Do Corruption-Family Text Residuals Help Zero-Shot CLIP? A Controlled Negative Result on CIFAR-C

Anonymous Authors

Abstract

Zero-shot CLIP is a strong frozen baseline, but its robustness under common corruptions remains limited. We study a narrow question: does prompt-side text that names a corruption family contain useful family-specific signal beyond simpler prompt baselines? Starting from frozen CLIP text prototypes, we build a bank of four corruption-family residual vectors for gaussian noise, motion blur, fog, and JPEG compression, combine them with a lightweight family posterior, and evaluate the resulting calibration rule against zero-shot CLIP, clean prompt ensembling, naive corruption prompts, a generic low-quality residual, and random family reassignment. A pilot on synthetic corruptions shows that the residual bank is not arbitrary: averaged over seeds, the family posterior reaches 34.87% accuracy and the matched-minus-mismatched alignment margin is 0.0806, compared with 0.0281 for random reassignment. However, this signal does not translate into better benchmark accuracy. On CIFAR-10-C, the family-residual method reaches 74.9985 mean corruption accuracy, essentially matching zero-shot CLIP at 75.0000, the generic residual control at 75.0140, and random reassignment at 74.9993, while trailing naive corruption prompts at 75.0500. On CIFAR-100-C, it reaches 50.2817, again matching zero-shot CLIP at 50.2760 and random reassignment at 50.2807 while trailing naive corruption prompts at 50.8675. The executed study deviates from the original confirmation plan by reselecting λ per seed rather than freezing one global value, so we interpret the paper as an executed controlled negative result, not as a strict global-freeze confirmation. Within that narrower scope, prompt-side corruption language mostly captures generic degradation rather than deployable family-specific structure.

1 Introduction

Zero-shot CLIP enables practical open-vocabulary recognition with no task-specific training, which makes it an attractive baseline for robustness studies [**?**]. Common-corruption benchmarks such as CIFAR-10-C and CIFAR-100-C were introduced precisely because standard image classifiers can fail sharply under seemingly small distribution shifts [**?**]. Frozen vision-language models inherit that brittleness, and a growing literature now explores prompt tuning, prompt distribution learning, auto-tuning, and test-time adaptation to improve robustness [**????**]. Within that landscape, a small prompt-only modification must clear a high bar: it should outperform simpler prompt baselines and show evidence that it captures structure not already explained by generic image degradation.

This paper studies a deliberately narrow question. Suppose we construct text residuals that explicitly describe corruption families such as gaussian noise or motion blur and add them to clean class prompts at test time. Do those residuals contain usable family-specific information, or do they behave like a generic "low-quality image" direction? We frame this as a controlled baseline study rather than a new adaptation paradigm. The goal is not to maximize raw accuracy, but to test whether named corruption families provide actionable prompt-side structure in a fully frozen CLIP pipeline.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

Our study is designed around falsification. In addition to standard baselines, we include two direct controls: a generic low-quality residual and a random family reassignment control that preserves the same scoring machinery while destroying family identity. We also run a pilot gate before the full benchmark to test whether the residual bank exhibits family alignment at all. The pilot produces a weak positive signal in representation space, but the full benchmark shows that this signal does not produce practically distinct accuracy gains.

One protocol caveat needs to be read up front. The original plan called for one globally frozen configuration across seeds, but the executed study reselected λ per seed on the unlabeled proxy objective. We therefore treat the paper as a controlled executed-study analysis with a weakened confirmatory claim, not as a strict validation of the original global-freeze design.

Our contributions are:

- We formulate corruption-family text residuals as a simple, fully frozen calibration rule for zero-shot CLIP and evaluate them in a controlled setting with explicit generic and random controls.
- We report a complete CIFAR-10-C and CIFAR-100-C study over four corruption families and five severities using the same frozen OpenCLIP backbone and prompt bank throughout, while explicitly flagging the executed per-seed λ reselection that weakens the original confirmation protocol.
- We document a negative result: although corruption-family residuals exhibit modest pilot alignment, they do not outperform naive corruption prompts, generic residuals, or random family reassignment in downstream corruption accuracy.

Section 2 reviews related work, Section 3 describes the method and the executed protocol, Section 4 presents the pilot, benchmark results, and calibration analysis, and Section 5 discusses limitations and implications.

2 Related Work

Vision-language zero-shot classification. CLIP established the frozen image-text interface that makes prompt-side robustness studies possible [**?**]. Subsequent work showed that richer text can improve zero-shot classification by describing class semantics more completely [**?**]. Our work differs in both goal and target signal: we do not refine class descriptions, but instead ask whether prompt-side nuisance descriptions encode family-specific corruption information.

Common-corruption robustness context. Our experiments sit in the common-corruption evaluation tradition initiated by **?**, which introduced CIFAR-10-C and CIFAR-100-C style testing to separate clean accuracy from robustness under corruptions such as noise, blur, weather, and digital artifacts. More recent vision-language robustness analyses suggest that corruption failures are not explained by a single scalar "image quality" axis alone [**?**]. That broader context motivates our narrower question: whether an explicitly named corruption family can be exploited using text-side structure alone.

Prompt adaptation and test-time tuning. Several archival methods adapt prompts or text classifiers using unlabeled test data. Test-Time Prompt Tuning (NeurIPS 2022) optimizes prompts online per batch or sample [**?**]. AutoCLIP (TMLR 2024) tunes zero-shot classifiers automatically from unlabeled data [**?**]. Frolic (NeurIPS 2024) learns a label-free prompt distribution and applies bias correction [**?**]. These approaches aim to improve zero-shot performance broadly, often through optimization or richer prompt distributions. In contrast, our method fixes a tiny hand-specified residual bank and asks whether explicit family variables are useful at all.

Corruption robustness analysis and adaptation. For our specific question, the closest analysis paper is the recent arXiv-only study of corruption robustness in vision-language models by **?**, which argues that corruption sensitivity is structured by corruption category and task rather than by a single uniform degradation trend. BATCLIP is a contemporaneous arXiv-only adaptation paper that asks whether heavier bimodal test-time adaptation can exploit such structure for robustness gains under common corruptions [**?**]. Our work sits between these threads: we test whether a much cheaper

frozen prompt-side mechanism can exploit named corruption families at all. The answer in our setting is largely no.

Debiasing and language-quality interventions. BendVLM is currently arXiv-only, while PRISM appears in the archival ICCV 2025 proceedings; both manipulate embedding geometry to reduce spurious or biased directions [**??**]. Quality Text, Robust Vision is archival (ACM MM 2025) and suggests that text quality can improve robustness without introducing explicit corruption variables [**?**]. Our results align more closely with that perspective: generic prompt-side degradation cues appear at least as effective as explicit family structure in the frozen setting we test.

Positioning. Unlike prior prompt adaptation work, our contribution is not a stronger adaptation algorithm. It is a controlled baseline study that isolates one interpretable hypothesis: if corruption-family text residuals carry deployable family-specific signal, they should beat a generic residual and should break when family labels are randomized. They do not.

3 Method

3.1 Setup

We use frozen OpenCLIP ViT-B/32 with the `laion2b_s34b_b79k` checkpoint. Let $v(x) \in \mathbb{R}^d$ denote the normalized image embedding of test image x, and let $c_y \in \mathbb{R}^d$ denote the normalized clean text prototype for class y.

The single-template zero-shot baseline uses

$$c_y = \mathrm{norm}(e(\texttt{a photo of a \{class\}})), \tag{1}$$

where $e(\cdot)$ is the frozen text encoder. We also evaluate a stronger clean prompt ensemble built from five templates: `a photo of a {class}`, `an image of a {class}`, `a close-up photo of a {class}`, `a centered photo of a {class}`, and `a blurry photo of a {class}`.

The study is restricted to four corruption families: `gaussian_noise`, `motion_blur`, `fog`, and `jpeg_compression`.

3.2 Corruption-Family Residuals

For each family z, we form a shared text residual by averaging paired clean-versus-corrupted prompt differences over all classes:

$$r_z = \mathrm{norm}\left(\frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}} \left[\mathrm{norm}\big(e(p_{y,z}^{\mathrm{corr}})\big) - \mathrm{norm}\big(e(p_y^{\mathrm{clear}})\big) \right] \right), \tag{2}$$

where p_y^{clear} is `a clear photo of a {class}` and $p_{y,z}^{\mathrm{corr}}$ is the family-specific corrupted prompt such as `a photo of a {class} with gaussian noise`. The executed runs use the `base` prompt variant for the final model selection.

Given residual r_z, we define a family-conditioned class prototype

$$t_{y,z} = \mathrm{norm}(c_y + \alpha r_z). \tag{3}$$

3.3 Family Posterior and Final Score

To estimate which corruption family is present, we construct one text query vector per family from prompts such as `an image with gaussian noise corruption`. For image x,

$$q(z \mid x) = \mathrm{softmax}_z\big(\beta\, v(x)^\top q_z\big). \tag{4}$$

The family-residual score for class y is

$$s_{\mathrm{fam}}(y \mid x) = v(x)^\top c_y + \lambda \sum_z q(z \mid x) \left(v(x)^\top t_{y,z} - v(x)^\top c_y \right). \tag{5}$$

This rule is fully frozen and optimization-free at test time.

3.4 Controls and Executed Protocol

We compare against five alternatives.

- **Zero-shot CLIP**: the single-template clean prototype.
- **Clean prompt ensemble**: the five-template clean ensemble.
- **Naive corruption prompt**: class prompts directly conditioned on each corruption family and mixed by the same family posterior.
- **Generic residual control**: a single residual built from `a low-quality photo of a {class}` minus `a clear photo of a {class}`, applied to all families.
- **Random family reassignment**: the same family-residual machinery, but with family labels randomly permuted before residual application.

The last two controls are the critical tests of family specificity. If family residuals are useful because they capture named corruption structure, they should beat both.

The predeclared plan called for one globally frozen configuration across seeds. The executed runs match that plan for the prompt variant and for (β, α), which are globally fixed to `base`, 2.0, and 0.05. However, λ was reselected per seed from the predeclared grid $\{0.25, 0.50, 1.00\}$ on the unlabeled proxy objective, yielding $\{0.5, 1.0, 0.5\}$ for seeds $\{7, 17, 27\}$. Appendix A.1 lists the selected values. This is the main protocol caveat of the paper: the executed study is not a strictly global frozen-confirmation test. We therefore frame the paper as an executed controlled study with a prominent limitation, not as a clean validation of the original fully frozen recipe. The deviation is mildly optimistic for the family-residual method because it adds one seed-local tuning degree of freedom, and the random-reassignment control uses the same per-seed λ values to keep that comparison fair.

4 Experiments

4.1 Experimental Setup

We evaluate on CIFAR-10-C and CIFAR-100-C [**?**], restricted to the four corruption families above and severity levels 1 through 5. Clean CIFAR-10 and CIFAR-100 test sets are also evaluated to check whether corruption-aware prompts damage in-distribution accuracy. All methods use the same frozen OpenCLIP backbone and precomputed image features. The final evaluation uses seeds 7, 17, and 27 for the methods that depend on proxy selection or randomized family assignment; deterministic baselines are run once.

The pilot gate uses synthetic corruptions generated from held-out clean CIFAR-10 images with the same four corruption families. It measures family-posterior accuracy, matched-versus-mismatched alignment between average visual corruption shifts and text residuals, and downstream corrupted-image accuracy against the generic and random controls. The proposal's pilot criterion is not met: `pilot_success` is `false`. We nevertheless complete the restricted CIFAR benchmark because the pilot still shows a measurable representational signal that is worth testing end-to-end.

Primary metrics are clean top-1 accuracy and mean corruption top-1 accuracy over all family-severity pairs. Secondary metrics are expected calibration error (ECE) with 15 equal-width bins and runtime per 10k images from the recorded experiment outputs.

Rounding and aggregation policy. All decimal values in prose and main-paper tables are copied from the experiment artifacts and rounded to the nearest displayed unit: accuracies and ECE values to two decimals, runtimes to three decimals, and alignment margins to four decimals. Seed-dependent rows report arithmetic means over seeds $\{7, 17, 27\}$ before rounding. Appendix tables report rounded copies of the artifact values to six decimals, so some rows that appear numerically identical in the main text differ slightly before rounding.

Prompt-bank sensitivity in the pilot. The pilot search also evaluated a stronger wording bank labeled `strong`. For the best strong-variant configuration per seed, family-posterior accuracy averaged 49.92, compared with 34.87 for the executed `base` runs, but the mean proxy objective was slightly lower at 0.500020 versus 0.500021 for the selected base recipe. Because the final benchmark

Table 1: Pilot-gate results on synthetic CIFAR-10 corruptions. The alignment margin is the difference between matched-family and mismatched-family cosine similarity. The pilot shows some family alignment, but no downstream advantage over the generic or random controls.

Seed	Family posterior	Main align.	Random align.	Pilot acc.	Generic acc.
7	34.15	0.0806	0.0357	59.25	59.45
17	35.20	0.0801	0.0336	59.45	59.50
27	35.25	0.0813	0.0148	60.15	60.15
Mean	34.87	0.0806	0.0281	59.62	59.70

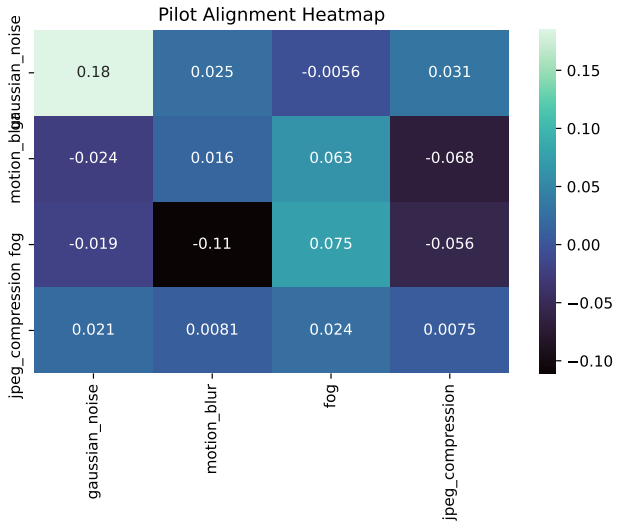

Figure 1: Pilot alignment heatmap between average visual corruption shifts and text residuals. Diagonal structure exists, especially for gaussian noise, but the pilot still fails to produce downstream gains over the stronger controls.

follows the proxy-selected executed recipe, the stronger prompt bank is discussed as an unexecuted pilot sensitivity rather than as a benchmark baseline.

4.2 Pilot Gate

Table 1 shows the pilot evidence. Averaged over seeds, the family posterior reaches 34.87 accuracy, and the family residual bank attains a matched-minus-mismatched cosine alignment margin of 0.0806. This is larger than the random-reassignment margin of 0.0281 and larger than the generic residual's margin of 0.0000. However, the corresponding downstream accuracy gain does not appear: the family-residual pilot accuracy averages 59.62, compared with 59.70 for the generic residual and 59.62 for random reassignment. Clean pilot accuracy is unchanged relative to zero-shot CLIP for every seed.

Figure 1 visualizes the same pattern. The residual bank is not random in representation space, but the structure is too weak to produce useful benchmark gains.

4.3 Main Results

Table 2 presents the main benchmark results. The headline outcome is negative. On CIFAR-10-C, family residuals average 75.00 mean corruption accuracy after rounding, essentially identical to zero-shot CLIP at 75.00, generic residuals at 75.01, and random family reassignment at 75.00. Naive corruption prompts are slightly higher at 75.05. On CIFAR-100-C, family residuals average 50.28, again matching zero-shot CLIP at 50.28, generic residuals at 50.27, and random family reassignment at 50.28, while naive corruption prompts are materially better at 50.87.

Table 2: Main results on CIFAR-10/CIFAR-10-C and CIFAR-100/CIFAR-100-C. Higher is better for accuracies. Seed-dependent rows report mean ± standard deviation over seeds 7, 17, and 27. Best corruption accuracy in each dataset is in **bold**.

Method	CIFAR-10 clean	CIFAR-10-C mean	CIFAR-100 clean	CIFAR-100-C mean	Runtime / 10k images (C10/C100)
Zero-shot CLIP	93.66	75.00	75.85	50.28	0.005 / 0.008
Clean prompt ensemble	93.44	74.79	75.58	50.01	0.006 / 0.008
Naive corruption prompt	93.57	**75.05**	75.25	**50.87**	0.005 / 0.015
Generic residual control	93.67	75.01	75.88	50.27	0.006 / 0.013
Family residual	**93.68 ± 0.00**	75.00 ± 0.00	**75.89 ± 0.01**	50.28 ± 0.00	0.007 / 0.019
Random family reassignment	**93.68 ± 0.00**	75.00 ± 0.00	**75.89 ± 0.01**	50.28 ± 0.00	0.007 / 0.019

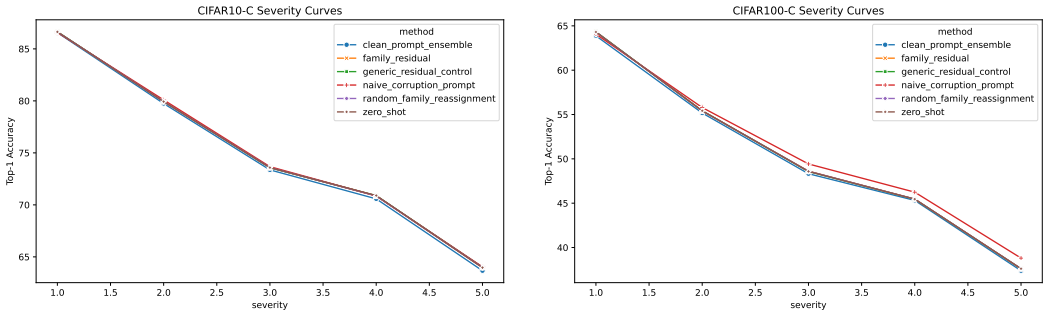

Figure 2: Per-severity corruption accuracy on CIFAR-10-C and CIFAR-100-C for the six plotted methods: zero-shot CLIP, clean prompt ensemble, naive corruption prompt, generic residual control, random family reassignment, and family residual. The family-residual method tracks zero-shot CLIP almost exactly across the full severity range, while naive corruption prompts sit slightly higher than the rest, especially on CIFAR-100-C.

The method is not harmful to clean accuracy: family residuals average 93.68 on CIFAR-10 clean versus 93.66 for zero-shot CLIP, and 75.89 on CIFAR-100 clean versus 75.85 for zero-shot CLIP. But the core hypothesis was stronger than "do no harm." The controls show that explicit family identity is not doing useful work.

4.4 Per-Severity and Per-Family Analysis

Figure 2 shows that the lack of improvement is consistent across severities for all six plotted methods. On CIFAR-10-C, family residuals follow the zero-shot curve almost exactly from severity 1 through severity 5, ending at 63.98 compared with 63.96 for zero-shot CLIP, while naive corruption prompts finish at 64.04. On CIFAR-100-C, the family-residual curve is again nearly indistinguishable from zero-shot CLIP, ending at 37.62 compared with 37.61 for zero-shot CLIP, while naive corruption prompts retain the clearest edge at the highest severity with 38.81.

Figure 3 shows the same story by corruption family. On CIFAR-10-C, naive prompts win three of the four families: 59.31 versus 58.29 on gaussian noise, 81.06 versus 80.88 on motion blur, and 88.53 versus 88.52 on fog. Family residuals are only slightly higher on JPEG compression, 72.31 versus 71.30. The pattern is therefore mildly favorable to naive corruption prompts, not symmetric across families. On CIFAR-100-C, naive prompts lead on all four families, reaching 33.61 on gaussian noise, 57.96 on motion blur, 65.23 on fog, and 46.67 on JPEG compression, while family residuals remain at 32.74, 57.78, 64.73, and 45.89 respectively.

4.5 Ablations and Calibration

Table 3 isolates the components of the final recipe. Two findings matter most. First, replacing the family residual bank with a generic low-quality residual changes performance by only hundredths of a point on both datasets. Second, random family reassignment is numerically identical to the intended family assignment within seed variation. Together, these results argue against a deployable family-specific effect.

Uniform family weighting also matches the full method closely, which suggests that even the estimated family posterior contributes little once the residual vectors are this weak. The only ablation

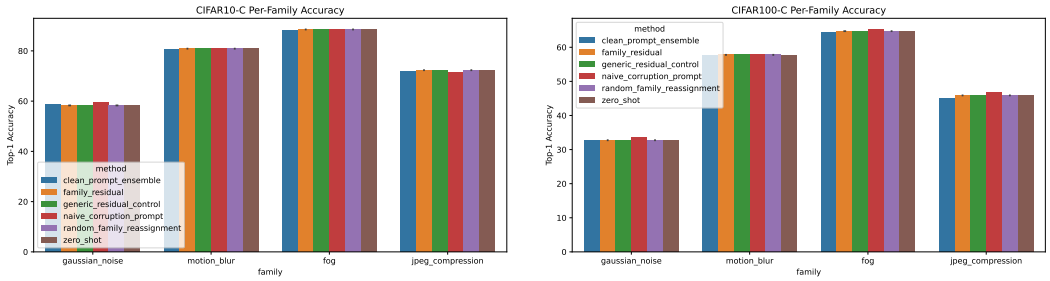

Figure 3: Per-family corruption accuracy for the same six plotted methods. Family residuals do not exhibit a family-specific advantage over zero-shot CLIP, the generic residual control, or random family reassignment; naive corruption prompts are strongest on CIFAR-100-C across all four families and are competitive on CIFAR-10-C as well.

Table 3: Ablation results. Values for methods with seed-dependent selection or randomization are averaged over seeds 7, 17, and 27. The family residual bank is not meaningfully better than generic, random, or uniform-weight controls.

Method	CIFAR-10-C mean	CIFAR-100-C mean
Family residual	75.00	50.28
Uniform family weighting	75.00	50.28
Generic residual control	75.01	50.27
Random family reassignment	75.00	50.28
Naive corruption prompt	**75.05**	**50.87**

with a clear accuracy difference is the naive corruption-prompt baseline, and that difference favors the simpler baseline rather than the proposed family residuals.

ECE follows the same pattern. On CIFAR-10-C, the family-residual method yields mean corruption ECE 0.6417, compared with 0.6418 for zero-shot CLIP and 0.6420 for the generic residual control. On CIFAR-100-C, the corresponding values are 0.4917, 0.4917, and 0.4916. Naive corruption prompts improve accuracy but not calibration, increasing corruption ECE to 0.6429 on CIFAR-10-C and 0.4977 on CIFAR-100-C. Higher-precision ECE values are reported in Appendix A.5.

5 Discussion and Limitations

The empirical picture is consistent. Prompt-side corruption language can induce a weak family alignment signal in embedding space, but that signal is not strong enough to matter once class prediction is the target metric. In downstream accuracy, family residuals collapse to the same behavior as a generic low-quality residual and even to random family reassignment. The simplest explanation is that the text encoder does capture a broad degradation direction, but the family-specific component is too small or too entangled with class semantics to support useful calibration.

The negative result is still informative. It sets a baseline for what can be achieved with a tiny frozen residual bank before moving to heavier adaptation methods. It also warns against over-interpreting semantically pleasing prompt constructions: interpretable latent variables are only valuable if they survive contact with controlled baselines.

Our study has several limitations. It uses one frozen backbone, one OpenCLIP checkpoint, four corruption families, and CIFAR-scale benchmarks only. Heavier comparators such as AutoCLIP and BATCLIP are discussed but not reproduced here; AutoCLIP was explicitly skipped after the pilot failed to show a clearly positive signal. The most important limitation is protocol-related: the executed evaluation selects λ per seed rather than enforcing one globally frozen value, so reviewers should not read these results as a clean confirmation of the original one-configuration plan. If anything, that deviation is mildly favorable to the family-residual method, which makes the downstream negative result more credible but the protocol claim weaker. The method is intentionally class-agnostic on the residual side, which may be too restrictive if family effects interact strongly with object

semantics. Finally, all conclusions are limited to prompt-side frozen calibration; they do not rule out family-aware gains from feature adaptation, prompt optimization, or richer text generation.

6 Conclusion

We studied whether corruption-family text residuals provide useful family-specific information for frozen zero-shot CLIP. A pilot gate showed modest representational evidence, with 34.87 family-posterior accuracy and a 0.0806 alignment margin, but the full benchmark did not support the downstream hypothesis. Family residuals reached 74.9985 mean corruption accuracy on CIFAR-10-C and 50.2817 on CIFAR-100-C, which is effectively the same as zero-shot CLIP, generic residuals, and random family reassignment, and below naive corruption prompts on both datasets.

The main conclusion is narrow but clear: in this frozen prompt-only setting, corruption-language residuals behave more like a generic degradation direction than a useful family-specific calibration signal. Because the executed study reselected λ per seed, this conclusion should be read as evidence from a controlled executed protocol rather than as a strict global-freeze confirmation. Future work should test whether stronger family modeling requires class-conditional text generation, adaptive prompt optimization, or joint feature-space adaptation rather than fixed shared residual banks.

A Reproducibility Appendix

A.1 Selected Configurations

Table 4 lists the executed hyperparameters. The global proxy recipe selected `base`, $\beta = 2.0$, $\alpha = 0.05$, and $\lambda = 0.5$, but the final seeded evaluation retained the seed-local λ selected on the proxy split.

Table 4: Executed hyperparameters copied from `exp/pilot_gate/results.json` and `exp/final_evaluation/config.json`.

Setting	Variant	β	α	λ
Global proxy mean	base	2.0	0.05	0.5
Seed 7	base	2.0	0.05	0.5
Seed 17	base	2.0	0.05	1.0
Seed 27	base	2.0	0.05	0.5

A.2 Prompt Bank

The exact prompt bank comes from `exp/shared/methods.py`.

Clean ensemble. `a photo of a {class}`; `an image of a {class}`; `a close-up photo of a {class}`; `a centered photo of a {class}`; `a blurry photo of a {class}`.

Family posterior prompts (executed `base` variant). `an image with gaussian noise corruption`; `an image with motion blur corruption`; `an image with fog corruption`; `an image with jpeg compression artifacts`.

Residual prompts (executed `base` variant). `a photo of a {class} with gaussian noise`; `a photo of a {class} with motion blur`; `a photo of a {class} with fog`; `a photo of a {class} with jpeg compression artifacts`.

Generic residual prompt. `a low-quality photo of a {class}` relative to `a clear photo of a {class}`.

Table 5: Pilot-gate values reported as rounded copies of the experiment artifact to six decimals.

Seed	λ	Family post. acc.	Main align.	Random align.	Pilot acc.	Generic acc.	Random acc.	Clean zero-shot / method
7	0.5	34.150001	0.080568	0.035735	59.249999	59.450000	59.249999	93.599999 / 93.599999
17	1.0	35.200000	0.080104	0.033607	59.449999	59.500000	59.449999	92.000002 / 92.000002
27	0.5	35.249999	0.081265	0.014835	60.150000	60.150000	60.150000	93.000001 / 93.000001

Table 6: Rounded CIFAR-10 and CIFAR-10-C values for the seed-dependent methods, copied from the artifacts to six decimals. Corruption ECE is the mean over the 20 family-severity conditions for that run.

Method / seed	Clean acc.	Clean ECE	C10-C mean acc.	C10-C mean ECE	Runtime / 10k
Family residual, seed 7	93.680000	0.825028	75.001000	0.641784	0.006893
Family residual, seed 17	93.680000	0.824980	74.993501	0.641676	0.007025
Family residual, seed 27	93.680000	0.825028	75.001000	0.641784	0.007149
Random reassignment, seed 7	93.680000	0.825028	75.002000	0.641794	0.007202
Random reassignment, seed 17	93.680000	0.824980	74.993501	0.641675	0.007255
Random reassignment, seed 27	93.680000	0.825028	75.001000	0.641784	0.007296

A.3 Rounded Pilot Per-Seed Results

A.4 Rounded Final Evaluation Tables

A.5 ECE Summary

A.6 Runtime and System Details

The recorded environment is Linux 6.8, Python 3.10.12, one NVIDIA RTX A6000, batch size 512, and four data-loader workers. The frozen backbone is OpenCLIP ViT-B/32 with `laion2b_s34b_b79k`. The experiment ledger records peak reserved GPU memory of 0.799 GB for the evaluation jobs.

Table 7: Rounded CIFAR-100 and CIFAR-100-C values for the seed-dependent methods, copied from the artifacts to six decimals. Corruption ECE is the mean over the 20 family-severity conditions for that run.

Method / seed	Clean acc.	Clean ECE	C100-C mean acc.	C100-C mean ECE	Runtime / 10k
Family residual, seed 7	75.880003	0.747311	50.279500	0.491698	0.019387
Family residual, seed 17	75.900000	0.747506	50.286000	0.491759	0.019095
Family residual, seed 27	75.880003	0.747311	50.279500	0.491698	0.018643
Random reassignment, seed 7	75.880003	0.747311	50.278500	0.491683	0.018847
Random reassignment, seed 17	75.900000	0.747506	50.285000	0.491749	0.019355
Random reassignment, seed 27	75.880003	0.747311	50.278500	0.491688	0.018863

Table 8: ECE summary. Deterministic methods are single runs. Seed-dependent methods report mean $\pm$ standard deviation over seeds 7, 17, and 27.

Method	CIFAR-10 clean ECE	CIFAR-10-C mean ECE	CIFAR-100 clean ECE	CIFAR-100-C mean ECE
Zero-shot CLIP	0.8249	0.6418	0.7470	0.4917
Clean prompt ensemble	0.8226	0.6396	0.7443	0.4891
Naive corruption prompt	0.8248	0.6429	0.7411	0.4977
Generic residual control	0.8250	0.6420	0.7473	0.4916
Family residual	0.8250 ± 0.0000	0.6417 ± 0.0001	0.7474 ± 0.0001	0.4917 ± 0.0000
Random family reassignment	0.8250 ± 0.0000	0.6418 ± 0.0001	0.7474 ± 0.0001	0.4917 ± 0.0000

Table 9: Runtime per 10k images from the experiment outputs, reported as rounded artifact copies. Seed-dependent rows report mean $\pm$ standard deviation over seeds 7, 17, and 27.

Method	CIFAR-10	CIFAR-100
Zero-shot CLIP	0.005174	0.008249
Clean prompt ensemble	0.006019	0.008184
Naive corruption prompt	0.005379	0.014902
Generic residual control	0.005670	0.012893
Family residual	0.007022 ± 0.000128	0.019042 ± 0.000375
Random family reassignment	0.007251 ± 0.000047	0.019022 ± 0.000288