Test-Time Visual Concept Anchoring via Entropic Optimal Transport

Pawan Kumar
International Institute of Information Technology, Hyderabad, India
pawan.kumar@iiit.ac.in

Abstract

Large vision-language models such as CLIP are widely deployed under conditions that differ from pre-training, causing visual patch tokens to drift from the semantic regions expected by the text-aligned head. We propose test-time concept anchoring (TTCA), a training-free module that treats the visual tokens of a test image as a source measure and a task-conditioned bank of text concepts as a target measure, then solves an entropic optimal transport problem to softly project selected tokens toward semantic anchors before the downstream head consumes them. TTCA operates per sample, requires no backpropagation, and admits an unbalanced variant with a reject sink for open-set noise. On CLIP ViT-B/16, TTCA improves zero-shot accuracy on CIFAR-100 by 1.0%, improves mean accuracy across 9 corruption types by +0.32% (89% of individual conditions improved), reduces distractor-induced accuracy degradation by 41%, and improves CIFAR-100 calibration all at roughly 4 ms per image with no model parameter changes. Project page: https://misterpawan.github.io/tt-visual-concept-project/.

Keywords: test-time adaptation, vision-language models, optimal transport, visual concepts, zero-shot robustness, concept anchoring

1. Introduction

Vision-language models (VLMs) such as CLIP [13] derive much of their transfer ability from broad image–text pre-training, but test-time inputs often differ from the pre-training distribution. Corruptions, stylization, and clutter can degrade zero-shot recognition and calibration, and related representation mismatches are also implicated in hallucination phenomena in downstream generative VLMs [6, 10]. We view these failures through the lens of visual-token misalignment: under shift, patch-level visual features may become less well aligned with task-relevant language concepts.

Many TTA methods update model parameters, normalization layers, or prompts at inference time [16, 19, 21, 24], often using backpropagation and multiple augmented views. These methods often require backpropagation, multiple augmented views, or assumptions about the test batch. TTCA instead keeps the model fixed and adapts only the test-time visual representation.

We introduce test-time concept anchoring (TTCA): for each test image, we encode a compact set of task-relevant text concepts with the frozen text encoder, select a subset of informative visual patch tokens, and solve an entropic optimal transport (OT) problem between the two sets in the shared CLIP latent space [4, 12]. The resulting transport plan yields barycentric projections that softly pull drifted tokens toward plausible semantic anchors. Entropy-aware residual fusion controls the correction strength per token, and an optional reject sink absorbs tokens with no convincing semantic match [2, 14].

TTCA is deliberately simple: it adapts each test image independently, requires no test batch, and adds roughly 4 ms of overhead per image on a modern GPU.

Its key properties are:

- Training-free and sample-wise: no backpropagation, no prompt parameters, and no test-batch dependence.
- Concept-grounded: corrections are guided by task-relevant text anchors, not generic denoising.
- Open-set aware: the reject sink helps avoid forced alignment to irrelevant concepts.

2. Method

Setup. Let $\{z_i\}_{i=1}^{N}$ be the ℓ_2-normalized visual patch tokens of a test image in the shared CLIP space, and g the normalized global image embedding, obtained from the ViT CLS (classification) token. Given a task context (e.g. class names), we build a concept bank of text strings, encode them with the frozen text encoder, and obtain anchor embeddings $\{e_j\}_{j=1}^{M_0}$.

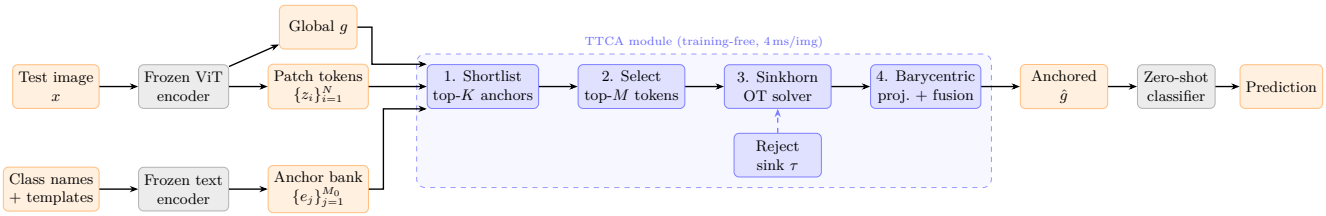

Figure 1. TTCA pipeline. The frozen CLIP encoder produces patch tokens and a global feature. The TTCA module (blue, dashed) shortlists relevant anchors, selects informative tokens, solves entropic OT via Sinkhorn iterations, and fuses corrected tokens into an anchored global feature $\hat{g}$ for classification. An optional reject sink (dashed) absorbs tokens with no semantic match.

Step 1: Anchor shortlisting. We shortlist K anchors most relevant to the current image by scoring each anchor e_j with $u_j = g^\top e_j$ and retaining the top-K. The target marginal is $b_j \propto \exp(u_j/T_b)$.

Step 2: Informative token selection. We score each token by its maximum similarity to the shortlisted anchors,

$$r_i = \max_{j \le K} z_i^\top e_j,$$

and keep the top-M tokens. The source marginal is

$$a_i \propto \exp(r_i/T_a).$$

Unselected tokens pass through unchanged.

Step 3: Entropic OT. We form the cosine-distance cost matrix

$$C_{ij} = 1 - z_i^\top e_j$$

and solve the balanced entropic OT problem

$$\min_{P \in U(a,b)} \langle P, C \rangle + \varepsilon \sum_{ij} P_{ij}(\log P_{ij} - 1), \qquad (1)$$

using Sinkhorn iterations [4]. The row-normalized plan gives weights $w_{ij} = P_{ij}/a_i$.

Step 4: Barycentric target and residual fusion. For each selected token, we compute its OT-weighted concept barycenter

$$\bar{e}_i = \sum_j w_{ij} e_j.$$

Over-correction occurs when an uncertain token is moved too strongly toward an unreliable barycentric target. To avoid this, we apply entropy-aware residual fusion:

$$\hat{z}_i = (1 - \gamma_i)\, z_i + \gamma_i\, \bar{e}_i, \qquad (2)$$

where

$$\gamma_i = \gamma_{\max}(1 - H(w_i)/\log K)$$

and $H(w_i)$ is the entropy of the transport row. Here the factor $1 - H(w_i)/\log K$ is a normalized confidence score for the token–anchor assignment: Since $0 \le H(w_i) \le \log K$, the factor $1 - H(w_i)/\log K$ lies in $[0, 1]$; it equals one for a one-hot transport row and zero for a uniform row. Thus it acts as a normalized confidence score for the token–anchor assignment. Thus, γ_i applies stronger anchoring only when the OT match is confident, while preserving the original token under uncertainty. Sharp rows, corresponding to confident assignments, receive stronger correction; diffuse rows fall back toward the original token.

Step 5: Global feature fusion. The anchored global feature is

$$\hat{g} = \frac{\beta\, g + (1 - \beta) \sum_i a_i\, \hat{z}_i}{\|\beta\, g + (1 - \beta) \sum_i a_i\, \hat{z}_i\|}, \qquad (3)$$

where $\beta \in [0, 1]$ controls the blend between the original global image feature and the anchored token aggregate. Classification proceeds by scoring $\hat{g}$ against the class text embeddings as in standard zero-shot CLIP.

Reject sink. For open-set or cluttered inputs, we append a synthetic sink column with fixed cost τ to the cost matrix and solve OT over the augmented target set. Sink mass continuously gates the residual correction, so poorly matched tokens are kept close to their original representation rather than being forced toward a semantic anchor.

3. Experiments

We evaluate TTCA on CLIP ViT-B/16 and ViT-L/14 (OpenAI checkpoints) across classification, corruption robustness, calibration, and distractor rejection. Default parameters: K=64, M=32, ε=0.05, T=10 Sinkhorn iterations, $\gamma_{\max}$=0.85, β=0.8 (ViT-B/16) or β=0.9 (ViT-L/14), τ=0.35. Templates: "a photo of a {class}." All experiments run on a single NVIDIA RTX 5090.

Algorithm 1 Test-Time Visual Concept Anchoring

Require: Image x; class names $\{s_c\}$; frozen CLIP; $K, M, \varepsilon, L, T_a, T_b, \gamma_{\max}, \beta$; optional sink cost τ
Ensure: Anchored feature $\hat{g}$

$g, \{z_i\}_{i=1}^{N} \leftarrow \text{ClipEncode}(x)$ ▷ normalized global feature and patch tokens
$\{e_j\}_{j=1}^{M_0} \leftarrow \text{TextEncode}(\{s_c\}, \text{templates})$ ▷ normalized text anchors
Compute $u_j = g^\top e_j$ and retain the top-K anchors; set $b_j \propto \exp(u_j/T_b)$
Score tokens by $r_i = \max_{j \le K} z_i^\top e_j$; retain the top-$M$ tokens; set $a_i \propto \exp(r_i/T_a)$
$C_{ij} \leftarrow 1 - z_i^\top e_j$ for selected tokens and shortlisted anchors
if reject sink is used then
    Append a sink column with fixed cost τ to C and allocate a small target mass to the sink
end if
$P \leftarrow \text{Sinkhorn}(a, b, C, \varepsilon, L)$ ▷ entropic OT over real anchors, optionally with sink
for each selected token i do
    $w_i \leftarrow P_i / \|P_i\|_1$
    if reject sink is used then
        $\rho_i \leftarrow w_{i,\text{sink}}$
        $\bar{e}_i \leftarrow \text{normalize}\left(\sum_{j=1}^{K} w_{ij} e_j\right)$
        $\gamma_i \leftarrow \gamma_{\max}(1 - H(w_i)/\log(K+1))\,(1 - \rho_i)$
    else
        $\rho_i \leftarrow 0$
        $\bar{e}_i \leftarrow \text{normalize}\left(\sum_{j=1}^{K} w_{ij} e_j\right)$
        $\gamma_i \leftarrow \gamma_{\max}(1 - H(w_i)/\log K)$
    end if
    $\hat{z}_i \leftarrow (1 - \gamma_i) z_i + \gamma_i \bar{e}_i$ ▷ entropy- and sink-gated residual fusion
end for
$\hat{g} \leftarrow \text{normalize}\left(\beta\, g + (1 - \beta) \sum_i a_i \hat{z}_i\right)$
return $\hat{g}$

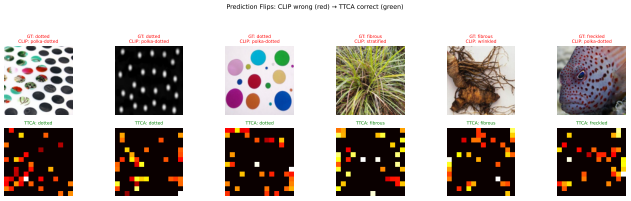

Figure 2. Prediction flips on DTD: CLIP confuses fine-grained textures (red), while TTCA corrects them via concept anchoring (green). Bottom row shows per-patch correction magnitude.

3.1. Zero-shot classification

Table 1 compares CLIP zero-shot, TPT-style entropy minimization [16], SuS-X [18] (enriched multi-template text), MTA [22] (multi-view augmentation averaging), and TTCA across 8 datasets. With per-dataset β tuning, TTCA improves or matches CLIP on all 8 datasets, with the largest gains on DTD (+2.13%), EuroSAT (+1.81%), and CIFAR-100 (+1.0%). TTCA runs at 4 ms/img 185× faster than TPT and 26× faster than MTA while achieving the best accuracy on 3 of 8 datasets.

Table 1. Zero-shot accuracy (%) on 8 datasets (ViT-B/16). TTCA uses per-dataset β. TPT uses 1K-sample subsets. MTA entries are omitted where comparable runs were unavailable. Best in bold.

	CIFAR-100	DTD	EuroSAT	OxPets	CIFAR-10	Cars	Food	IN-R	ms/img
CLIP	62.3	38.9	42.7	85.2	87.7	57.4	84.4	70.9	**1.2**
TPT	63.0	33.3	33.6	83.2	85.6	**64.4**	**88.7**	**94.5**	739
SuS-X	62.7	39.6	41.3	**85.9**	**88.2**	58.0	84.3	72.8	1.5
MTA	59.5	40.3	38.3	–	–	–	–	92.0	106
TTCA	**63.3**	**41.1**	**44.5**	85.6	87.7	57.4	84.4	70.9	4.0

Table 2. CIFAR-100-C mean accuracy (%) per corruption, averaged over 5 severities (ViT-B/16). TTCA improves 8/9 types, 40/45 individual conditions (89%).

	Gau.	Shot	Blur	Mot.	Con.	Bri.	JPG	Pix.	Snow	Mean
CLIP	12.6	5.2	5.5	7.9	49.7	60.1	13.3	**5.0**	41.6	22.3
Ours	**12.8**	**5.6**	**5.8**	**8.3**	**50.2**	**60.5**	**13.5**	**5.0**	**42.1**	**22.7**

3.2. Corruption robustness

We synthesize 9 corruption types at 5 severity levels on CIFAR-100, yielding 45 conditions. Table 2 reports the per-corruption mean accuracy averaged over severities.

The overall delta on CIFAR-100-C is +0.32% in mean accuracy, with 40/45 individual conditions (89%) improved. We also evaluate corruption robustness on DTD, where TTCA improves all 15 conditions tested (5 types × 3 severities), with a mean delta of +3.18%. This confirms that concept anchoring generalizes across domains, with larger gains on harder tasks.

With ViT-L/14, TTCA produces the largest gains under moderate-to-severe shift, such as contrast at severity 5 (+1.45%) and snow at severity 5 (+0.50%).

3.3. Calibration

TTCA reduces the expected calibration error (ECE) on CIFAR-100 from 0.157 to 0.146 (−0.011) with ViT-B/16 and from 0.130 to 0.125 (−0.005) with ViT-L/14, indicating that concept anchoring also tightens confidence–accuracy alignment.

3.4. Distractor rejection

To directly test the reject sink, we create a controlled distractor setting by pasting random CIFAR-10 crops (20–35% area) onto CIFAR-100 test images. These pasted regions introduce salient but label-irrelevant visual tokens, allowing us to test whether the sink avoids

Table 3. Distractor rejection (ViT-B/16, CIFAR-100).

Method	Clean	Distractor	Drop
CLIP zero-shot	63.4	50.8	12.6
TTCA + sink	63.4	56.0	7.4

Table 4. Component ablation on CIFAR-100 (ViT-B/16, 2K samples). Baseline CLIP: 63.3%.

Component	Acc.	Component	Acc.
Default TTCA	64.4	Uniform marginals	64.4
Small OT (K=16, M=8)	64.3	With reject sink	64.1
Large OT (K=100, M=64)	64.4	T=1 Sinkhorn	64.4
Fixed γ=0.5	62.1	T=50 Sinkhorn	64.4
Pure barycenter (γ=1)	59.1	ε=0.001	64.3

forcing such tokens toward CIFAR-100 anchors. Table 3 shows that TTCA with the sink reduces the distractor-induced accuracy drop by 41% (from 12.6% to 7.4%).

3.5. Ablation

Component ablation. Table 4 isolates contributions on CIFAR-100. Entropy-adaptive γ is critical: fixing γ=0.5 loses accuracy, and pure barycentric replacement (γ=1) drops to 59.1%. In contrast, OT parameters (K, M, ε, T) are remarkably robust even T=1 Sinkhorn iteration matches T=50.

Fusion weight β and backbone scaling. β controls how much of the original global image feature is preserved. With per-dataset tuning, the optimal β ranges from 0.70 (DTD, low baseline) to 0.97 (Food-101, high baseline), confirming that stronger baselines need less correction. On ViT-L/14 with per-dataset β, TTCA improves 7/8 datasets, with EuroSAT +2.74%, DTD +0.69%, and OxfordPets +0.46% (see appendix).

4. Related Work

Test-time adaptation for VLMs. TPT [16] and C-TPT [21] adapt prompts via entropy minimization over augmentations, requiring backpropagation through the full model (∼2 s/img). MTA [22] averages logits over entropy-filtered augmented views without backpropagation (∼80 ms/img). SuS-X [18] enriches text representations with diverse templates (Table 1). TTCA differs from all three: it corrects the visual representation directly via OT, at only 4 ms/img.

OT in VLMs; concept bottlenecks. PLOT [1] and AWT [28] use OT during prompt learning; we use OT at test time with a frozen model. Concept bottleneck models [7] learn supervised concept predictors; OPERA [6] and VCD [9] intervene during decoding. TTCA acts earlier, on the representation stream, and is complementary.

5. Conclusion

We presented TTCA, a training-free OT-based module that anchors visual tokens to task-conditioned text concepts at test time. The method consistently improves accuracy and calibration under corruption, reduces vulnerability to distractors via a reject sink, and adds negligible latency.

Future directions. TTCA suggests several extensions. Richer concept banks, built from attributes, parts, materials, or automatically generated descriptions, could provide finer semantic anchors than class names alone. The reject sink could also be generalized to more structured open-set handling, separating background, distractors, and genuinely novel concepts. Beyond zero-shot classification, token-to-concept anchoring may be useful for retrieval, visual question answering, and generative VLMs, where irrelevant visual evidence and hallucination remain important challenges. Finally, adaptive selection of fusion and OT hyperparameters could further reduce per-dataset tuning while retaining the training-free, sample-wise nature of TTCA.

Acknowledgments. This research was supported by a Qualcomm Faculty Grant for compute resources at the International Institute of Information Technology, Hyderabad.

References

[1] Guangyi Chen, Weiran Yao, Xiangchen Song, Xinyue Li, Yongming Rao, and Kun Zhang. PLOT: Prompt learning with optimal transport for vision-language models. In The Eleventh International Conference on Learning Representations, 2023. 4, 8

[2] Lénaïc Chizat, Gabriel Peyré, Bernhard Schmitzer, and François-Xavier Vialard. Scaling algorithms for unbalanced transport problems. Mathematics of Computation, 87(314):2563–2609, 2018. 1, 8

[3] Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. IEEE Trans. Pattern Anal. Mach. Intell., 39 (9):1853–1865, 2017. 8

[4] Marco Cuturi. Sinkhorn distances: lightspeed computation of optimal transport. In Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, page 2292–2300, Red Hook, NY, USA, 2013. Curran Associates Inc. 1, 2, 8

[5] Bharath Bhushan Damodaran, Benjamin Kellenberger, Rémi Flamary, Devis Tuia, and Nicolas Courty. Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation. In Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part IV, page 467–483, Berlin, Heidelberg, 2018. Springer-Verlag. 8

[6] Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. Opera: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 13418–13427, 2024. 1, 4, 8

[7] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In Proceedings of the 37th International Conference on Machine Learning, pages 5338–5348. PMLR, 2020. 4, 8

[8] Youngjun Lee, Doyoung Kim, Junhyeok Kang, Jihwan Bang, Hwanjun Song, and Jae-Gil Lee. RA-TTA: Retrieval-augmented test-time adaptation for vision-language models. In The Thirteenth International Conference on Learning Representations, 2025. 8

[9] Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 13872–13882, 2024. 4, 8

[10] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 292–305, Singapore, 2023. Association for Computational Linguistics. 1, 8

[11] XIAOSONG MA, Jie ZHANG, Song Guo, and Wenchao Xu. Swapprompt: Test-time prompt adaptation for vision-language models. In Advances in Neural Information Processing Systems, pages 65252–65264. Curran Associates, Inc., 2023. 8

[12] Gabriel Peyré and Marco Cuturi. Computational optimal transport. Found. Trends Mach. Learn., 11(5–6): 355–607, 2019. 1, 8

[13] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Proceedings of the 38th International Conference on Machine Learning, pages 8748–8763. PMLR, 2021. 1

[14] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 4937–4946, 2020. 1

[15] Lijun Sheng, Jian Liang, Ran He, Zilei Wang, and Tieniu Tan. The illusion of progress? a critical look at test-time adaptation for vision-language models. In The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track, 2026. 8

[16] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. In Proceedings of the 36th International Conference on Neural Information Processing Systems, Red Hook, NY, USA, 2022. Curran Associates Inc. 1, 3, 4, 8

[17] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In Proceedings of the 37th International Conference on Machine Learning, pages 9229–9248. PMLR, 2020. 7

[18] Vishaal Udandarao, Ankush Gupta, and Samuel Albanie. SuS-X: Training-free name-only transfer of vision-language models. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2023. 3, 4, 8

[19] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In International Conference on Learning Representations, 2021. 1, 7

[20] Yue Yang, Artemis Panagopoulou, Shenghao Zhou, Daniel Jin, Chris Callison-Burch, and Mark Yatskar. Language in a bottle: Language model guided concept bottlenecks for interpretable image classification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 19187–19197, 2023. 8

[21] Hee Suk Yoon, Eunseop Yoon, Joshua Tian Jin Tee, Mark A. Hasegawa-Johnson, Yingzhen Li, and Chang D. Yoo. C-TPT: Calibrated test-time prompt tuning for vision-language models via text feature dispersion. In The Twelfth International Conference on Learning Representations, 2024. 1, 4, 8

[22] Maxime Zanella and Ismail Ben Ayed. On the test-time zero-shot generalization of vision-language models: Do we really need prompt learning? In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 23783–23793, 2024. 3, 4, 8

[23] Maxime Zanella, Benoît Gérin, and Ismail Ben Ayed. Boosting vision-language models with transduction. In The Thirty-eighth Annual Conference on Neural Information Processing Systems, 2024. 8

[24] Marvin Zhang, Sergey Levine, and Chelsea Finn. Memo: test time robustness via adaptation and augmentation. In Proceedings of the 36th International

Conference on Neural Information Processing Systems, Red Hook, NY, USA, 2022. Curran Associates Inc. 1, 8

[25] Shuai Zhao, Xiaohan Wang, Linchao Zhu, and Yi Yang. Test-time adaptation with CLIP reward for zero-shot generalization in vision-language models. In The Twelfth International Conference on Learning Representations, 2024. 8

[26] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 16795–16804, 2022. 8

[27] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. Int. J. Comput. Vision, 130(9):2337–2348, 2022. 8

[28] Yuhan Zhu, Yuyang Ji, Zhiyu Zhao, Gangshan Wu, and Limin Wang. Awt: transferring vision-language models via augmentation, weighting, and transportation. In Proceedings of the 38th International Conference on Neural Information Processing Systems, Red Hook, NY, USA, 2024. Curran Associates Inc. 4, 8

Supplementary Material

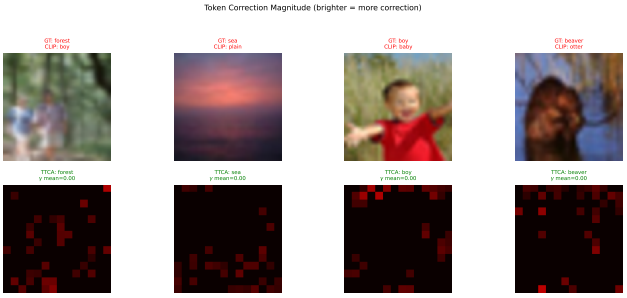

Figure A1. Token correction heatmaps on CIFAR-100. Top: input images with CLIP predictions (red = wrong). Bottom: per-patch correction magnitude (brighter = more correction). TTCA corrects all four predictions.

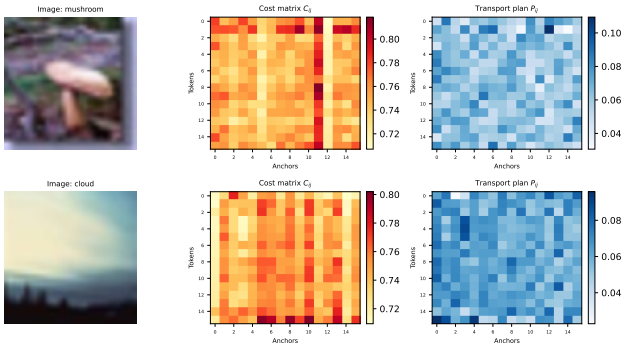

Figure A2. Cost matrix C_{ij} (left) and Sinkhorn transport plan P_{ij} (right) for two CIFAR-100 images. In the cost map, lighter/yellow cells indicate lower cosine distance and darker red cells indicate higher distance; in the transport map, darker blue cells indicate larger transported mass. Thus high mass generally appears on low-cost regions, but not in exact one-to-one correspondence because Sinkhorn also enforces the source and target marginals and smooths assignments through the entropy temperature ε.

A. Additional Qualitative Results

Figures A4–A7 show extended token correction visualizations across four datasets. Each panel pairs the input image (top, with CLIP prediction) and the per-patch correction heatmap (bottom, with TTCA prediction). Red titles indicate wrong predictions; green indicates correct. The first rows show prediction flips (CLIP wrong → TTCA correct); the last row shows images where both are correct, illustrating that TTCA still applies targeted corrections to semantically active patches.

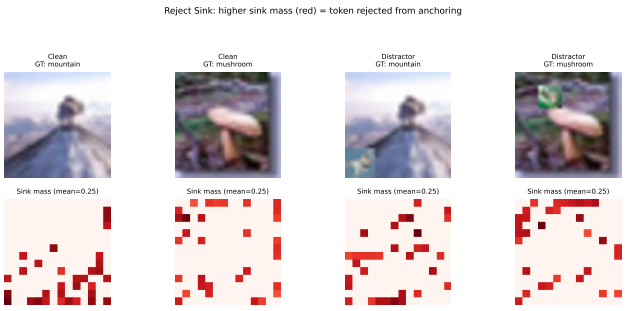

Figure A3. Reject sink visualization. The top row shows two clean CIFAR-100 images followed by two images with pasted CIFAR-10 distractor crops. The bottom row shows the sink mass assigned to the selected TTCA patch tokens; darker red means that a larger fraction of that token's transport mass goes to the reject sink, so its residual correction is more strongly gated. The heatmap is therefore a token-level abstention signal, not a pixel-level segmentation mask: only selected tokens are shown, each token covers a coarse ViT patch, and a distractor region need not become uniformly red if some of its patches still match shortlisted CIFAR-100 anchors or overlap with background content.

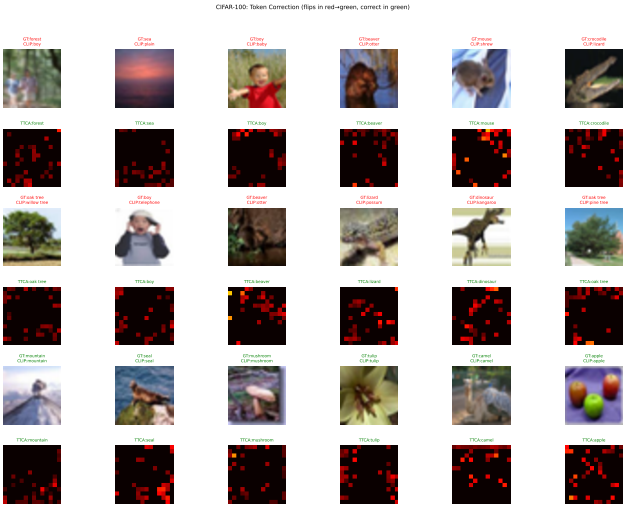

Figure A4. CIFAR-100: 12 prediction flips + 6 correct examples. TTCA corrects confusions between visually similar classes (e.g., beaver/otter, oak/willow tree, boy/baby).

B. Extended Related Work

This appendix provides an extended literature review that contextualizes TTCA within four research streams.

B.1. General test-time adaptation

Test-time adaptation studies how to improve a model using only unlabeled test inputs. Early work on test-time training [17] introduced a self-supervised auxiliary objective optimized on the current example. Tent [19]

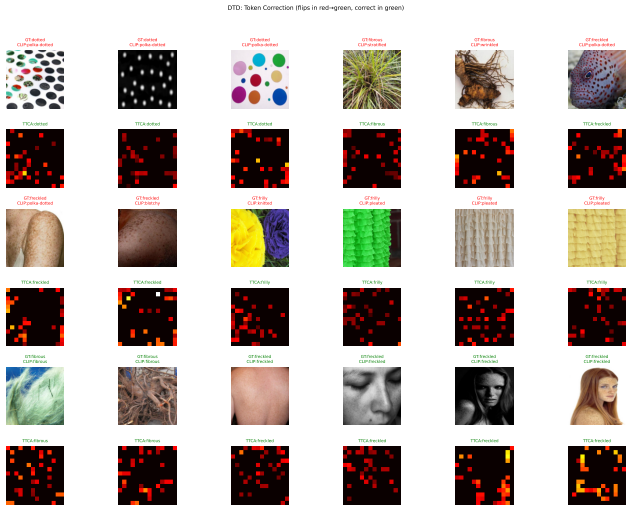

Figure A5. DTD (Describable Textures): prediction flips on fine-grained texture classes. Concept anchoring resolves subtle distinctions like dotted vs. polka-dotted, fibrous vs. stratified.

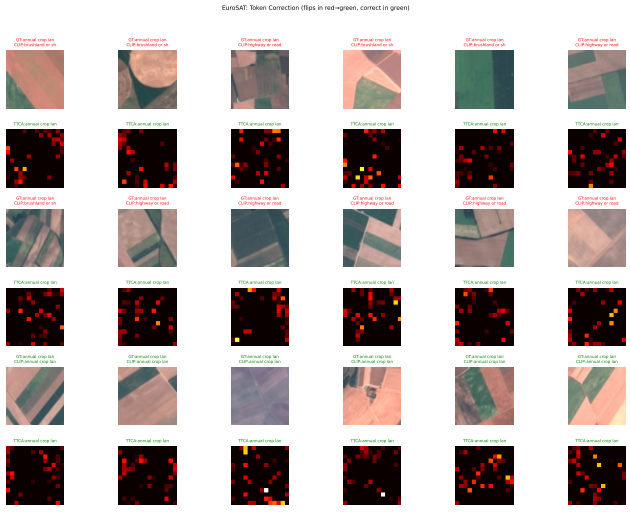

Figure A6. EuroSAT (satellite imagery): correction examples. TTCA helps ground satellite scene tokens toward the correct land-use concept anchors.

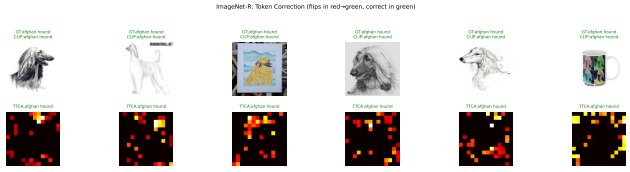

Figure A7. ImageNet-R (renditions): correction examples on artistic depictions of objects. With high β=0.97, TTCA applies minimal but targeted corrections.

later simplified the idea by minimizing prediction entropy while updating normalization parameters online. MEMO [24] emphasized augmentation consistency by minimizing the entropy of the marginal prediction across augmentations. These methods helped establish the modern TTA paradigm, but they were developed mainly for conventional classifiers and often rely on parameter updates and repeated forward/backward passes.

Our proposal inherits the single-example adaptation ethos of this line, but diverges in mechanism. We do not optimize model weights or prompts. Instead, we insert a representation-level geometric correction step between the vision encoder and the downstream head. This makes the method immediately compatible with frozen backbones and attractive when even small parameter changes are undesirable.

B.2. Prompt learning and test-time transfer for VLMs

For CLIP-like models, prompt learning became a prominent adaptation route after CoOp and CoCoOp [26, 27] demonstrated that lightweight prompt parameters could rival manual templates under supervision. Test-time prompt tuning then moved adaptation into inference time: TPT [16] adapts prompts from a single test input via entropy minimization over augmentations; C-TPT [21] supplements accuracy with calibration-oriented dispersion control; RLCF [25] uses reinforcement-style feedback from a frozen CLIP scorer and extends beyond classification to retrieval and captioning; SwapPrompt [11] injects contrastive self-supervision into online prompt adaptation. In parallel, training-free transfer methods such as SuS-X [18] show that strong zero-shot gains can be obtained from better textual descriptions even without test-time optimization.

Recent work also highlights batch-based or transductive improvements. MTA [22] argues that multi-view inference can outperform prompt learning under some protocols; TransCLIP [23] leverages unlabeled batch structure explicitly; RA-TTA [8] augments test-time prediction with retrieved support information. A recent critique points out that progress claims in VLM TTA can depend strongly on experimental assumptions, including favorable class composition and transductive availability [15].

Relative to these works, our method occupies a distinct corner of the design space: TTCA requires no backpropagation and performs no model, prompt, or classifier-parameter updates; its only test-time optimization is the Sinkhorn solve for a small entropic OT plan.

Rather than adapting prompts or classifier heads, TTCA adapts the geometry of the test-time visual token representations.

B.3. Optimal transport for adaptation and multimodal alignment

Optimal transport has a long history in domain adaptation and distribution alignment. Entropic regularization made OT practically scalable via Sinkhorn iterations [4]. OT-based domain adaptation aligns source and target samples under label or structure constraints [3], while DeepJDOT [5] aligns joint feature–label distributions in deep networks. Unbalanced OT extends the framework to settings where mass should be created or destroyed rather than strictly preserved [2]; the monograph by Peyré and Cuturi [12] provides a broad treatment.

Within VLMs, OT has already appeared in supervised or few-shot prompt learning. PLOT [1] uses OT to align learned prompts and visual features during prompt optimization; AWT [28] applies OT to augmented image and text views for downstream transfer. These are important precedents, but their emphasis is still on training or prompt learning. Our method instead uses OT at test time, with frozen models, to map visual tokens toward text concepts on the current example.

B.4. Concept grounding, bottlenecks, and hallucination mitigation

Here, a concept denotes a human-interpretable semantic attribute, part, object, or category-level property used to describe visual content.

Concept bottleneck models [7] predict interpretable concepts before labels and support concept intervention at test time. Language-guided bottlenecks [20] show that concept spaces can be generated with language models rather than fully hand-crafted. Our work is related in spirit both seek semantically meaningful intermediate structure but differs fundamentally in supervision and objective. We do not train a supervised concept predictor or require concept annotations. The anchors are soft, task-conditioned textual prototypes used only during inference.

For large multimodal generative models, hallucination benchmarks such as POPE [10] have made the problem concrete. Decoding-time interventions including OPERA [6] and VCD [9] aim to mitigate hallucination after visual representations have already been computed. Those methods are complementary to ours: concept anchoring acts earlier, on the visual representation stream, and can in principle be combined with decoding-time safeguards.

B.5. Novelty summary

Existing OT-based VLM work mostly optimizes prompts or transfer objectives offline or in few-shot settings. Existing test-time VLM adaptation mostly optimizes prompts, logits, or batch-level decision rules. Existing concept bottleneck work learns explicit concept predictors with concept supervision. In contrast, TTCA is a single-example, training-free, token-level, concept-anchored OT layer with an explicit reject option. That combination, to our knowledge, is new.

C. Theoretical Properties

We formalize why barycentric anchoring is a sensible correction once a transport row has been estimated.

Lemma C.1 (Barycentric optimality). Let $w_1, \dots, w_K$ be nonnegative weights summing to one, and let $e_1, \dots, e_K \in \mathbb{R}^d$ be anchor vectors. The barycenter $\bar{e} = \sum_{j=1}^K w_j e_j$ is the unique minimizer of

$$u \mapsto \sum_{j=1}^K w_j \|u - e_j\|_2^2.$$

Proof. Expanding the objective gives

$$\sum_j w_j \|u - e_j\|_2^2 = \|u\|_2^2 - 2u^\top \sum_j w_j e_j + \sum_j w_j \|e_j\|_2^2,$$

where we used $\sum_j w_j = 1$. Its gradient is

$$2u - 2\sum_j w_j e_j,$$

which vanishes uniquely at $u = \bar{e}$. □

Proposition C.2 (Transport-weighted discrepancy reduction). Let $w_{ij} \geq 0$ be row weights over the real anchors with $\sum_{j=1}^K w_{ij} = 1$, and define the Euclidean barycenter $\bar{e}_i = \sum_{j=1}^K w_{ij} e_j$. Then, for every selected token i,

$$\sum_{j=1}^K w_{ij} \|\bar{e}_i - e_j\|_2^2 \leq \sum_{j=1}^K w_{ij} \|z_i - e_j\|_2^2. \tag{A.1}$$

Proof. By Lemma C.1, $\bar{e}_i$ minimizes $u \mapsto \sum_j w_{ij} \|u - e_j\|_2^2$. Evaluating this objective at $u = z_i$ gives the result. □

Theorem C.3 (Monotonic improvement under residual anchoring). Define

$$D_i(u) = \sum_{j=1}^K w_{ij} \|u - e_j\|_2^2,$$

where $w_{ij} \geq 0$ and $\sum_j w_{ij} = 1$, and let $\widetilde{z}_i = \sum_j w_{ij} e_j$. For $u_\gamma = (1-\gamma) z_i + \gamma \widetilde{z}_i$ with $\gamma \in [0, 1]$, $D_i(u_\gamma)$ is nonincreasing in γ. In particular,

$$D_i(u_\gamma) \leq D_i(z_i).$$

Proof. By Lemma C.1, $\widetilde{z}_i$ is the minimizer of D_i. Since $\sum_j w_{ij} = 1$, expanding around the minimizer gives

$$D_i(u) = D_i(\widetilde{z}_i) + \|u - \widetilde{z}_i\|_2^2.$$

Therefore,

$$D_i(u_\gamma) = D_i(\widetilde{z}_i) + \|(1-\gamma)(z_i - \widetilde{z}_i)\|_2^2 \tag{A.2}$$

$$= D_i(\widetilde{z}_i) + (1-\gamma)^2 \|z_i - \widetilde{z}_i\|_2^2. \tag{A.3}$$

This expression is nonincreasing for $\gamma \in [0, 1]$, and at $\gamma = 0$ equals $D_i(z_i)$. □

Remark C.4. The theorem justifies residual anchoring: increasing γ can only decrease the row-wise anchor discrepancy under the quadratic proxy. Residual fusion is therefore a controlled, monotonically improving path from the original token to the semantically anchored token.

Proposition C.5 (Row-wise sink preference under a margin condition). Consider a row-wise Gibbs relaxation with semantic anchor costs C_{ij} and sink cost τ, yielding $w_{ij} \propto \exp(-C_{ij}/\varepsilon)$ and $w_{i\varnothing} \propto \exp(-\tau/\varepsilon)$. If $C_{ij} \geq \tau + \delta$ for all semantic anchors j and some $\delta > 0$, then

$$w_{i\varnothing} \geq \frac{1}{1 + K \exp(-\delta/\varepsilon)}. \tag{A.4}$$

Proof. For each semantic anchor,

$$\exp(-C_{ij}/\varepsilon) \leq \exp(-\tau/\varepsilon) \exp(-\delta/\varepsilon).$$

Hence

$$\sum_{j=1}^K \exp(-C_{ij}/\varepsilon) \leq K \exp(-\tau/\varepsilon) \exp(-\delta/\varepsilon).$$

Normalizing the sink weight gives the claimed bound. □

Remark C.6. This formalizes the intuition that when every semantic anchor is uniformly worse than the sink by a clear margin, entropy-regularized assignment strongly favors rejection exactly the desired behavior for irrelevant clutter.

D. Experimental Setup Details

D.1. Implementation

TTCA is implemented in PyTorch and uses the Python Optimal Transport (POT) library for Sinkhorn iterations. Visual patch tokens are extracted from the frozen CLIP ViT using the output_tokens interface in the open_clip library, followed by the model's ln_post layer normalization and projection via the proj matrix to map tokens into the shared CLIP latent space. All tokens and anchors are ℓ_2-normalized before computing cosine distances.

D.2. Concept bank construction

For classification, the concept bank consists of class names wrapped in the template "a photo of a {class}." Each class name produces one anchor embedding via the frozen CLIP text encoder. For CIFAR-100 this yields $M_0 = 100$ anchors; for ImageNet-R, $M_0 = 200$ anchors. From these, $K = 64$ are shortlisted per image based on global-feature similarity.

D.3. Corruption generation

We synthesize 9 corruption types on CIFAR-100 test images: Gaussian noise (std $= s \times 15$), shot noise (Poisson with scale $1/(10s)$), Gaussian blur (radius $= 1.5s$), motion blur (radius $= 1.2s$), contrast reduction (factor $= \max(0.1, 1 - 0.18s)$), brightness reduction (factor $= \max(0.2, 1 - 0.15s)$), JPEG compression (quality $= \max(3, 40 - 8s)$), pixelation (downscale factor $= 2s$), and snow (additive Gaussian with mean 200, scaled by $0.1s$), where $s \in \{1, 2, 3, 4, 5\}$ is the severity level.

D.4. Distractor overlay

For the controlled distractor experiment (Section 3.4 of the main paper), we randomly select a CIFAR-10 image, resize it to occupy 20–35% of the CIFAR-100 base image area (preserving aspect ratio), and alpha-composite it at a random position. The base label is preserved. We evaluate 500 clean and 500 distractor-overlaid images.

D.5. Hardware and runtime

All experiments run on a single NVIDIA RTX 5090 GPU (32 GB VRAM) with PyTorch 2.10 and CUDA 13.2. ViT-B/16 inference uses ∼2 GB VRAM; TTCA adds negligible memory overhead since the OT problem is solved on CPU via POT and the matrices are small ($M \times K = 32 \times 64$).

E. Additional Results

E.1. Per-severity corruption results

Table A1 provides the full per-corruption, per-severity accuracy delta (TTCA − CLIP) for ViT-B/16. Positive entries (40 out of 45) are shown in bold.

Table A1. Per-severity accuracy delta (TTCA − CLIP, in %) on CIFAR-100-C with ViT-B/16. Bold indicates improvement.

Corruption	s=1	s=2	s=3	s=4	s=5
Gaussian noise	−0.22	+0.33	+0.74	+0.22	+0.30
Shot noise	+1.09	+0.30	+0.41	+0.31	+0.02
Blur	+0.60	+0.13	+0.25	+0.31	+0.31
Motion blur	+1.10	+0.06	+0.24	+0.34	+0.31
Contrast	+0.26	+0.52	+0.61	+0.33	+0.66
Brightness	+0.20	+0.30	+0.40	+0.35	+0.53
JPEG	−0.17	+0.09	+0.22	+0.33	+0.20
Pixelate	+0.36	+0.32	−0.13	−0.34	−0.48
Snow	+0.18	+0.79	+0.42	+0.76	+0.39
Mean	+0.38	+0.32	+0.35	+0.29	+0.25

E.2. ViT-L/14 selected corruption results

Table A2 reports selected ViT-L/14 results with the recommended β=0.9. The largest gains appear under moderate-to-severe shift, consistent with the thesis that concept anchoring is most useful when visual tokens have drifted substantially from the language-aligned space.

Table A2. ViT-L/14 results with per-dataset β tuning. TTCA improves 7/8 datasets.

Dataset	β	CLIP	TTCA	Δ
EuroSAT	0.85	32.70	35.44	+2.74
DTD	0.93	52.61	53.30	+0.69
OxfordPets	0.93	90.22	90.68	+0.46
Food-101	0.85	89.95	90.15	+0.19
CIFAR-10	0.85	93.11	93.24	+0.13
CIFAR-100	0.97	73.05	73.11	+0.06
ImageNet-R	0.95	83.27	83.28	+0.01
StanfordCars	0.99	70.82	70.77	−0.05

E.3. OT parameter sensitivity

Table A3 provides the full OT parameter sweep on CIFAR-100 (ViT-B/16, 1,000 samples). Performance is remarkably stable across a wide range of configurations, indicating that TTCA does not require careful hyperparameter tuning.

E.4. Calibration details

Table A4 reports expected calibration error (ECE, 15 bins) for both backbones. TTCA consistently reduces

Table A3. Full OT parameter sweep (CIFAR-100, ViT-B/16).

Config	K	M	ε	T	Acc. (%)	ms/img
Baseline (CLIP)	–	–	–	–	63.40	1.2
K=16, M=8, T=5	16	8	0.05	5	63.80	4.0
K=32, M=16	32	16	0.05	10	63.70	4.1
K=64, M=32	64	32	0.05	10	64.20	4.0
K=64, M=32, ε=0.02	64	32	0.02	10	64.20	4.2
K=64, M=32, ε=0.10	64	32	0.10	10	64.20	4.1
K=64, M=32, T=20	64	32	0.05	20	64.20	4.1
K=128, M=64	128	64	0.05	10	64.20	4.1
K=64, M=32 + sink	64	32	0.05	10	63.70	4.4

ECE on CIFAR-100, indicating improved confidence–accuracy alignment.

Table A4. Expected calibration error (ECE ↓, 15 bins).

	ViT-B/16		ViT-L/14	
Dataset	CLIP	TTCA	CLIP	TTCA
CIFAR-100	0.157	0.146	0.130	0.125
ImageNet-R	0.047	0.058	0.054	0.067

E.5. Full dataset results with SuS-X comparison

The full 8-dataset accuracy comparison for CLIP zero-shot, SuS-X, and TTCA is reported in Table 1; we do not repeat the same numbers here. TTCA outperforms both CLIP and SuS-X on CIFAR-100, DTD, and EuroSAT, where the zero-shot baseline is lowest, suggesting that concept anchoring is most effective when visual tokens are poorly aligned. On well-aligned datasets (Food-101, StanfordCars, ImageNet-R), SuS-X's richer text representations provide a complementary advantage.

E.6. Component ablation details

The component ablation on CIFAR-100 (2,000 samples, ViT-B/16) is summarized in Table 4; we do not repeat the same experiment here. The key finding is that entropy-adaptive γ is the most important design choice: without it, performance degrades significantly. All other OT parameters (K, M, ε, T) can vary by an order of magnitude with negligible effect on accuracy.

E.7. Notation summary

Table A5 collects the main symbols used in the paper for reference.

Table A5. Summary of notation.

Symbol	Meaning
z_i	Normalized visual patch token in shared CLIP space
g	Normalized global (CLS) image embedding
e_j	Normalized text anchor embedding
M_0	Full anchor bank size
K	Number of shortlisted anchors
M	Number of selected informative tokens
a, b	Source (token) and target (anchor) marginals
C	Cosine-distance cost matrix
P	Optimal transport plan
ε	Entropic regularization strength
T	Number of Sinkhorn iterations
w_{ij}	Row-normalized transport weight
$\widetilde{z}_i$	Barycentric projection of token z_i
γ_i	Per-token anchoring strength (entropy-adaptive)
$\gamma_{\max}$	Maximum anchoring strength
$\hat{z}_i$	Residual-fused anchored token
β	Global feature fusion weight
$\hat{g}$	Anchored global feature
τ	Reject sink cost