# Reliable Test-time Adaptation via Evidential Uncertainty Modeling in Vision–Language Models

Supplementary Material

## A. Discussion for Opinion Aggregation

In this section, we first introduce two opinion aggregation strategies: the average-based fusion strategy and the DST-based fusion strategy. Then, we use the average-based joint opinion as an illustrative example to provide a deeper analysis of the behavior and effectiveness of the proposed Reliable Evidential Entropy (REE) loss.

**Definition 1.** *Given two opinions $\mathcal{O}^1$ and $\mathcal{O}^2$, the average-based joint opinion $\tilde{\mathcal{O}} = \{\tilde{\boldsymbol{b}}, \tilde{u}\}$ is defined as:*

$$\tilde{b}_k = \frac{u^2}{u^1+u^2} b_k^1 + \frac{u^1}{u^1+u^2} b_k^2, \tilde{u} = \frac{2u^1u^2}{u^1+u^2}. \quad (1)$$

**Proposition 1.** *The average-based joint opinion $\tilde{\mathcal{O}}$ can be obtained by averaging the evidence vector across all selected views:*

$$\tilde{\boldsymbol{e}} = \frac{1}{|\mathcal{V}(\boldsymbol{x})|} \sum_{v \in \mathcal{V}(\boldsymbol{x})} \boldsymbol{e}^v. \quad (2)$$

*Proof.* Let $\mathcal{O}^1 = \{\boldsymbol{b}^1, u^1\}$ and $\mathcal{O}^2 = \{\boldsymbol{b}^2, u^2\}$ be the multinomial opinion. When considering the joint opinion $\tilde{\mathcal{O}} = \mathcal{O}^1 \bigoplus \mathcal{O}^2$, the belief mass vector $\tilde{\boldsymbol{b}}$ and uncertainty measure $\tilde{u}$ are given by:

$$\tilde{b}_k = \frac{u^2}{u^1+u^2} b_k^1 + \frac{u^1}{u^1+u^2} b_k^2, \quad \tilde{u} = \frac{2u^1u^2}{u^1+u^2}. \quad (3)$$

Based on Eq. (6) in main paper and Eq. (3), the evidence $\boldsymbol{e}$ for the joint opinion is updated as:

$$\begin{aligned} \tilde{e}_k &= \tilde{b}_k \tilde{S} = \frac{\tilde{b}_k C}{\tilde{u}} && (4)\\ &= \frac{b_k^1 u^2 + b_k^2 u^1}{u^1+u^2} \frac{C(u^1+u^2)}{2u^1u^2} && (5)\\ &= \frac{C}{2}\left(\frac{\frac{K}{S^2}\frac{e_k^2}{S^1}}{\frac{K}{S^1}\frac{K}{S^2}} + \frac{\frac{K}{S^1}\frac{e_k^1}{S^2}}{\frac{K}{S^1}\frac{K}{S^2}}\right) && (6)\\ &= \frac{e_k^1 + e_k^2}{2}. && (7) \end{aligned}$$

Therefore, in order to calculate the joint opinion, we can easily compute the average of the evidence parameter of the Dirichlet distribution. □

**Definition 2.** *Given two opinions $\mathcal{O}^1$ and $\mathcal{O}^2$, the DST-based joint opinion $\tilde{\mathcal{O}} = \{\tilde{\boldsymbol{b}}, \tilde{u}\}$ is defined as:*

$$\tilde{b}_k = \frac{1}{1-F}(b_k^1 b_k^2 + u^1 b_k^2 + u^2 b_k^1), \ \tilde{u} = \frac{1}{1-F}(u^1 u^2), \quad (8)$$

*where $F = \sum_{i \neq j} b_i^1 b_j^2$ quantifies the conflict between the two opinions.*

Then, we take average-based fusion strategy as an example, give in-depth analysis about the effect of the proposed Reliable Evidential Entropy loss.

**Proposition 2.** *Given two opinion $\mathcal{O}^1$, $\mathcal{O}^2$, and the average-based joint opinion $\tilde{\mathcal{O}}$. The objective of our propose REE loss can be interpreted as minimizing the entropy of the joint opinion:*

$$\mathcal{L}_{\mathrm{REE}}(\tilde{\boldsymbol{\alpha}}) = \mathbb{E}_{D(\boldsymbol{p}|\boldsymbol{\alpha})}\left[H(\tilde{\boldsymbol{p}})\right]. \quad (9)$$

*Then, we have*

$$\begin{aligned} \mathbb{E}_{D(\boldsymbol{p}|\boldsymbol{\alpha})}\left[H(\tilde{\boldsymbol{p}})\right] = &\frac{u^2}{u^1+u^2}\mathbb{E}_{D(\boldsymbol{p}|\boldsymbol{\alpha})}\left[H(\boldsymbol{p}^1) + D_{\mathrm{KL}}(\boldsymbol{p}^1||\tilde{\boldsymbol{p}})\right] \\ &+ \frac{u^1}{u^1+u^2}\mathbb{E}_{D(\boldsymbol{p}|\boldsymbol{\alpha})}\left[H(\boldsymbol{p}^2) + D_{\mathrm{KL}}(\boldsymbol{p}^2||\tilde{\boldsymbol{p}})\right]. \end{aligned} \quad (10)$$

*Proof.* Here, we give the proof of Proposition 2. First, given two opinion $\mathcal{O}^1 = \{\boldsymbol{b}^1, u^1\}$ and $\mathcal{O}^2 = \{\boldsymbol{b}^2, u^2\}$, the average-based joint opinion $\tilde{\mathcal{O}} = \{\tilde{\boldsymbol{b}}, \tilde{u}\}$ can be obtained by:

$$\tilde{b}_k = \frac{u^2}{u^1+u^2} b_k^1 + \frac{u^1}{u^1+u^2} b_k^2, \tilde{u} = \frac{2u^1u^2}{u^1+u^2} \quad (11)$$

Then, we calculate the expectation prediction $\tilde{\boldsymbol{p}}$ for the joint opinion by:

$$\tilde{p}_k = \tilde{b}_k + \frac{1}{C}\tilde{u} = \frac{u^2}{u^1+u^2} b_k^1 + \frac{u^1}{u^1+u^2} b_k^2 + \frac{2u^1u^2}{C(u^1+u^2)}. \quad (12)$$

We also have:

$$p_k^1 = b_k^1 + \frac{1}{C}u^1, \quad (13)$$

$$p_k^2 = b_k^2 + \frac{1}{C}u^2. \quad (14)$$

Combining with Eq. (12), we have:

$$\tilde{p}_k = \frac{u^2}{u^1+u^2} p_k^1 + \frac{u^2}{u^1+u^2} p_k^2. \quad (15)$$

The REE loss is minimizing $H(\tilde{\boldsymbol{p}})$, which can be further derive:

$$\begin{aligned} H(\tilde{\boldsymbol{p}}) = &H\left(\frac{u^2}{u^1+u^2}\boldsymbol{p}^1 + \frac{u^2}{u^1+u^2}\boldsymbol{p}^2\right) \\ = &\frac{u^2}{u^1+u^2}\left[H(\boldsymbol{p}^1) + D_{\mathrm{KL}}(\boldsymbol{p}^1||\tilde{\boldsymbol{q}})\right] + \\ &\frac{u^1}{u^1+u^2}\left[H(\boldsymbol{p}^2) + D_{\mathrm{KL}}(\boldsymbol{p}^2||\tilde{\boldsymbol{q}})\right]. \end{aligned} \quad (16)$$

The Eq. (10) can be obtained by taking the expectation of both sides in Eq. (16). □

**Remark 2.** *Proposition 2 provides an informative decomposition of the REE objective. It shows that minimizing the expected entropy of the joint opinion $\tilde{\boldsymbol{p}}$ is equivalent to minimizing two components: (1) the uncertainty-weighted entropy of the individual opinions $\boldsymbol{p}^1$ and $\boldsymbol{p}^2$, and (2) the uncertainty-weighted KL divergence between each opinion and the aggregated opinion. This decomposition leads to two key insights. First, the REE loss implicitly favors sharper, more decisive individual opinions, since reducing the entropy of $\boldsymbol{p}^v$ directly decreases the objective. Second, the KL divergence terms encourage the individual opinions to remain consistent with joint opinion, promoting agreement across augmented views. Meanwhile, each opinion's contribution is scaled by the other opinion's uncertainty.*

*Moreover, each opinion's contribution is weighted by the uncertainty of the other opinion. Interpreting larger $u^v$ as greater epistemic uncertainty, the loss exerts stronger pressure on the more reliable opinion when another is uncertain. Concretely, if $\mathcal{O}^2$ is noisy (large $u^2$), the REE loss places greater emphasis on reducing $H(\boldsymbol{p}^1)$ and $D_{\mathrm{KL}}(\boldsymbol{p}^1||\tilde{\boldsymbol{p}})$, effectively relying more on the stable view. In extreme cases, the weighting shifts almost all optimization pressure to the most certain opinion, allowing the mechanism to adaptively discount unreliable sources.*

*Overall, the REE objective jointly promotes confident (low-entropy, low-uncertainty) predictions and coherent alignment among opinions, enabling reliable and stable test-time adaptation.*

## B. Derivation For $\mathcal{L}_{\mathrm{REE}}$

In this section, we provide the detailed derivation of our proposed Reliable Evidential Entropy (REE) loss. Specifically, for augmented view $v$, the REE loss is defined as the expected Shannon entropy of the categorical distribution under the Dirichlet prior:

$$\mathcal{L}_{\mathrm{REE}}(\boldsymbol{\alpha}^v) = \mathbb{E}_{D(\boldsymbol{p}|\boldsymbol{\alpha})}\left[-\sum_{c=1}^{C} p_c^v \log(p_c^v)\right] \tag{17}$$

$$= \sum_{c=1}^{C} \frac{\alpha_c^v}{S}[\psi(S+1) - \psi(\alpha_c^v + 1)], \tag{18}$$

where $S = \sum_{c=1}^{C} \alpha_c^v$, $\psi(\cdot)$ denotes to the digamma function, and $D(\boldsymbol{p}|\boldsymbol{\alpha})$ the Dirichlet distribution parameterized by $\boldsymbol{\alpha}$.

*Proof.* Our Reliable Evidential Entropy (REE) loss is derived by taking the expected Shannon entropy under the Dirichlet prior (as defined in Eq. (5) of the main paper). The expected entropy is given by:

$$\mathcal{L}_{\mathrm{REE}} = \int \Big[-\sum_{c=1}^{C} p_c \log(p_c)\Big] \frac{1}{B(\boldsymbol{\alpha})} \prod_{j=1}^{C} p_j^{\alpha_j - 1} d\boldsymbol{p} \tag{19}$$

$$= \mathbb{E}_{\boldsymbol{p}\sim D(\boldsymbol{p}|\boldsymbol{\alpha})}\left[-\sum_{c=1}^{C} p_c^v \log(p_c^v)\right] \tag{20}$$

$$= -\sum_{c=1}^{C} \mathbb{E}_{\boldsymbol{p}\sim D(\boldsymbol{p}|\boldsymbol{\alpha})}\left[p_c \log(p_c)\right]. \tag{21}$$

To simplify Eq. (21), we first calculate $\mathbb{E}_{D(\boldsymbol{p}|\boldsymbol{\alpha})}\left[p_c \log(p_c)\right]$. Besides, we define:

$$K(\boldsymbol{\alpha}) = \frac{\Gamma(S)}{\Gamma(\alpha_1)\Gamma(\alpha_2)\cdots\Gamma(\alpha_c)\cdots\Gamma(\alpha_C)}, \tag{22}$$

$$K(\boldsymbol{\alpha}') = \frac{\Gamma(S+1)}{\Gamma(\alpha_1)\Gamma(\alpha_2)\cdots\Gamma(\alpha_c+1)\cdots\Gamma(\alpha_C)}, \tag{23}$$

where $\boldsymbol{\alpha}' = (\alpha_1, \alpha_2, \cdots, \alpha_c + 1, \cdots, \alpha_C)$, and we have $K(\boldsymbol{\alpha}) = \frac{\alpha_c}{S} K(\boldsymbol{\alpha}')$. Consequently, for $c \in \{1, 2, \cdots, C\}$, we can rewrite:

$$\frac{\partial D(\boldsymbol{p}|\boldsymbol{\alpha}')}{\partial \alpha_c'} = \partial(K(\boldsymbol{\alpha}') \prod_{j=1}^{C} p_j^{\alpha_j' - 1})/\partial \alpha_c' \tag{24}$$

$$= \frac{\partial K(\boldsymbol{\alpha}')}{\partial \alpha_c'} \prod_{j=1}^{C} p_j^{\alpha_j' - 1} + K(\boldsymbol{\alpha}') \frac{\partial \prod_{j=1}^{C} p_j^{\alpha_j' - 1}}{\partial \alpha_c'} \tag{25}$$

$$= \frac{\partial K(\boldsymbol{\alpha}')}{\partial \alpha_c'} \prod_{j=1}^{C} p_j^{\alpha_j' - 1} + \frac{S}{\alpha_c} p_c \log(p_c) \cdot D(\boldsymbol{p}|\boldsymbol{\alpha}). \tag{26}$$

By performing integral to both sides, we have:

$$\int \frac{\partial D(\boldsymbol{p}|\boldsymbol{\alpha}')}{\partial \alpha_c'} d\boldsymbol{p} = \int \frac{\partial K(\boldsymbol{\alpha}')}{\partial \alpha_c'} \prod_{j=1}^{C} p_j^{\alpha_j' - 1} d\boldsymbol{p} + \frac{S}{\alpha_c} \int p_c \log(p_c) \cdot D(\boldsymbol{p}|\boldsymbol{\alpha}) d\boldsymbol{p}. \tag{27}$$

The left side of the equation can be simplified through the following steps:

$$\text{left side} = \int \frac{\partial D(\boldsymbol{p}|\boldsymbol{\alpha}')}{\partial \alpha_c'} d\boldsymbol{p} = \frac{\partial \mathbf{1}}{\partial \alpha_c'} = 0, \tag{28}$$

while the right side can be further simplified as:

$$\text{right side} = \int \frac{\partial K(\boldsymbol{\alpha}')}{\partial \alpha_c'} \prod_{j=1}^{C} p_j^{\alpha_j' - 1} d\boldsymbol{p} + \frac{S}{\alpha_c} \mathbb{E}_{D(\boldsymbol{p}|\boldsymbol{\alpha})}[p_c \log(p_c)]. \tag{29}$$

| Dataset | Domain / Type | Classes | Size |
|---|---|---|---|
| ImageNet-V2 | Natural images (resampled) | 1,000 | 10k |
| ImageNet-A | Natural adversarial images | 200 | 7.5k |
| ImageNet-R | Artistic renderings | 200 | 30k |
| ImageNet-Sketch | Hand-drawn sketches | 1,000 | 50k |
| Flower102 | Fine-grained flowers | 102 | 8,189 |
| DTD | Textures | 47 | 5,640 |
| Pets | Dog & cat breeds | 37 | 7,349 |
| Cars | Fine-grained cars | 196 | 16,185 |
| UCF101 | Human actions (frames) | 101 | 9,537 |
| Caltech101 | Generic objects | 101 | 9,146 |
| Food101 | Food dishes | 101 | 101k |
| SUN397 | Scene understanding | 397 | 108k |
| Aircraft | Aircraft variants | 100 | 10k |
| EuroSAT | Satellite imagery | 10 | 27k |

Table 1. The dataset statistics used for natural distribution shift and cross-domain generalization evaluation.

For the first term, we have:

$$\int \frac{\partial K(\boldsymbol{\alpha}^{'})}{\partial \alpha_c^{'}} \prod_{j=1}^{C} p_j^{\alpha_j^{'}-1} d\boldsymbol{p} \tag{30}$$

$$= \int \frac{\partial K(\boldsymbol{\alpha}^{'})}{\partial \alpha_c^{'}} \frac{1}{K(\boldsymbol{\alpha}^{'})} D(\boldsymbol{p}|\boldsymbol{\alpha}^{'}) d\boldsymbol{p}, \tag{31}$$

$$= \frac{\partial \log K(\boldsymbol{\alpha}^{'})}{\partial \alpha_c^{'}}. \tag{32}$$

Consequently, combining Eq. (27), Eq. (28), Eq. (29), and Eq. (32), we can obtain:

$$\mathrm{E}_{D(\boldsymbol{p}|\boldsymbol{\alpha})}[p_c \log(p_c)] \tag{33}$$

$$= -\frac{\alpha_c}{S} \frac{\partial \log K(\boldsymbol{\alpha}^{'})}{\partial \alpha_c^{'}} \tag{34}$$

$$= -\frac{\alpha_c}{S} \frac{\partial \{\log \Gamma(S+1) - \sum_{j=1}^{C} \log \Gamma(\alpha_j^{'})\}}{\partial \alpha_c^{'}} \tag{35}$$

$$= \frac{\alpha_c}{S} \left[ \frac{\partial \log \Gamma(\alpha_c^{'})}{\partial \alpha_c^{'}} - \frac{\partial \log \Gamma(S+1)}{\partial \alpha_c^{'}} \right] \tag{36}$$

$$= \frac{\alpha_c}{S} [\psi(\alpha_c^{'}) - \psi(S+1)] \tag{37}$$

$$= \frac{\alpha_c}{S} [\psi(\alpha_c + 1) - \psi(S+1)], \tag{38}$$

where $\psi(\cdot)$ denotes to the digamma function. Finally, we give the derivation of our REE loss:

$$\mathcal{L}_{\mathrm{REE}}(\boldsymbol{\alpha}^v) = \mathbb{E}_{D(\boldsymbol{p}|\boldsymbol{\alpha})} \left[ -\sum_{c=1}^{C} p_c^v \log(p_c^v) \right] \tag{39}$$

$$= -\sum_{c=1}^{C} \mathbb{E}_{D(\boldsymbol{p}|\boldsymbol{\alpha})} \left[ p_c^v \log(p_c^v) \right] \tag{40}$$

$$= \sum_{c=1}^{C} \frac{\alpha_c^v}{S} [\psi(S+1) - \psi(\alpha_c^v + 1)]. \tag{41}$$

This concludes the proof. □

## C. Experiment Details

### C.1. Benchmarks

We provides additional details on the datasets used to evaluate the model's robustness under natural distribution shifts and its cross-dataset generalization ability. To facilitate reproducibility and offer a clearer understanding of each benchmark's characteristics, we summarize dataset scales, class structures, and domain properties in the accompanying Table 1.

**Natural Distribution Shift Benchmarks.** To comprehensively evaluate the model's robustness under real-world distribution variations, we use four ImageNet-derived out-of-distribution (OOD) benchmarks: ImageNet-V2, ImageNet-A, ImageNet-R, and ImageNet-Sketch. Although they share the same label space as ImageNet, these datasets intentionally introduce different types of natural shifts—including resampling-based variation, adversarial-like difficulty, artistic stylization, and drastic appearance

changes—to examine robustness from multiple complementary perspectives.

Specifically, **ImageNet-V2** provides a mild but realistic shift by re-collecting a subset of the original validation set with an independently sampled pipeline, allowing the evaluation under the subtle distribution mismatch. In contrast, **ImageNet-A** contains naturally occurring yet highly adversarial samples that frequently mislead the modern classifiers, thereby testing resilience to challenging real-world edge cases. Extending beyond difficulty shifts, **ImageNet-R** introduces a wide range of stylized renditions—such as cartoons, paintings, graffiti, and sculptures—capturing substantial texture and style variations. Finally, **ImageNet-Sketch** presents the hand-drawn sketch representations, imposing an extreme appearance shift that primarily preserves the object shape cues while removing the most texture information.

Together, these four benchmarks offer a unified and complementary suite for evaluating robustness across style, texture, acquisition, and difficulty variations, thereby providing a holistic assessment of model performance under natural distribution shifts.

**Cross-Dataset Generalization Benchmarks.** To more comprehensively assess the model's adaptability to unseen domains and novel category distributions, we further evaluate it across ten heterogeneous image classification datasets drawn from a wide spectrum of visual domains. These datasets are intentionally selected to feature completely disjoint label spaces from ImageNet, ensuring a clean and unbiased setting for evaluating zero-shot cross-domain generalization. By covering distinct visual characteristics, acquisition conditions, and semantic categories, they jointly provide a broad and challenging testbed for examining real-world transferability.

Specifically, the benchmark suite includes the datasets focusing on *fine-grained object categories* such as Flower102, Pets, Cars, and Aircraft, each requiring attention to subtle intra-class variations and high inter-class similarity. To capture the *texture-centric domains*, DTD provides rich low-level pattern diversity that significantly differs from natural object-centric datasets. UCF101 introduces *human action frames*, testing the ability to generalize from static training data to the motion-related semantics. Caltech101 and SUN397 further broaden the scope by covering *generic objects* and *diverse indoor/outdoor scene categories*, respectively, emphasizing shape, spatial layout, and contextual relationships. In addition, Food101 evaluates recognition under substantial *appearance variability due to cooking styles and presentation*, while EuroSAT contributes *multi-spectral satellite imagery* distinct from natural RGB photography.

Collectively, these datasets span multiple levels of visual abstraction—from textures and objects to scenes and remote sensing—and encompass domains with varying granularity, structural complexity, and environmental conditions. This extensive diversity enables a thorough examination of the model's capacity to transfer beyond the ImageNet taxonomy and generalize robustly across fundamentally different visual distributions.

### C.2. Implementation

All experiments are conducted using the pre-trained CLIP model [37], which consists of an image encoder and a text encoder. The image encoder is instantiated as either a ResNet-50 (RN50) or a ViT-B/16 backbone, while the text encoder is a Transformer-based architecture.

For train-time adaptation methods, models are trained on the ImageNet training set using 16 shots per class and evaluated on the remaining datasets following [55]. In contrast, test-time adaptation methods (i.e., TPT, DiffTPT, TDA, and our RTA) do not access the ImageNet training set. Instead, they adapt online using a stream of unlabeled test samples drawn from the target domain. Test-time adaptation is performed in a single-image setting with a batch size of 1. For each test sample, we generate 64 augmented views following the TPT protocol [41].

Hyperparameters are tuned via validation on each benchmark dataset and kept fixed when evaluating on unseen datasets [23]. We set the shot capacity of the uncertainty-aware cache model to $K = 3$, resulting in the size of the uncertainty-aware cache model is $CK$. The trade-off parameter is set to $\lambda = 0.7$ for all benchmarks. The cutoff percentile for reliable view selection is set to $\rho = 0.9$ for all benchmarks. For the $\alpha$ and $\beta$ in Eq. (14), we follow the configuration of Karmanov et al. and adopt benchmark-specific values. We employ AugMix [17] as a stronger data augmentation strategy to generate diverse views. We optimize the learnable prompt for only one step per test sample during inference. All experiments are executed on a single NVIDIA RTX 4090 GPU.

## D. More Experimental Result and Analysis

### D.1. Combination with Train-time Adaptation

Given the strong performance of train-time adaptation baselines shown in Table 1 of the main paper, we further investigate the effectiveness of combining our RTA framework with a representative train-time adaptation method. For the implementation of RTA + CoOp, we first apply CoOp to fine-tune the learnable prompts on ImageNet using 16-shot training samples per class. We then apply our RTA framework on the test set to perform additional test-time prompt refinement. As shown in Table 2, our RTA framework consistently improves performance when integrated with the CoOp baseline. Specifically, by employ-

| Method | ImageNet | Pets | Average |
|---|---|---|---|
| CLIP-ViT-B/16 | 68.34 | 86.92 | 77.63 |
| CoOp [55] | 71.15 | 89.14 | 80.15 |
| CoCoOp [54] | 71.02 | 90.46 | 80.74 |
| TPT [41] | 68.69 | 87.79 | 78.24 |
| TPT + CoOp [41] | 72.98 | 90.29 | 81.64 |
| RTA-AVG | 71.32 | 88.25 | 79.79 |
| RTA-DS | 71.88 | 89.83 | 80.86 |
| RTA-AVG + CoOp | 73.01 | 90.44 | 81.73 |
| RTA-DS + CoOp | **73.29** | **90.82** | **82.06** |

Table 2. Ablation study for the combination of RTA and train-time adaptation baselines with CLIP-ViT-B/16 backbone on ImageNet and Pets, respectively.

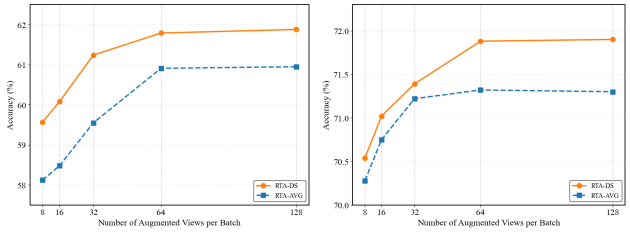
ing CoOp, RTA-AVG achieves improvement of **1.69%** and RTA-DS achieves improvement of **1.41%** on ImageNet, respectively. Meanwhile, both two variants achieve better performance compare to the TPT + CoOp setting. Notably, RTA-DS + CoOp reaches the best average of **82.06**, demonstrating that combining train-time and test-time adaptation strategies offers a promising direction for further enhancing CLIP's downstream performance.

### D.2. Impact of the Number of Augmented Views $V$

We analyze how the number of augmented views $V$ influences the adaptation quality of our method, as it effectively expands the set of test-time samples available for optimization. Figure 1 reports the accuracy curves of RTA-AVG and RTA-DS on ImageNet using CLIP-RN50 and CLIP-ViT-B/16 backbones. Across both architectures, we observe a consistent trend: accuracy improves steadily as $V$ increases, indicating that additional augmentations provide richer and more diverse evidential cues for uncertainty modeling. Performance continues to rise up to approximately $V = 64$, after which the improvement saturates. When $V$ exceeds this threshold, the additional gains become marginal, suggesting diminishing returns once the uncertainty estimates and aggregated evidence have reached sufficient stability. However, increasing $V$ also introduces higher computational overhead. Considering the trade-off between efficiency and performance, we set $V = 64$ for all benchmark experiments.

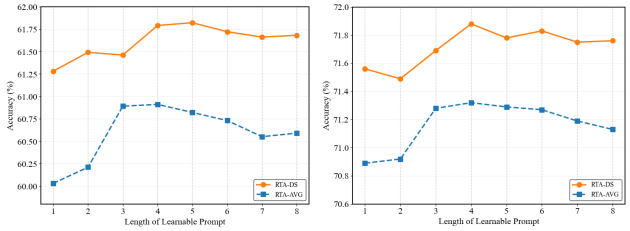
### D.3. Impact of the Length of Learnable Prompt $M$

We further investigate the influence of the learnable prompt length on the adaptation performance of RTA. Figure 2 presents the results for both RTA-AVG and RTA-DS using CLIP-RN50 and CLIP-ViT-B/16 on ImageNet. Across both backbones, we observe that increasing the prompt length generally leads to performance improvements, particularly




(a) CLIP-RN50 (b) CLIP-ViT-B/16

Figure 1. Ablation study on different number of augmented views.




(a) CLIP-RN50 (b) CLIP-ViT-B/16

Figure 2. Ablation study on length of learnable prompt.

in the range of 1 to 5 tokens. A longer prompt provides a more expressive parameterization, enabling the model to better align visual features with textual descriptions under distribution shifts. This trend is especially pronounced for RTA-DS, whose uncertainty-guided selection benefits from richer prompt semantics and thus exhibits a more stable upward trajectory. However, when the prompt length continues to grow beyond 5 tokens, the gains begin to diminish or even slightly fluctuate. This phenomenon is consistent with prior findings in prompt tuning: overly long prompts may introduce redundant or noisy parameters, which can hinder optimization, especially under the constrained test-time setting where updates are performed online and without ground-truth supervision.

### D.4. Comparison of Training and Inference Budgets

We compare the training and inference budgets for various prompting strategies applied to CLIP in Table 3. The main computational cost of our RTA framework comes from the 1-step prompt optimization, which involves backpropagation through CLIP's text encoder. Compared with the TPT baseline, RTA introduces additional overhead from data augmentation, opinion aggregation, and the uncertainty-aware cache model. Importantly, we should point out that RTA operates exclusively at test time; it does not require any training budget, similar to TPT. Moreover, our empirical results show that prompt tuning performed without access to training data can often generalize more effectively to unseen distributions, highlighting an advantage of test-time adaptation approaches, especially for our proposed RTA.

| | CoOp | CoCoOp | TPT | RTA-DS | RTA-AVG |
|---|---|---|---|---|---|
| Inference speed (s/iter) | 0.10 | 0.11 | 0.25 | 0.42 | 0.39 |
| Training samples | 16K | 16K | 0 | 0 | 0 |
| Training iterations | 12.5K | 800K | 0 | 0 | 0 |
| Learnable parameters | 2,048 | 34,816 | 2,048 | 2,048 | 2,048 |

Table 3. Comparison of training and testing budgets of prompting strategies for CLIP on ImageNet with CLIP-RN50 backbone.

**Algorithm 1:** Pseudo code for Reliabel Test-time Adaptation framework

**Input:** Test sample set $\mathcal{D}_{\text{test}} = \{\boldsymbol{x}_i\}_{i=1}^{N}$, augmentation function set $\{\mathcal{A}^v\}_{v=1}^{V}$, pre-trained CLIP model $\mathcal{F} = \{F^I, F^T\}$ ;

**Output:** Test sample prediction $\boldsymbol{p}_{\text{refine}}$ ;

**Initialization:** Learnable prompt $\boldsymbol{P}$ for each class, uncertainty-aware cache model $\mathcal{C}$ ;

**for** $\boldsymbol{x} \in \mathcal{D}_{\text{test}}$ **do**

`# Reliable Test-time Prompt Tuning`

Construct augmented view set $\{\mathcal{A}^v(\boldsymbol{x})\}_{v=1}^{V}$ ;

Calculate the text prompt representation $\boldsymbol{z}_c^T = F^T(\boldsymbol{P}_c)$ ;

**for** $v = 1, 2, ..., V$ **do**

Calculate the image representation $\boldsymbol{z}^v = F^I(\mathcal{A}^v(\boldsymbol{x}))$ ;

Obtain view-level evidence $\boldsymbol{e}^v = \text{softplus}(\cos(\boldsymbol{z}^v, \boldsymbol{z}^T))$ ;

Convert $\boldsymbol{e}^v$ to view-level opinion $\mathcal{O}^v = \{\boldsymbol{b}^v, u^v\}$ ;

Select reliable augmented view subset $\mathcal{V}(\boldsymbol{x}) = \{v | \mathbb{I}(u^v < \tau) = 1, v = 1, 2, \cdots, V\}$ ;

Fuse joint opinion $\tilde{\mathcal{O}} = \bigoplus_{v \in \mathcal{V}(\boldsymbol{x})} \mathcal{O}^v$ ;

Update learnable prompt $\boldsymbol{P}$ using loss $\mathcal{L}_{\text{REE}}$ ;

`# Uncertainty-aware Cache Model Refinement`

**if** $|\mathcal{C}| > 0$ **then**

Set $K = |\mathcal{C}|$ ;

**for** $k = 1, 2, ..., K$ **do**

Calculate similarity score $s_k = \alpha \exp(-\beta(1 - \cos(\boldsymbol{z}^I, \boldsymbol{z}_k^I)))$ ;

Calculate uncertainty-aware adjustment factor $\gamma_k = 1 - \frac{u_k}{\max_{j=1}^{K} u_j}$ ;

Calculate cache prediction $\boldsymbol{p}_{\text{cache}} = \text{softmax}(\sum_{k=1}^{K} \gamma_k s_k \cdot \hat{\boldsymbol{y}}_k)$ ;

Calculate the CLIP-based prediction of original image $\boldsymbol{p} = \boldsymbol{b} + \frac{1}{C} u$ ;

Refine the prediction $\boldsymbol{p}_{\text{refine}} = \lambda \boldsymbol{p} + (1 - \lambda) \boldsymbol{p}_{\text{cache}}$ ;

Obtain pseudo label of test sample $\hat{\boldsymbol{y}} = \arg\max_{c=1}^{C} p_{\text{refine}}^{(c)}$ ;

**if** $u < \max_{\hat{\boldsymbol{y}}_k = \hat{\boldsymbol{y}}} u_k$ *and* $H(\boldsymbol{p}) < \max_{\hat{\boldsymbol{y}}_k = \hat{\boldsymbol{y}}} \boldsymbol{p}_k$ **then**

Update $\mathcal{C}$ with $\{\boldsymbol{z}^I, \hat{\boldsymbol{y}}\}$ ;

## E. Pseudo Code

The pseudo-code for our proposed Reliable Test-time Adaptation framework is summarized in Algorithm 1. RTA contains two key components:(1) Reliable Test-time Prompt Tuning, which converts multiple augmented views into Dirichlet-based evidential opinions, selects reliable views based on uncertainty, and fuses them into a joint opinion. The proposed $\mathcal{L}_{\text{REE}}$ loss is then used to guide consistent and uncertainty-aware prompt adaptation; and (2) Uncertainty-aware Cache Model, which dynamically update the cache by replacing less reliable samples with confident ones, and integrates prior CLIP knowledge to refine test sample prediction for robust and adaptive inference.