Is Less More? Exploring Token Condensation as Training-free Test-time Adaptation

Supplementary Material

This supplementary material provides additional details of TCA, including method descriptions, theoretical analysis, empirical results, and the algorithm. We also discuss TCA's applicability and limitations. To further illustrate the method, we include visual aids for token condensation.

- **Sec. 5.1**: Interpretation of Fig. 2
- **Sec. 5.2:** Details of the Coreset Selection Strategy
- **Sec. 5.3:** Theoretical Analysis
- **Sec. 5.4:** Additional Experiments and Ablation Study
- **Sec. 5.5:** Token Condensation Algorithm
- **Sec. 5.6:** Quantitative Study ($R = 0.7$).
- **Sec. 5.7:** Discussion on TCA's Generalizability
- **Sec. 5.8:** Potential Limitation of TCA

5.1. Interpretation of Fig. 2

In Fig. 2a, we sort the image tokens by their attention to the `<cls>` token and group them into bins. We then sequentially prune each group and measure the change in CLIP's visual-text alignment. The y-axis shows the alignment degradation when a specific token group is removed. A larger drop means the pruned tokens were important for alignment, while a negative drop suggests a slight improvement. It shows that removing attentive tokens (x-axis left) harms alignment, whereas pruning low-attentive tokens has little or even a positive impact. This validates our motivation that token attentiveness is strongly correlated with their semantic importance. **In Fig. 2b**, the y-axis is the average cosine similarity between the stored anchor tokens (*i.e.*, the `<cls>` tokens of low-entropy samples) and the corresponding ground-truth text embeddings. As the reservoir is progressively updated with lower-entropy samples, the average alignment between anchor tokens and text embeddings improves, validating the domain- and class-representativeness of saved domain anchors.

5.2. Details of Coreset Selection

In domain-aware token merging, we first identify the most representative tokens $\hat{\mathbf{V}}^l_{\text{merge}} \in \mathbb{R}^{K \times D_v}$ from $\mathbf{V}^l_{\Phi}$ and assigns the remaining ambiguous tokens to these selected tokens. This strategy is equivalent to solving the K-Center problem [55, 72]. The objective is to select K center tokens such that the maximum distance between any token and its nearest center is minimized. The greedy search for coreset optimization is defined as follows:

$$\mathbf{C}^* = \arg\min_{\mathbf{C} \subseteq \mathbf{V}^l_{\Phi}, |\mathbf{C}|=K} \max_{\mathbf{v}^l_i \in \mathbf{V}^l_{\Phi}} \min_{\mathbf{v}^l_c \in \mathbf{C}} d(\mathbf{v}^l_i, \mathbf{v}^l_c), \tag{10}$$

where $\mathbf{C}^* \in \mathbb{R}^{K \times D_v}$ represents the set of selected center tokens, K is the number of centers, and $d(\cdot, \cdot)$ is the distance metric between token $\mathbf{v}^l_i$ and center token $\mathbf{v}^l_c$. Once the center tokens $\mathbf{C}^*$ are selected, the remaining tokens are assigned to their nearest centers, and the ambiguous tokens are merged as:

$$\hat{\mathbf{V}}^l_{\text{merged}} = \frac{1}{|\mathcal{N}(k)|} \sum\nolimits_{\mathbf{v}^l_i \in \mathcal{N}(k)} \mathbf{v}^l_i, \tag{11}$$

where $\mathcal{N}(k)$ represents the set of tokens assigned to center k. The value of K is kept small, with $K \ll N$, allowing our merging algorithm to operate with linear complexity.

5.3. Theoretical Analysis

The theoretical foundations of CLIP's generalization remain underexplored, with ongoing debates on whether it arises from train-test similarity [43], spurious feature reliance [65], or other factors. While we did not include rigorous proof, we connect our TCA to *PAC-Bayesian generalization theory*. We model token selection as a stochastic hypothesis, where the *posterior* $\mathbb{Q}$ over retained tokens follows a *Gibbs formulation*, favoring subsets that minimize cosine similarity variance with texts:

$$\mathbb{Q}(\hat{\mathbf{V}}) = \frac{1}{Z} \exp\left(-\lambda \mathrm{Var}\left(\cos(\mathbf{V}, \mathbf{t}_c)\right)\right),$$

$$\mathbb{E}_{\text{ood}}[-\cos(\hat{\mathbf{V}}, \mathbf{t}_c)] \leq \mathbb{E}_{\text{id}}[-\cos(\mathbf{V}, \mathbf{t}_c)] + \sqrt{\frac{1}{2}(D_{\text{KL}}(\mathbb{Q} \| \mathbb{P}) + \log \frac{m}{\delta}}.$$

This supports the PAC-Bayes bound, where TCA improves generalization by reducing KL divergence between test-time token selection and CLIP's inaccessible pretraining distribution, which we approximate using DTR. Empirical results in Fig. 2b confirm this, showing that retained tokens act as stable anchors for text alignment.

5.4. Additional Results

Impact of Visual Backbone. Trends similar to ViT-B/16 are observed with the **ViT-L/14** architecture, as shown in Tab. 6. TCA consistently surpasses TDA across multiple datasets, including Aircraft, Caltech101, EuroSAT, Flower102, Pets, and UCF101, while adhering to a limited GFLOPs budget (19.6% GFLOPs reduction). Even with a 48.9% reduction in GFLOPs, TCA continues delivering satisfactory results. This demonstrates the scalability and robustness of our method across different model sizes, reinforcing its effectiveness without additional training.

Table 6. Results on the cross-dataset benchmark with CLIP ViT-L/14. * denotes the averaged GFLOPs across all datasets.

Method	Aircraft	Caltech101	Cars	DTD	EuroSAT	Flower102	Food101	Pets	SUN397	UCF101	**Average**	**GFLOPs**
CLIP	31.59	94.56	78.12	57.03	63.00	79.58	90.92	93.46	69.05	76.13	73.34	81.14
Tent	27.45	94.97	76.93	57.15	66.20	74.83	89.20	93.27	68.73	75.73	72.45	81.14
SAR	26.07	94.52	75.58	56.91	63.77	75.03	89.13	93.05	68.39	75.50	71.80	81.14
TPT	30.06	95.21	76.84	52.30	55.11	76.21	88.56	93.08	67.69	73.78	70.88	143.31
TDA	33.42	95.46	78.72	57.39	66.27	79.94	90.83	93.27	70.74	78.14	74.42	81.14
EViT$_{R=0.9}$	31.23	94.56	76.59	56.38	63.04	79.13	90.08	93.32	68.54	76.40	72.93	65.19
ToME$_{R=0.9}$	28.29	92.54	71.26	56.68	60.30	77.87	89.77	91.28	68.21	72.22	70.84	64.74
ATS$_{R=0.9}$	25.74	93.39	67.69	55.02	52.81	76.78	86.48	91.50	66.26	72.56	68.82	43.62*
EViT$_{R=0.7}$	26.94	92.94	62.55	53.96	52.04	73.24	80.69	90.00	63.70	71.21	66.73	40.78
ToME$_{R=0.7}$	15.60	83.73	38.43	49.82	44.51	59.36	72.65	77.73	58.32	50.99	55.11	40.05
ATS$_{R=0.7}$	6.87	67.87	16.37	40.78	30.12	37.43	34.50	60.94	30.07	33.44	35.84	26.76*
TCA$_{R=0.9}$	33.84	96.39	76.93	56.38	67.74	80.71	90.21	93.54	70.02	78.24	74.40	**65.24**$_{-19.6\%}$
TCA$_{R=0.7}$	29.73	94.81	63.72	53.72	60.69	76.00	81.55	90.02	65.61	73.14	68.90	**41.44**$_{-48.9\%}$

Table 7. Impact of scale factor β.

β	**0.01**	**0.05**	**1**	**3**	**5**
Pets	89.51	**89.53**	89.37	89.42	89.26
Flower102	**73.33**	73.08	70.93	70.56	70.44
EuroSAT	63.64	64.06	69.86	70.26	**70.43**

Impact of Logits Correction Temperature β. Tab. 7 examines how different logits correction temperatures β affect the adaptation results. The intuition is that with a smaller β value, the logits correction will emphasize the tokens in shallower layers (Eq. (9)), while a larger β value will shift the focus to deeper layers. We observe that a smaller value of β is preferred for the Pets dataset as it contains animals as objects, requiring more high-level contextual information for accurate predictions [49]. In contrast, for EuroSAT, the best predictions are obtained with larger β values, suggesting that low-level, local information is crucial. This aligns well with the nature of the dataset, where different types of land can be distinguished by features such as colors and edges. Nevertheless, our method consistently provides significant improvements across all β values, with accuracy gains of up to 20%, highlighting the effectiveness of logits correction using the domain anchor tokens.

Table 8. Impact of correction weight λ.

λ	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Pets	**89.53**	89.32	89.13	88.96	88.96	88.66	88.44
Flower102	72.43	72.76	73.20	73.16	73.16	**73.33**	73.16
EuroSAT	60.15	65.74	68.80	69.51	69.84	70.16	**70.43**

Impact of Correction Weight λ. To investigate how different correction weights λ affect performance, as described in Eq. (9), we conducted experiments across a wide range of λ values, from 2 to 8, as shown in Tab. 8. We observe that Pets exhibits stable results across different λ values, indicating that less aggressive correction is sufficient. In contrast, datasets such as Flower102 and EuroSAT which initially do not perform well on CLIP, benefit from stronger corrections, achieving their best performance with larger correction weights of 7 and 8, respectively. This highlights the effectiveness of our logits correction module.

Table 9. Impact of token merging/pruning ratio.

Merging:Pruning	**0:1**	**1:2**	**2:1**
Pets	89.04	88.99	**89.53**
EuroSAT	69.63	69.98	**70.43**

Impact of Pruning & Merging Ratio. We experiment with different token pruning and merging ratios under the same computational budget, as shown in Tab. 9. Incorporating token diversity through merging consistently enhances performance. Specifically, the 2:1 merging-to-pruning ratio outperforms other configurations, especially those favoring pruning. This is because merging preserves diverse token representations by K coresets that pure pruning might discard. When comparing pruning-only (0:1) with the 1:2 merging-pruning ratio on Pets, pruning-only performs better. This may be because the dataset features images with a single prominent object, meaning that pruning background tokens has minimal impact since essential object information remains intact. In contrast, for the EuroSAT dataset, which comprises diverse satellite imagery, simply pruning tokens leads to the loss of important contextual features necessary for accurate classification.

Impact of Merging Center Number K. We evaluate TCA performance by giving different numbers of merging centers K for Pets, EuroSAT, and Food101 datasets. As shown in Tab. 10, setting $K = 2$ consistently yields the best results. This choice balances preserving important information and reducing redundancy. A smaller K (*i.e.,* $K = 1$) may oversimplify the merging process, leading to the loss of

Table 10. Impact of the merging center number K.

K	1	2	3	4
Pets	89.29	**89.53**	89.29	89.21
EuroSAT	66.25	**70.43**	66.96	67.44
Food101	85.15	**85.31**	85.31	85.38

critical details, especially in diverse datasets like EuroSAT. Conversely, increasing K beyond 2 introduces unnecessary complexity and can over-segment the token space, retaining redundant tokens that contribute little to classification. Therefore, maintaining a very small K (where $K \ll N$) is sufficient and advantageous.

Impact of Benchmark Datasets. We conducted experiments on the OOD benchmark which focuses on evaluating the model's effectiveness on shifted data using label sets previously seen by CLIP. This includes variants of ImageNet [14]: ImageNet-A [24], ImageNet-V2 [51], ImageNet-R [25], and ImageNet-S [62]. A consistent observation can be seen in the out-of-distribution (OOD) benchmark, where TCA demonstrates significant improvements over the CLIP baseline under a constrained GFLOPs budget of $R = 0.95$, as shown in Tab. 11. TCA outperforms traditional test-time adaptation methods while maintaining efficiency. TCA also achieves superior results on ImageNet-R and ImageNet-S, outperforming TPT without augmentation. Additionally, when compared to other training-based approaches, even those with unlimited computational budgets, TCA delivers comparable performance. However, we observe that TCA does not perform as strongly on the OOD benchmark as it does on the CD benchmark even with a higher rate R. This may be due to the conceptual shifts in OOD datasets, as shown in Sec. 5.8, which could present a challenge for training-free adaptation methods.

5.5. Algorithm

Algorithm 1 outlines the process for performing token pruning and merging at layer l in a ViT-based CLIP model. We first obtain the averaged domain anchor tokens $\mathbf{A}_{c^*}^{l-1}$ by the `<cls>` tokens saved in the reservoir $\mathfrak{R}$. Token condensation is then conducted given the domain anchor token. Specifically, we conduct token pruning by relative ranking positions of token i across multiple attention heads. Then, coreset selection is used for token merging. Finally, we concatenate the `<cls>` token $\mathbf{v}_{\text{cls}}^l$ with the retained tokens as the input for the next layer, where the original $N+1$ tokens are shrunk to $(R \cdot N) + 1$, thereby reducing the computational cost.

5.6. Quantitative Study

We visualize the token condensation masks at layer 3, layer 6, and layer 9, and compare them with the original image across multiple datasets, as shown in Fig. 8. As the layers go deeper, we observe that class-irrelevant patches are gradually pruned, as indicated by the black mask. TCA also merges class-ambiguous patches, such as fur in cat images, and ground and sky in aircraft and car images. All similar tokens are merged into a single token using our proposed coreset selection strategy. After token condensation, the sample features retain only discriminative information, thereby bridging the gap between visual and text features, and mitigating the distribution shift between pretrained data and unseen datasets.

Algorithm 1 Token Condensation at the l-Layer in E_v

```
Require:
  Token reservoir $\mathfrak{R}$;
  Visual patches $\mathbf{V}^{l-1}$ at layer $l-1$;
  Pruning threshold $\theta_{\text{prune}}(\alpha \cdot R)$;
  Merging threshold $\theta_{\text{merge}}(R)$
Ensure: Token-efficient visual feature $\hat{\mathbf{V}}^l$
  Domain Anchor Token Selection: Obtain $\mathbf{A}_{c^*}^{l-1}$, using domain anchor tokens in $\mathfrak{R}$ and sample's <cls> token $\mathbf{v}_{\text{cls}}^l$
  Compute cross-head scores $\mathbf{S}_i^{\text{head}}$ for every token $i$
  if $\forall i, S_i^{\text{head}} \leq \theta_{\text{prune}}(\alpha \cdot R)$ then
      Token Pruning: Obtain $\hat{\mathbf{V}}_{\text{prune}}^l$ via Eq. (7)
  end if
  if $\forall i, \theta_{\text{merge}}(R) \leq S_i^{\text{head}} \leq \theta_{\text{prune}}(\alpha \cdot R)$ then
      Token Merging: Obtain $\hat{\mathbf{V}}_{\text{merged}}^l$ via Eq. (11)
  end if
  return $\hat{\mathbf{V}}^l$, which is composed of $\mathbf{v}_{\text{cls}}^l$, $\hat{\mathbf{V}}_{\text{prune}}^l$ (excluding merged tokens), and $\hat{\mathbf{V}}_{\text{merged}}^l$
```

5.7. Discussion on TCA's Generalizability

TCA is designed for VLMs such as CLIP, SigLIP, and SigLIP v2, requiring only minor modifications. These models share a key characteristic: they compute cosine similarity between modalities for zero-shot image classification. For CLIP, we use the `<cls>` token as a guiding indicator throughout the method. In contrast, for the SigLIP series, we take the average over attention weights since their architecture does not include a visual `<cls>` token. The way we determine the domain anchor token and perform token condensation is inherently tied to how each VLM extracts visual features for alignment. We acknowledge that TCA may not directly apply to models like LLaVA [36], as they are not designed for cross-modal alignment but rather for text generation, dictated by their architectural constraints. While this limits direct applicability, it does not diminish TCA's effectiveness in its intended scope. Adapting it to such models would likely require a fundamental architectural redesign.

Table 11. Results on the out-of-distribution benchmark with CLIP ViT-B/16. * denotes the averaged GFLOPs across all datasets.

Method	Aug-free	ImageNet	ImageNet-A	ImageNet-V2	ImageNet-R	ImageNet-S	**Average**	**OOD Average**	**GFLOPs**
CLIP	✓	68.34	49.89	61.88	77.65	48.24	61.20	59.42	17.59
Tent	✓	65.49	44.57	59.26	78.72	22.52	54.11	51.27	17.59
SAR	✓	58.52	33.71	53.95	76.08	39.24	52.30	50.75	17.59
TPT	✗	68.98	54.77	63.45	77.06	47.94	62.44	60.81	1108.61
Diff-TPT	✗	70.30	55.68	65.10	75.00	46.80	62.28	60.52	-
C-TPT	✗	69.30	52.90	63.40	78.00	48.50	62.42	60.70	1108.61
MTA	✗	70.08	58.06	64.24	78.33	49.61	64.06	62.56	-
TDA	✓	69.26	50.82	62.23	77.93	50.26	62.10	60.31	17.59
EViT$_{R=0.95}$	✓	68.32	49.46	61.73	77.00	47.76	60.85	58.99	16.31
ToME$_{R=0.95}$	✓	67.57	48.81	60.88	75.78	47.05	60.02	58.13	16.21
ATS$_{R=0.95}$	✓	65.83	49.80	59.47	71.09	43.38	57.91	55.94	11.50*
TCA$_{R=0.95}$	✓	68.88	50.13	62.10	77.11	48.95	61.43	59.57	16.55

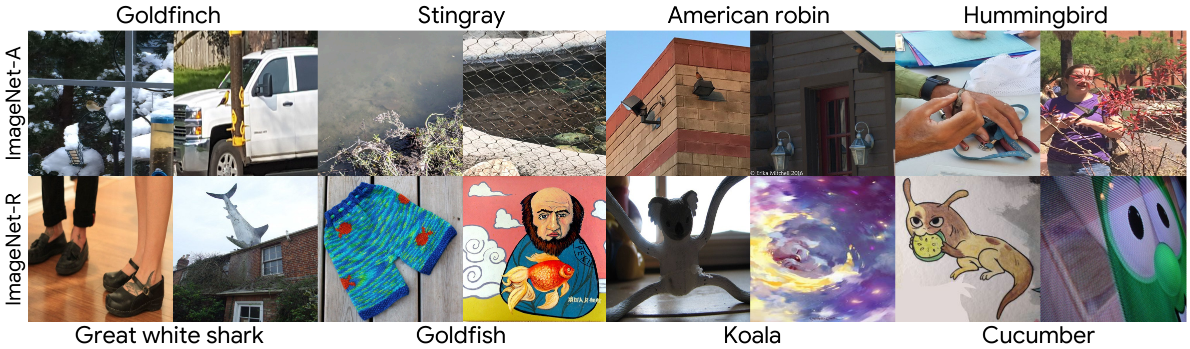

Figure 7. **Sample data from the OOD benchmark.** The samples from the same class exhibit significant diversity. For instance, in the ImageNet-R dataset, one image of a great white shark is dominated by shoes and human legs, while another is on top of a building, showing extreme variability.

5.8. Discussion on the Limitation of TCA

In this section, we discuss the potential limitations of our proposed TCA. Due to the training-free nature of the approach, it is challenging to mitigate the performance gap when the testing domain diverges significantly from the training domain. As observed in the out-of-distribution (OOD) samples shown in Fig. 7, the ground truth object is not always centrally located, and larger class-irrelevant objects (*e.g.,* humans or shoes) can sometimes dominate the prediction. This issue is particularly prominent in CLIP models, where text features for all classes are predefined. When the dominant object is included in the label set, accurately directing visual features to the correct class without additional training becomes difficult. Moreover, the diversity of OOD samples introduces further complexity, especially in the absence of data augmentation. These observations raise important questions for future research: (1) How can we quantify the capacity to mitigate domain shift effectively? (2) What lightweight solutions can be developed for backpropagation and network updates to facilitate test-time adaptation? We leave these questions for future work.

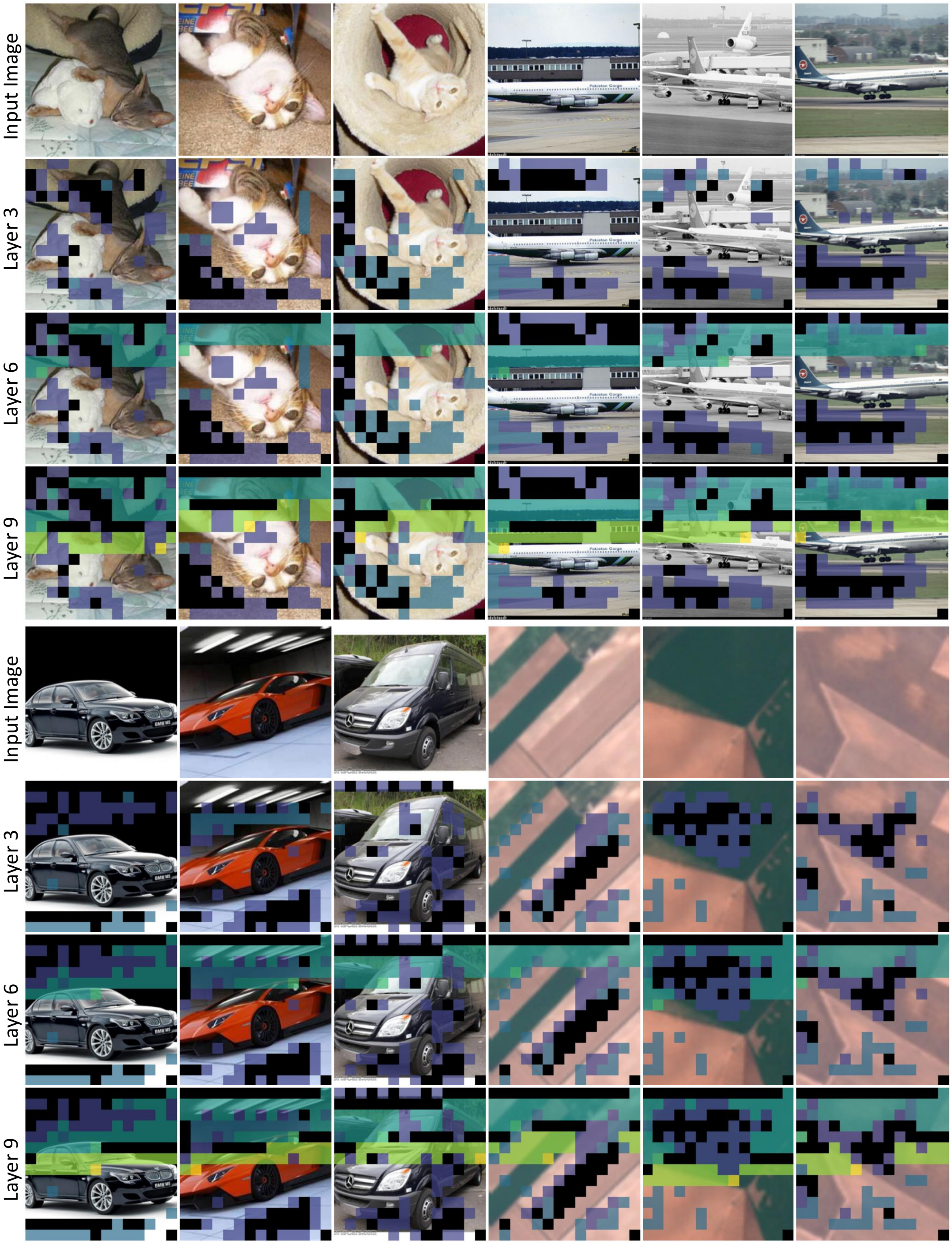

Figure 8. Visualization of our proposed token condensation with $R = 0.7$. Pruned tokens are masked in black, while different colors represent distinct merging clusters.