pruning does not necessarily improve downstream performance.

Rankings by mean external AP over all subsets include at least one algorithm output in the top two for each size $K \in \{1, 2, 3, 4\}$ (Appendix B); these rankings are not inputs to selection. Among the four-condition subsets, RSDL has the highest internal mean and shares the highest external mean. The two-condition outputs DL and RL rank first and second, respectively, in external mean AP among all ten pairs.

For the matched-ratio comparison, Full generation retains all five conditions and the same image pool. The sampler reduces acceptance of generated images uniformly across conditions to match each subset's clean/generated proportion, while keeping the training-update budget unchanged. Under this control, RSDL, SDL, DL, and R exceed Full generation by 0.30, 2.11, 0.22, and 0.81 external AP points, respectively (Appendix B.2). Internal evaluation shares the source domain and generation pipeline with training, whereas the external benchmark tests transfer across scenes and image sources. Together, these comparisons support the practical value of response-guided condition selection under the evaluated protocols.

Table 2: **Main detection results with RT-DETRv2-S.** Mean AP under the protocols in Section 5.1. All distinct algorithm outputs are shown; the $\lambda = 0$ output is listed once as Full generation. K denotes the number of generated conditions. Bold indicates the highest mean in each AP column among the rows shown. Appendix B reports results for all subsets and individual training runs.

Configuration	K	Retained conditions	Internal AP ↑	External AP ↑
Clean-only	0	–	72.23	31.93
Full generation	5	All five	73.06	34.08
Ours, $\lambda = 0.3$	4	Rain, Snow, Dust, Low light	**74.59**	34.64
Ours, $\lambda = 0.6$	3	Snow, Dust, Low light	73.49	**34.88**
Ours, $\lambda = 0.8$	3	Rain, Dust, Low light	73.80	34.36
Ours, $\lambda = 2$	2	Dust, Low light	73.44	34.60
Ours, $\lambda = 4$	2	Rain, Low light	73.55	34.29
Ours, $\lambda = 6$	1	Rain	73.83	33.89

5.5 Evaluation with YOLO11m

We recompute the response matrix with a frozen YOLO11m detector and apply the same condition-selection rule. Downstream evaluation uses the same external training pools, test sets, and AP aggregation as RT-DETRv2-S, with three training seeds. Table 3 reports all distinct terminal sets. The selected two- to four-condition subsets exceed Clean-only by 0.62–0.71 AP points and Full generation by 1.05–1.14 points. Appendix A provides the detector-specific implementation, response matrix, and complete per-seed results.

Table 3: **Condition selection and external AP with YOLO11m.** Mean AP over random seeds. Bold marks the highest mean among the rows shown; condition abbreviations follow Table 1.

Configuration	λ	K	Conditions	External AP ↑	Rank within K
Clean-only	–	0	–	28.53	–
Full generation	0	5	All five	28.10	–
Ours	0.1	4	R,S,F,D	29.18	1/5
Ours	0.3	3	R,S,D	**29.24**	3/10
Ours	1	2	R,D	29.15	2/10

6 Conclusion

We combined two-stage decoupled generation with detector-guided condition selection for UAV detection. Context responses guide a coverage-redundancy criterion for selecting compact condition sets. With RT-DETRv2-S and YOLO11m, the selected two- to four-condition configurations achieve a higher external mean AP than Clean-only and Full generation. These results support detector-guided condition selection for generative augmentation.

AI USE STATEMENT

In this work, we used models such as Flux.2-dev to generate synthetic datasets and LLM-based tools to assist with implementing parts of our method. We also used translation software to assist with translation. Additionally, AI tools were used to identify relevant literature and to edit and polish the manuscript. All generated images underwent manual review, and we performed the quality assessment and qualitative analysis ourselves. All AI-assisted code, AI-suggested references, and AI-assisted manuscript edits were also manually reviewed. We take responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

Our code is provided in the supplementary material.

REFERENCES

Black Forest Labs. FLUX.2: Frontier Visual Intelligence. https://bfl.ai/blog/flux-2, 2025.

Black Forest Labs. FLUX.2 [klein]: Towards Interactive Visual Intelligence. https://bfl.ai/blog/flux2-klein-towards-interactive-visual-intelligence, January 2026.

Kai Chen, Enze Xie, Zhe Chen, Yibo Wang, Lanqing HONG, Zhenguo Li, and Dit-Yan Yeung. Geodiffusion: Text-prompted geometric control for object detection data generation. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 8979–9001, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/25886d7a7cf4e33fd44072a0cd81bf30-Paper-Conference.pdf.

Yu Guo, Shengfeng He, Yuxu Lu, Haonan An, Yihang Tao, Huilin Zhu, Jingxian Liu, and Yuguang Fang. Neptune-x: Active x-to-maritime generation for universal maritime object detection. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 146587–146614. Curran Associates, Inc., 2025. doi: 10.52202/085713-4903. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/d7b0baefb84b8ddf6fbf6ec0f5d4fda3-Paper-Conference.pdf.

Unai Gurbindo, Axel Brando, Jaume Abella, and Caroline König. Object detection in adverse weather conditions for autonomous vehicles using Instruct Pix2Pix. In *2025 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2025. doi: 10.1109/IJCNN64981.2025.11228623. URL https://doi.org/10.1109/IJCNN64981.2025.11228623.

Huang Hsiang and Yung-Yuan Chen. Development of an effective corruption-related scenario-based testing approach for robustness verification and enhancement of perception systems in autonomous driving. *Sensors*, 24(1):301, 2024. doi: 10.3390/s24010301. URL https://www.mdpi.com/1424-8220/24/1/301.

Glenn Jocher and Jing Qiu. Ultralytics YOLO11, 2024. URL https://github.com/ultralytics/ultralytics.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4015–4026, October 2023.

Tamara R. Lenhard, Andreas Weinmann, and Tobias Koch. Beyond clear skies: Synthetic seasonal and weather variations for real-world drone detection, 2026. URL https://arxiv.org/abs/2608.16191. arXiv preprint arXiv:2608.16191.

Wenhao Li, Zimeng Wu, Yu Wu, Zehua Fu, and Jiaxin Chen. Visual Prototype Conditioned Focal Region Generation for UAV-Based Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3772–3782, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Li_Visual_Prototype_Conditioned_Focal_Region_Generation_for_UAV-Based_Object_Detection_CVPR_2026_paper.html.

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 *Computer Vision – ECCV 2014*, pp. 740–755, 2014. doi: 10.1007/978-3-319-10602-1_48. URL https://doi.org/10.1007/978-3-319-10602-1_48.

Wenyu Lv, Yian Zhao, Qinyao Chang, Kui Huang, Guanzhong Wang, and Yi Liu. RT-DETRv2: Improved baseline with bag-of-freebies for real-time detection transformer, 2024. URL https://arxiv.org/abs/2407.17140. arXiv preprint arXiv:2407.17140.

Eric Mintun, Alexander Kirillov, and Saining Xie. On interaction between augmentations and corruptions in natural corruption robustness. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 3571–3583. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/1d49780520898fe37f0cd6b41c5311bf-Paper.pdf.

Samarth Mishra, Rameswar Panda, Cheng Perng Phoo, Chun-Fu (Richard) Chen, Leonid Karlinsky, Kate Saenko, Venkatesh Saligrama, and Rogerio S. Feris. Task2Sim: Towards Effective Pre-Training and Transfer From Synthetic Data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9194–9204, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Mishra_Task2Sim_Towards_Effective_Pre-Training_and_Transfer_From_Synthetic_Data_CVPR_2022_paper.html.

Adnan Munir, Abdul Jabbar Siddiqui, Saeed Anwar, Aiman El-Maleh, Ayaz H. Khan, and Aqsa Rehman. Impact of adverse weather and image distortions on vision-based UAV detection: A performance evaluation of deep learning models. *Drones*, 8(11):638, 2024. doi: 10.3390/drones8110638. URL https://www.mdpi.com/2504-446X/8/11/638.

Yuwei Ning, Liangzhi Wang, Yi Xiao, Zhenhua Wu, Yun Pang, Mingkun Chang, Jichang Li, and Guanbin Li. RealWeather: Realistic and scene-faithful weather translation with driving world models, 2026. URL https://arxiv.org/abs/2608.02953. arXiv preprint arXiv:2608.02953.

Chenghao Qian, Yuhu Guo, Yuhong Mo, and Wenjing Li. Weatherdg: Llm-assisted procedural weather generation for domain-generalized semantic segmentation. *IEEE Robotics and Automation Letters*, 10(6):5919–5926, 2025. doi: 10.1109/LRA.2025.3559821.

Simuletic Team. Simuletic synthetic drone swarm & saturation attack dataset. Dataset, 2025. URL https://huggingface.co/datasets/Simuletic/Military-Drone-Swarm-Saturation-Attack-Dataset.

Izzeddin Teeti, Valentina Musat, Salman Khan, Alexander Rast, Fabio Cuzzolin, and Andrew Bradley. Vision in adverse weather: Augmentation using CycleGANs with various object detectors for robust perception in autonomous racing. In *ICML Workshop on Safe Learning for Autonomous Driving*, 2022. URL https://izzeddin.net/publication/22_icml_cyclegan/.

Maxime Tremblay, Shirsendu Sukanta Halder, Raoul de Charette, and Jean-François Lalonde. Rain rendering for evaluating and improving robustness to bad weather. *International Journal of Computer Vision*, 129(2):341–360, 2021. doi: 10.1007/s11263-020-01366-3. URL https://arxiv.org/abs/2009.03683.

Yibo Wang, Ruiyuan Gao, Kai Chen, Kaiqiang Zhou, Yingjie Cai, Lanqing Hong, Zhenguo Li, Lihui Jiang, Dit-Yan Yeung, Qiang Xu, and Kai Zhang. Detdiffusion: Synergizing generative and

perceptive models for enhanced data generation and perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7246–7255, June 2024.

Kai Wei, Rishabh Iyer, and Jeff Bilmes. Submodularity in data subset selection and active learning. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1954–1963, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/wei15.html.

Jie Zhao, Jingshu Zhang, Dongdong Li, and Dong Wang. Vision-based anti-UAV detection and tracking. *IEEE Transactions on Intelligent Transportation Systems*, 23(12):25323–25334, 2022. doi: 10.1109/TITS.2022.3177627. URL https://doi.org/10.1109/TITS.2022.3177627.

Yian Zhao, Wenyu Lv, Shangliang Xu, Jinman Wei, Guanzhong Wang, Qingqing Dang, Yi Liu, and Jie Chen. DETRs beat YOLOs on real-time object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16965–16974, June 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Zhao_DETRs_Beat_YOLOs_on_Real-time_Object_Detection_CVPR_2024_paper.html.

Ye Zheng, Zhang Chen, Dailin Lv, Zhixing Li, Zhenzhong Lan, and Shiyu Zhao. Air-to-Air Visual Detection of Micro-UAVs: An Experimental Evaluation of Deep Learning. *IEEE Robotics and Automation Letters*, 6(2):1020–1027, 2021. doi: 10.1109/LRA.2021.3056059. URL https://cris.technion.ac.il/en/publications/air-to-air-visual-detection-of-micro-uavs-an-experimental-evaluat/.

Jingyuan Zhu, Shiyu Li, Yuxuan Liu, Jian Yuan, Ping Huang, Jiulong Shan, and Huimin Ma. Odgen: Domain-specific object detection data generation with diffusion models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 63599–63633. Curran Associates, Inc., 2024. doi: 10.52202/079017-2031. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/743771397cef2aa0ef497c428c3a46b7-Paper-Conference.pdf.

Yixin Zhu, Zuo-Liang Zhu, Jian Yang, Miloš Hašan, Jin Xie, and Beibei Wang. IntrinsicWeather: Controllable weather editing in intrinsic space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 30772–30781, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Zhu_IntrinsicWeather_Controllable_Weather_Editing_in_Intrinsic_Space_CVPR_2026_paper.html.

A Additional Evaluation with YOLO11m

A.1 Experimental Setup and Response Features

The training pools and evaluation protocol follow Section 5.5. Starting from COCO-pretrained weights, each detector is trained with AdamW, batch size 16, and an initial learning rate of 3×10^{-4} for 5,500 successful optimizer updates. The learning-rate schedule uses 300 warmup updates followed by cosine decay defined for a 9,000-update run; training uses the first 5,500 updates of this schedule. Images are resized to 960×960 without random augmentation, and evaluation uses the final EMA weights. Mixed-precision training computes CIoU geometry in FP32. For response measurement, we freeze a separate detector trained on the 409 additional clean training images at its final EMA endpoint and compute similarities on the same 102 sources and their generated variants. We use an internal activation in the stride-16 P4 neck fusion block of YOLO11m (Jocher & Qiu, 2024). As in RT-DETRv2-S, the measurement uses a detection-trained spatial representation within cross-scale fusion, before prediction-specific heads. YOLO pooling uses the full image outside the target-protection region, rather than the target-centered window used for RT-DETRv2-S. In both cases, background pooling specifies where features are read; their receptive fields can still include the UAV. We measure generated-minus-clean feature changes and average within-source cosine similarities across sources.

A.2 Response Matrix and Condition Selection

Figure 5 shows the YOLO context-response similarity matrix. Dust and low light have the largest off-diagonal similarity (0.823), followed by fog–dust (0.766) and fog–low light (0.749). Rain is weakly aligned with snow (0.008) and fog (0.053), and negatively aligned with dust (-0.264) and low light (-0.145). The matrix differs from the RT-DETRv2-S matrix in Figure 4. The two matrices use their respective detector representations and pooling supports and do not define a detector-independent ordering of conditions.

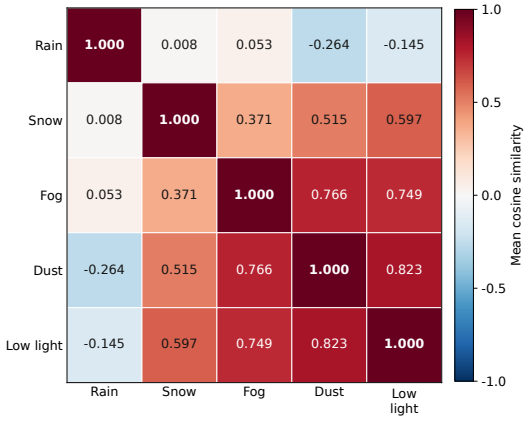


Figure 5: **YOLO context-response similarity matrix.** Each entry is the mean cosine similarity between generated-minus-clean background-pooled feature responses for a pair of conditions, computed within each of 102 training source images and averaged across sources.

We apply the objective in Equation (10) and the same backward greedy rule to this matrix, retaining signed similarities without clipping them to zero. Negative pairs reduce the signed redundancy sum $R(\mathcal{A})$ and therefore favor retaining oppositely directed responses. A negative cosine similarity does not establish negative training transfer. Table 3 in Section 5.5 reports all distinct terminal sets. Each run restarts from all five conditions using the full-precision matrix and the same tie-breaking order. The deletion sequences are L at $\lambda = 0.1$, L $\rightarrow$ F at $\lambda = 0.3$, and L $\rightarrow$ F $\rightarrow$ S at $\lambda = 1$.

A.3 External Detection Results

Table 4 reports the external AP for all 32 training configurations, including the results for each seed and the mean across seeds.

Table 4: **YOLO11m external AP for all 32 configurations.** Mean is computed over seeds 0, 1, and 2. Rank is defined separately for each condition count K; the two baselines are unranked. † marks representative outputs from Table 3. Bold marks the highest external mean within each K.

Rank	Conditions	Seed 0	Seed 1	Seed 2	Mean
Baselines					
—	Clean-only	28.37	28.42	28.80	28.53
—	Full generation†	28.12	27.60	28.57	28.10
$K = 4$ *generated conditions*					
1	R,S,F,D†	28.48	29.86	29.19	**29.18**
2	S,F,D,L	29.02	28.57	29.60	29.06
3	R,F,D,L	28.62	28.25	29.29	28.72
4	R,S,D,L	28.34	29.32	28.40	28.69
5	R,S,F,L	28.08	29.44	28.16	28.56
$K = 3$ *generated conditions*					
1	S,F,D	28.95	30.32	30.32	**29.86**
2	S,D,L	28.59	30.28	29.49	29.45
3	R,S,D†	29.68	29.24	28.80	29.24
4	R,S,F	29.31	27.74	29.93	28.99
5	R,F,D	28.04	28.98	29.31	28.77
6	S,F,L	29.12	27.76	29.13	28.67
7	R,S,L	27.66	30.02	27.78	28.49
8	F,D,L	28.41	27.95	28.83	28.39
9	R,D,L	26.50	28.76	28.56	27.94
10	R,F,L	28.14	27.47	28.17	27.93
$K = 2$ *generated conditions*					
1	S,D	29.07	29.70	30.68	**29.81**
2	R,D†	29.80	29.66	27.99	29.15
3	R,L	29.97	28.56	28.88	29.13
4	F,D	28.00	29.12	29.74	28.95
5	S,F	28.31	28.74	29.63	28.89
6	F,L	28.75	29.02	28.65	28.81
7	R,F	28.68	30.45	27.00	28.71
8	R,S	27.04	29.43	29.23	28.56
9	S,L	27.69	29.80	28.13	28.54
10	D,L	27.11	27.91	27.19	27.40
$K = 1$ *generated conditions*					
1	S	27.86	29.26	30.28	**29.14**
2	F	28.17	29.43	29.66	29.09
3	D	28.51	28.15	28.96	28.54
4	R	28.14	27.88	29.28	28.43
5	L	26.97	28.80	27.36	27.71

B PER-SEED DETECTION RESULTS IN MAIN EXPERIMENTS

B.1 RESULTS FOR ALL TRAINING CONFIGURATIONS

Table 5 reports the results for Clean-only, Full generation, and the 30 subsets containing between one and four generated conditions. Within each K, subsets are ranked in descending order of the average external AP, calculated from the seed scores. These rankings summarize the evaluation results; they are not inputs for Algorithm 1. The internal and external results come from separate model fittings, even when their seed identifiers coincide. All scores are expressed as AP on the 0–100 scale. R/S/F/D/L denote rain/snow/fog/dust/low light. A dagger (†) marks a terminal set returned for a representative value of λ in Table 1. Full generation is also the $\lambda = 0$ output.

Table 5: **Per-seed aggregate AP and within-size external rankings.** Rank is defined separately for each condition count K; the two baselines are unranked. † marks algorithm outputs. Bold marks the highest external mean within each K.

		Internal AP				External AP			
Rank	Conditions	Seed 0	Seed 1	Seed 2	Mean	Seed 0	Seed 1	Seed 2	Mean
Baselines									
—	Clean-only	72.48	72.09	72.13	72.23	29.61	31.63	34.55	31.93
—	Full generation†	74.14	72.79	72.25	73.06	33.27	36.09	32.89	34.08
$K = 4$ *generated conditions*									
1	R,S,F,L	72.99	73.22	73.95	73.39	32.21	37.68	34.03	**34.64**
1	R,S,D,L†	74.04	74.87	74.85	74.59	32.01	36.34	35.57	**34.64**
3	S,F,D,L	74.60	73.02	74.00	73.88	34.51	33.09	32.86	33.49
4	R,S,F,D	72.73	73.33	73.93	73.33	32.45	32.07	34.29	32.94
5	R,F,D,L	74.20	73.38	73.49	73.69	27.45	37.86	31.05	32.12
$K = 3$ *generated conditions*									
1	R,F,D	73.58	73.43	73.86	73.62	32.63	36.88	35.73	**35.08**
2	S,D,L†	73.55	73.10	73.83	73.49	31.35	38.69	34.59	34.88
3	S,F,L	73.84	74.15	74.11	74.03	32.96	36.60	34.28	34.61
4	R,F,L	74.41	72.72	73.92	73.68	32.90	36.78	33.48	34.39
5	R,D,L†	73.41	73.97	74.00	73.80	34.78	36.79	31.51	34.36
6	S,F,D	74.41	73.75	74.03	74.06	33.53	35.26	33.58	34.12
7	R,S,L	73.25	73.26	73.90	73.47	32.44	37.04	32.30	33.93
8	R,S,D	74.30	72.32	73.38	73.33	31.63	36.09	33.64	33.79
9	F,D,L	74.19	73.32	72.85	73.46	31.32	36.24	33.15	33.57
10	R,S,F	73.04	72.80	72.84	72.89	33.81	35.79	30.93	33.51
$K = 2$ *generated conditions*									
1	D,L†	73.53	74.07	72.71	73.44	32.58	37.13	34.08	**34.60**
2	R,L†	73.00	73.65	74.01	73.55	34.47	35.87	32.52	34.29
3	F,D	73.45	74.35	73.45	73.75	33.60	36.03	32.18	33.94
4	F,L	73.62	73.01	73.20	73.28	31.26	38.30	31.84	33.80
5	R,F	74.82	73.05	73.62	73.83	31.81	35.95	33.13	33.63
6	R,S	73.62	73.76	73.24	73.54	30.61	37.28	31.87	33.25
7	R,D	74.16	74.40	73.75	74.11	31.22	36.55	31.79	33.19
8	S,L	74.30	73.92	72.87	73.70	28.27	36.75	33.67	32.90
9	S,F	73.32	73.96	74.03	73.77	31.59	35.42	30.03	32.35
10	S,D	73.77	74.42	73.02	73.74	29.71	32.79	33.74	32.08
$K = 1$ *generated condition*									
1	S	72.91	73.89	74.27	73.69	31.06	36.05	35.95	**34.36**
2	R†	74.52	73.52	73.44	73.83	32.76	37.12	31.79	33.89
3	L	73.64	73.44	72.17	73.08	29.42	38.42	32.78	33.54
4	D	74.63	73.94	73.25	73.94	30.12	35.58	32.48	32.73
5	F	72.06	73.05	72.57	72.56	28.08	34.09	32.94	31.70

B.2 External Evaluation at Matched Sampling Ratios

Under uniform image sampling, a subset of K conditions contributes $146K$ generated images alongside 730 clean images, giving a generated sampling proportion of $p_{\text{gen}} = K/(5 + K)$. To control this proportion, we retain all five conditions in Full generation and reduce its generated sampling proportion to the corresponding value for each K. The underlying image pool is unchanged: the sampler thins the original shuffled stream by retaining clean entries and reducing the acceptance rate of generated entries uniformly across the five conditions. We follow the external training protocol in Section 5.1, using 6,000 successful optimizer updates, final EMA evaluation, and seeds 0, 1, and 2.

Table 6: **Full generation at matched clean/generated sampling ratios.** K denotes the size of the selected subset whose sampling ratio is matched; Full generation retains all five conditions in every row. Scores are external AP, defined as the equal-weight mean of AP on the pooled real test set and the synthetic test set. Mean averages seeds 0, 1, and 2.

Matched K	p_{gen}	Seed 0	Seed 1	Seed 2	Mean
4	$4/9$ (44.44%)	32.04	36.21	34.78	34.34
3	$3/8$ (37.50%)	31.23	37.08	30.01	32.77
2	$2/7$ (28.57%)	32.57	35.78	34.80	34.38
1	$1/6$ (16.67%)	32.62	32.26	34.36	33.08

Comparing Table 6 with the algorithm outputs in Table 2, the RSDL, SDL, DL, and R outputs exceed the corresponding matched-ratio Full means by 0.30, 2.11, 0.22, and 0.81 AP points, respectively. The alternative three-condition output RDL also exceeds its matched-ratio Full baseline, whereas the two-condition output RL is slightly lower (34.29 versus 34.38).

C Response Similarity and Cross-Condition Transfer

We relate the context-response matrix used for condition selection (Figure 4) to single-condition transfer measurements. The response matrix comes from the detector trained on 409 clean images with seed 0 and frozen after 6,000 successful updates; the transfer models follow the separate protocol below. The transfer measurements are used only for this supporting analysis, not to construct the response matrix or select conditions.Figure 6 shows the directed transfer gains and the relationship between symmetric transfer gains and context-response similarity.

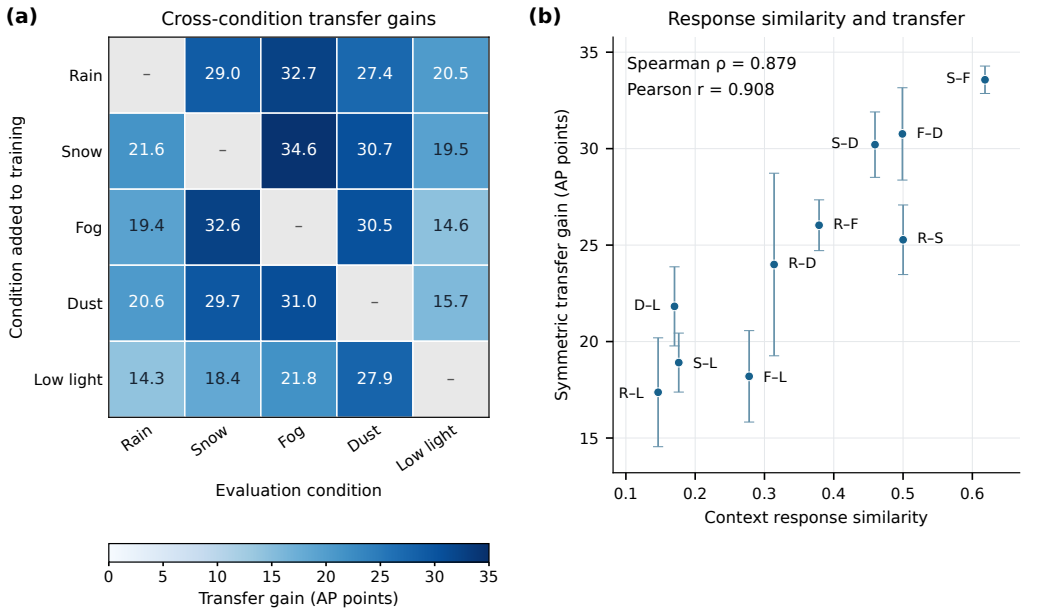


Figure 6: **Cross-condition transfer.** (a) Directed transfer gains averaged over seeds 1, 2, and 3. Rows identify the condition added to clean training data, and columns identify the evaluation condition. Entries are AP-point improvements over a duplicated-clean control, not response similarities; self-condition entries are omitted. (b) Mean symmetric transfer gains versus context-response similarities for the ten unordered condition pairs. Error bars show sample standard deviations across the three transfer-training seeds. R/S/F/D/L denote rain/snow/fog/dust/low light.

Auxiliary training protocol. For each seed $s \in \{1, 2, 3\}$, five RT-DETRv2-S models are trained on 102 clean sources and their 102 generated variants under one condition; a sixth model uses two copies of the clean sources as a matched 204-record control. All models start independently from the same official COCO-pretrained weights. Each fit runs for 400 attempted updates with batch size 17 (6,800 image presentations), using AdamW with learning rate and weight decay 10^{-4}, 50 warmup attempts, and a tenfold learning-rate reduction after attempt 300. Inputs are resized to 960×960 without random image augmentation. EMA and scheduling advance on every attempt, including mixed-precision optimizer skips. Among EMA checkpoints at attempts $50, 100, \ldots, 400$, the earliest maximum AP on 22 clean-validation images selects the checkpoint. Each selected model is evaluated on the five generated-condition versions of those 22 validation sources, yielding 90 AP measurements in total; neither internal test nor external test images are used.

Transfer measures. Let $f_a^{(s)}$ be the model trained with condition a, and $f_{\text{ctrl}}^{(s)}$ its same-seed duplicated-clean control. For $a \neq b$, the directed transfer gain is

$$g_{a \to b}^{(s)} = \text{AP}_b\left(f_a^{(s)}\right) - \text{AP}_b\left(f_{\text{ctrl}}^{(s)}\right), \tag{12}$$

Table 7: **Symmetric cross-condition transfer gains.** S_{ab} is the context-response similarity used for selection. Gains are AP points, and Mean averages seeds 1, 2, and 3. All calculations use full-precision inputs.

Pair	S_{ab}	Seed 1	Seed 2	Seed 3	Mean
R–S	0.500	27.29	23.80	24.75	25.28
R–F	0.379	27.55	25.33	25.21	26.03
R–D	0.314	29.14	19.82	23.03	23.99
R–L	0.146	20.41	14.84	16.87	17.37
S–F	0.618	32.79	33.73	34.19	33.57
S–D	0.460	30.63	31.65	28.34	30.21
S–L	0.176	20.00	19.57	17.16	18.91
F–D	0.499	32.83	31.32	28.14	30.76
F–L	0.278	17.82	20.73	16.04	18.20
D–L	0.170	22.81	23.19	19.47	21.82

where AP uses the 0–100 scale, so gains are absolute AP points. For each unordered pair, we average the two directions and then the three seeds:

$$G_{ab}^{(s)} = \frac{g_{a\to b}^{(s)} + g_{b\to a}^{(s)}}{2}, \qquad \overline{G}_{ab} = \frac{1}{3}\sum_{s=1}^{3} G_{ab}^{(s)}. \tag{13}$$

The primary correlations compare S_{ab} with $\overline{G}_{ab}$ over the ten unordered pairs, excluding the diagonal. We compute correlations after averaging gain values, rather than averaging correlation coefficients. Pearson r is calculated from the raw similarity and gain values; Spearman ρ uses their average ranks. The directed analysis retains all 20 off-diagonal directions, associating the same symmetric similarity $S_{ab} = S_{ba}$ with each direction separately.

Detailed results. Table 7 reports the symmetric gains for each training seed and their mean. For context responses, the mean-gain correlations are $\rho = 0.879$ and $r = 0.908$ for symmetric transfer, and $\rho = 0.737$ and $r = 0.753$ for directed transfer (Table 8). The symmetric per-seed Spearman correlations range from 0.733 to 0.830. For completeness, the table also reports mean-gain correlations for object-pooled responses, using the same frozen detector and paired-difference calculation. Object pooling uses the saved binary source-UAV segmentation masks (SAM core masks), from which the protection masks are derived by dilation. These binary masks are mapped directly from the source-image grid to the feature grid by area averaging. For RT-DETRv2-S context pooling, the binary protection masks are resized to the 960×960 detector-input grid using nearest-neighbor interpolation and further dilated with a 9×9 square kernel, corresponding to a radius of four pixels. The pooling window is centered on the source bounding box. Its side length is four times the longer box side on this grid, with a minimum of 64 pixels, and the window is clipped at image boundaries. The remaining region is mapped to the feature grid by area averaging.

Table 8: **Response–transfer correlations.** Mean-gain rows correlate similarities with gains averaged over seeds; seed rows use that seed's gains with the same fixed response matrix. n counts condition pairs or directed entries, not independent experimental replicates. Object-pooled results are secondary comparisons.

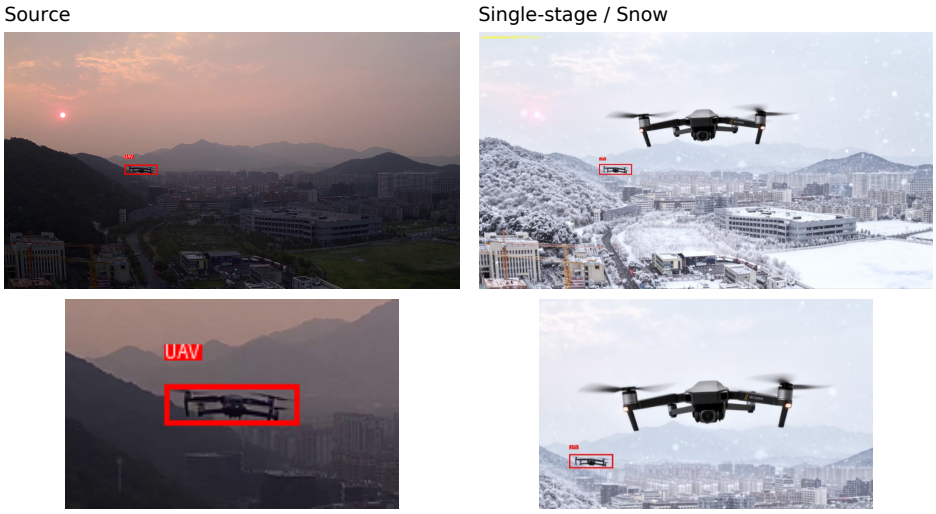
Response	Transfer	n	Gain values	Spearman ρ	Pearson r
Context	Symmetric	10	Mean gains	0.879	0.908
Context	Symmetric	10	Seed 1	0.733	0.843
Context	Symmetric	10	Seed 2	0.830	0.853
Context	Symmetric	10	Seed 3	0.830	0.921
Context	Directed	20	Mean gains	0.737	0.753
Context	Directed	20	Seed 1	0.625	0.690
Context	Directed	20	Seed 2	0.580	0.615
Context	Directed	20	Seed 3	0.764	0.821
Object	Symmetric	10	Mean gains	0.855	0.903
Object	Directed	20	Mean gains	0.694	0.748

D ADDITIONAL SINGLE-STAGE GENERATION EXAMPLES

Figure 7 presents additional single-stage editing examples. Panel (a) enlarges and displaces the source UAV, while panel (b) introduces an additional oversized UAV.



(a) Target enlargement and displacement



(b) Additional UAV generation

Figure 7: **Additional single-stage editing examples.** Top rows show source images and single-stage snow outputs; bottom rows show close-ups. Red boxes show the source ground-truth UAV boxes mapped to the corresponding displayed images and crops. (a) Matched crop windows and display scales reveal target enlargement and displacement. (b) The output close-up highlights the additional generated UAV; the source and output close-ups use different crop regions.

E TWO-STAGE GENERATION WITH A SECOND GENERATOR

We apply the pipeline in Section 4.1 with pretrained FLUX.2-klein-base-9B (Black Forest Labs, 2026), without fine-tuning or adapters. We generate rain, snow, fog, and dust variants for the 22 generation sources in the internal validation partition, yielding 88 source–condition pairs. Figure 8 shows one selected source.

For the illustrated comparison, both generators use 28 steps, guidance 3.0, and seed 43. The dev and klein background outputs are 1024×576 and 512×288, respectively; klein inference uses BF16. On the 1024×576 preparation grid, we threshold the stored 8-bit protection masks at > 127 without further dilation. Pillow GaussianBlur radii are 28 pixels for masks and 60 pixels for source images. Both outputs are blended with the same source image and soft mask at 512×288, using Lanczos resampling for images and nearest-neighbor resampling for masks. Soft blending can reduce target contrast.



(a) Source reference



(b) FLUX.2-dev



(c) FLUX.2-klein-base-9B

Figure 8: **Two-stage generation with two image editors.** (a) Clean source reference. (b,c) Final outputs after background editing and source-UAV soft-mask blending under four weather conditions. Red boxes show the source ground-truth annotation mapped to the shared display grid. Lower-right insets show the same source-defined target region at the same magnification.