A-TPT: Angular Diversity Calibration Properties for Test-Time Prompt Tuning of Vision-Language Models

Shihab Aaqil Ahamed[1, 2] Udaya S.K.P. Miriya Thanthrige[1] Ranga Rodrigo[1]
Muhammad Haris Khan[2]

[1]Dept. of Electronic and Telecommunication Engineering, University of Moratuwa, Sri Lanka

[2]Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE

The Fourteenth International Conference on Learning Representations
ICLR 2026

Outline

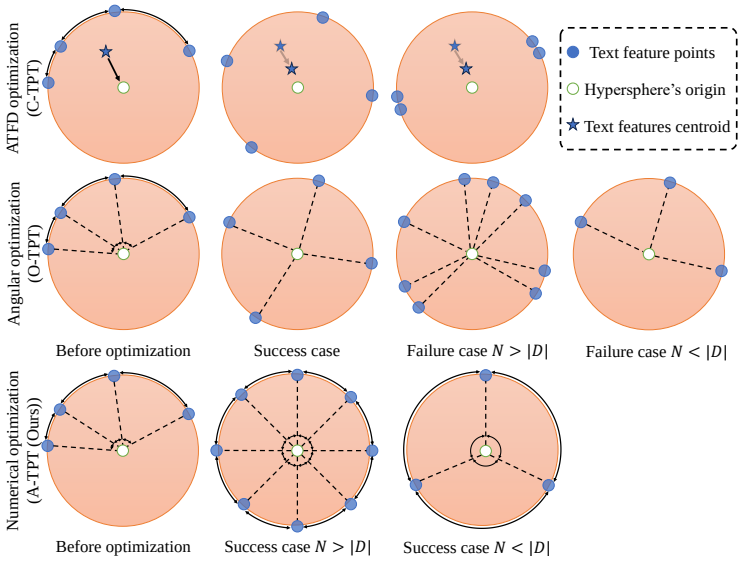
Background

- Test-time prompt tuning (TPT) is a parameter-efficient technique for adapting large vision-language models (VLMs) to unseen tasks at inference time without labeled data.
- While TPT improves accuracy, it often suffers from poor calibration, causing models to make overconfident or unreliable predictions.
- Current approaches address this by maximizing textual feature dispersion, but they often fail to achieve optimal angular separation, overlooking the critical role of angular diversity

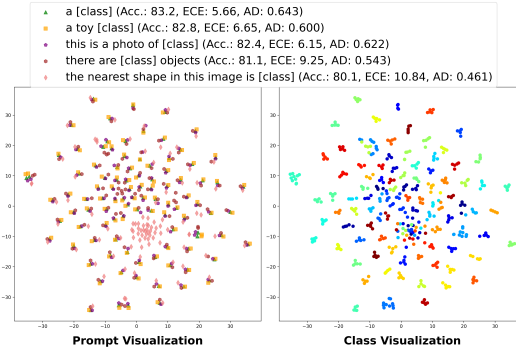
Contributions

- We introduce a numerical optimization method, called A-TPT, for better calibration of test-time prompt tuning for VLMs. This resolves the suboptimal performance of existing leading calibration techniques for test-time prompt tuning.
- We introduce novel angular diversity that effectively promotes the diversity among textual features, thereby improving the calibration capabilities of VLMs when $N > |D|$ and $N < |D|$. This is accomplished by maximizing the minimum pairwise angular distance between normalized textual features.
- We conduct extensive experiments to validate the generalizability of our approach on different datasets, including medical datasets, across various baselines. The results show that A-TPT surpasses state-of-the-art methods in calibration performance. We also provide thorough analyses, including theoretical aspects. Moreover, our approach provides superior calibration compared to the zero-shot CLIP model, which reveals improved calibration.

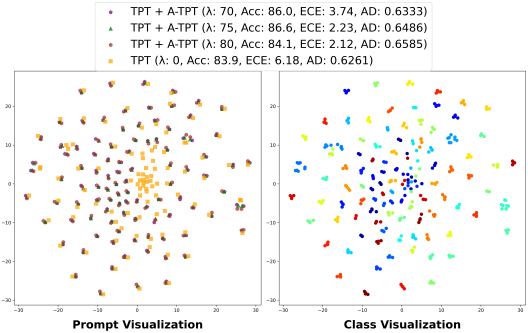
Method


ATFD optimization (C-TPT)
Angular optimization (O-TPT)
Numerical optimization (A-TPT (Ours))
Text feature points
Hypersphere's origin
Text features centroid
Before optimization
Success case
Failure case $N > |D|$
Failure case $N < |D|$
Before optimization
Success case $N > |D|$
Success case $N < |D|$

Grounded & Theoretical Analysis



(a) Hard prompt



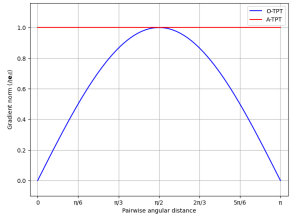
(b) Tuned prompt

Gradient Norm

In this paper, we optimize angular distance rather than cosine similarity or L2 distance.

$$\left\| \frac{\partial\, \hat{\mathbf{E}}\hat{\mathbf{E}}^T}{\partial \mathbf{e}_i} \right\| = \frac{\|(I - M_{\mathbf{e}_i})\, \mathbf{e}_j\|}{\|\mathbf{e}_i\|\, \|\mathbf{e}_j\|} = \frac{\|\sin\theta_{ij}\|}{\|\mathbf{e}_i\|},\ M_{\mathbf{e}_i} = \frac{\mathbf{e}_i \mathbf{e}_i^T}{\|\mathbf{e}_i\|^2}$$

$$\left\| \frac{\partial \theta_{ij}}{\partial \mathbf{e}_i} \right\| = \left\| \frac{\partial \theta_{ij}}{\partial\, \hat{\mathbf{E}}\hat{\mathbf{E}}^T} \frac{\partial\, \hat{\mathbf{E}}\hat{\mathbf{E}}^T}{\partial \mathbf{e}_i} \right\| = \frac{1}{\sin\theta_{ij}} \frac{\sin\theta_{ij}}{\|\mathbf{e}_i\|} = \frac{1}{\|\mathbf{e}_i\|}$$



Algorithm

A mini-batch `AdamW` is used for solving the "max-min" Tammes' problem.

$$\max_{\theta} \frac{1}{N} \sum_{i=1}^{N} \min_{j \in \{1,\ldots,N\} \setminus \{i\}} \theta_{ij}, \quad \theta = \arccos(\hat{\mathbf{E}}\hat{\mathbf{E}}^T), \quad \text{s.t. } \# \; \forall_i \; \hat{\mathbf{E}}_i = \frac{\mathbf{e}_i^T}{|\mathbf{e}_i|},$$

Algorithm 1: Angular Diversity

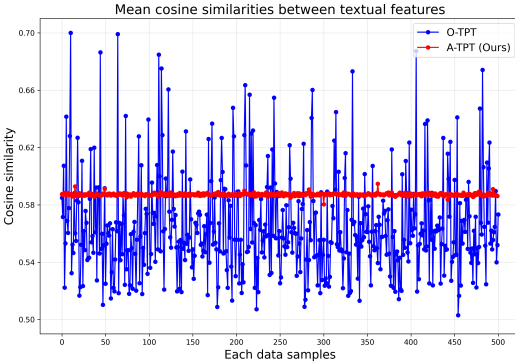
while *test sample not finished* **do**
    Initialize prompt p with pretrained state;
    Generate T augmented views of the input image;
    for $t \leftarrow 1$ *to* T **do**
        Compute logits and select low-entropy subset;
        Compute $\mathcal{L}_{\text{TPT}}$ and extract text features $\mathbf{E}$;
        Calculate pairwise angular distance via nearest neighbors;
        Update p to minimize objective via gradient descent with learning rate ζ;
    Infer final label using adapted p on original image;

PyTorch Implementation

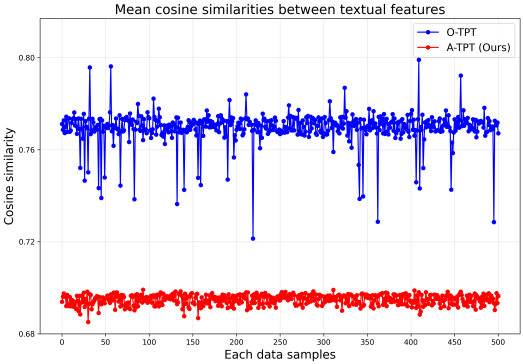
Source code: https://github.com/MB-Shihab-Aaqil-Ahamed/A-TPT/

```
optimizer = AdamW(params, lr=5e-3)
for image in test_loader:
    def closure():
        optimizer.zero_grad()
        views = augment(image)
        logits = model(views, p)
        loss = select_and_compute_entropy(logits)
        loss += angular_diversity(text_features)
        loss.backward()
        return outputs, loss
    outputs, loss = optimizer.step(closure)
```

Hypersphere's Optimal Text Feature Seperation:



(a) $N > |D|$



(b) $N < |D|$

Experimental Setup

Backbones:

- CLIP ViT-B/16
- CLIP ResNet-50

Metrics:

- Accuracy
- Expected Calibration Error (ECE)
- Static Calibration Error (SCE)

Image classification datasets:

- Fine-grained: Food101, DTD, Pets, Cars, etc...
- General: ImageNet, Caltech101
- Natural distribution shifts: ImageNet-A, V2, R, S

Results on Image Classification Benchmarks

Method	Metric	ImageNet	DTD	Flowers102	Food101	SUN397	Aircrafts	OxfordPets	Caltech101	UCF101	EuroSAT	Stanford Cars	*Average*
Pre-trained Backbone: CLIP ViT-B/16 | *Embedding dimension: 512-d*													
2*Baseline	Acc.	66.70	44.30	67.30	83.60	62.50	23.90	88.00	92.90	65.00	41.30	65.30	63.70
	ECE	2.12	8.50	3.00	2.39	2.53	5.11	4.37	5.50	3.59	13.89	4.25	4.43
2*TPT	Acc.	69.00	46.70	69.00	84.70	64.50	23.40	87.10	93.80	67.30	42.40	66.30	65.00
	ECE	10.60	21.20	13.50	3.98	11.30	16.80	5.77	4.51	2.54	13.20	5.16	11.60
2*C-TPT	Acc.	68.50	46.00	69.80	83.70	64.80	24.85	88.20	93.63	65.70	43.20	65.80	64.57
	ECE	3.15	11.90	5.04	3.43	5.04	4.36	1.90	4.24	2.54	13.20	1.59	5.13
2*O-TPT	Acc.	67.33	45.68	70.07	84.13	64.23	23.64	87.95	93.95	64.16	42.84	64.53	64.41
	ECE	1.96	7.88	3.87	1.46	4.93	3.68	1.90	3.80	2.34	12.98	1.78	4.23
pink!20	**Acc.**	67.70	45.51	69.22	83.64	66.04	23.76	88.33	93.87	66.16	44.06	65.78	64.92
pink!20 -2***A-TPT (Ours)**	**ECE**	1.45	4.76	3.61	1.37	3.28	3.14	1.17	2.76	2.12	3.92	1.09	**2.61**
Pre-trained Backbone: CLIP RN50 | *Embedding dimension: 1024-d*													
2*Baseline	Acc.	58.10	40.00	61.00	74.00	58.60	15.60	83.80	85.80	58.40	23.70	55.70	55.90
	ECE	2.09	9.91	3.19	3.11	3.54	6.45	5.91	4.33	3.05	15.40	4.70	5.61
2*TPT	Acc.	60.70	41.50	62.50	74.90	61.10	17.00	84.50	87.00	59.50	28.30	58.00	57.70
	ECE	11.40	25.70	13.40	5.25	9.24	16.10	3.65	5.04	12.40	22.50	3.76	11.70
2*C-TPT	Acc.	60.20	42.20	65.20	74.70	61.00	17.00	84.10	86.90	59.70	27.80	56.50	57.75
	ECE	3.01	19.80	4.14	1.86	2.93	10.70	2.77	2.07	3.83	15.10	1.94	6.19
2*O-TPT	Acc.	58.97	41.90	65.61	74.22	60.85	16.77	83.40	86.86	58.84	28.35	56.44	57.47
	ECE	3.10	16.53	2.50	1.20	3.20	8.18	3.50	2.75	2.60	14.71	1.69	5.45
pink!20	**Acc.**	58.44	40.90	64.89	74.10	60.46	14.58	83.48	86.57	60.24	32.14	57.08	57.53
pink!20 -2***A-TPT (Ours)**	**ECE**	2.49	6.41	2.39	1.11	2.90	6.14	2.47	1.98	2.34	2.51	1.38	**2.92**

(a) Fine-grained datasets

Method	Metric	ImageNet-A	ImageNet-V2	ImageNet-R	ImageNet-S	*Average*
Pre-trained Backbone: CLIP ViT-B/16 | *Embedding dimension: 512-d*						
2*Baseline	Acc.	47.80	60.80	74.00	46.10	57.20
	ECE	8.61	3.01	3.58	4.95	5.04
2*TPT	Acc.	52.60	63.00	76.70	47.50	59.90
	ECE	16.40	11.10	4.36	16.10	12.00
2*C-TPT	Acc.	51.60	62.70	76.00	47.90	59.60
	ECE	8.16	6.23	1.54	7.35	5.82
2*O-TPT	Acc.	49.87	61.65	72.55	47.12	57.80
	ECE	7.22	3.97	1.46	6.87	4.88
pink!20	**Acc.**	50.39	60.90	74.87	46.09	58.06
pink!20 -2***A-TPT (Ours)**	**ECE**	6.45	2.96	1.39	4.87	**3.92**
Pre-trained Backbone: CLIP RN50 | *Embedding dimension: 1024-d*						
2*Baseline	Acc.	21.70	51.40	56.00	33.30	40.60
	ECE	21.30	3.33	2.07	3.15	7.46
2*TPT	Acc.	25.20	54.60	58.90	35.10	43.50
	ECE	31.00	13.10	9.18	13.70	16.70
2*C-TPT	Acc.	23.40	54.70	58.00	35.10	42.80
	ECE	25.40	8.58	4.57	9.70	12.10
2*O-TPT	Acc.	23.07	53.11	54.47	33.98	41.16
	ECE	24.56	3.87	4.47	5.85	9.69
pink!20	**Acc.**	21.66	51.48	55.78	33.37	40.57
pink!20 -2***A-TPT (Ours)**	**ECE**	21.14	3.10	3.96	3.09	**7.82**

(b) Natural distribution shift datasets

Table: Accuracy (Acc.) and Expected Calibration Error (ECE) across datasets.

Improvement over Data Augmentation

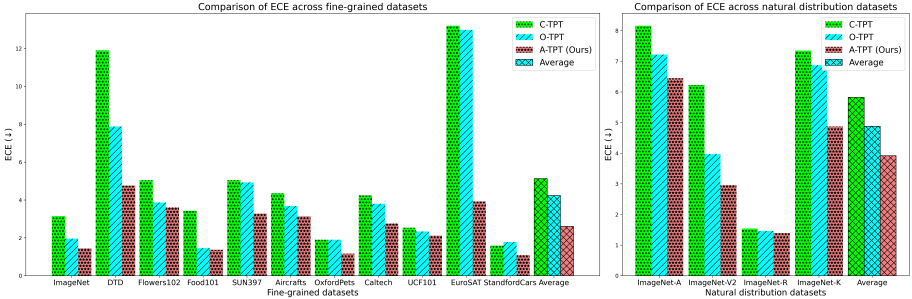
Method	Metric	DTD	Flowers102	Food101	SUN397	Aircrafts	OxfordPets	Caltech101	UCF101	EuroSAT	Standford Cars	Average
Pre-trained Backbone: CLIP ViT-B/16 | Baseline + CoOp | *Embedding dimension: 512-d*												
2*Baseline + CoOp	Acc.	43.10	67.40	83.20	63.70	18.00	89.20	93.60	66.00	40.10	63.10	63.50
	ECE	7.71	3.92	1.55	1.72	9.21	2.92	3.65	3.47	15.30	6.86	5.25
2*TPT + CoOp	Acc.	44.50	68.70	83.80	65.60	20.00	89.10	94.00	67.20	40.60	65.60	63.91
	ECE	34.80	19.90	9.66	20.80	29.60	7.40	3.65	19.90	31.30	6.63	18.36
2*TPT + CoOp + C-TPT	Acc.	45.00	69.00	83.70	65.10	19.20	89.30	93.90	66.60	40.70	63.10	63.56
	ECE	21.00	10.20	4.49	11.80	21.50	2.12	1.66	12.00	13.20	2.45	10.04
2*TPT + CoOp + O-TPT	Acc.	45.45	68.57	83.55	64.01	18.69	89.07	93.71	65.64	40.17	64.12	63.14
	ECE	16.02	6.81	3.59	7.23	16.82	1.92	0.92	9.16	13.76	2.85	7.91
pink!20	**Acc.**	43.21	68.94	83.23	65.34	20.58	90.02	93.23	69.99	40.28	65.89	64.07
pink!20 -2***TPT + CoOp + A-TPT**	**ECE**	6.33	2.91	3.12	2.63	5.51	1.06	1.08	3.78	7.85	2.04	**3.63**
Pre-trained Backbone: CLIP ViT-B/16 | Baseline + CoCoOp | *Embedding dimension: 1024-d*												
2*Baseline + CoCoOp	Acc.	44.60	68.40	84.10	63.00	24.20	88.30	91.00	67.00	44.10	64.90	64.30
	ECE	3.82	3.82	3.25	4.61	4.06	4.60	3.52	3.28	5.81	6.51	4.20
2*TPT + CoCoOp	Acc.	45.00	68.60	84.60	64.00	24.90	88.50	91.20	67.80	44.50	65.90	64.90
	ECE	6.91	4.70	1.94	3.16	6.13	2.22	2.74	3.47	9.03	5.22	4.35
2*TPT + CoCoOp + C-TPT	Acc.	44.70	69.30	84.20	63.60	24.60	88.80	91.40	67.10	44.30	64.90	64.70
	ECE	4.18	3.13	2.66	2.96	4.90	3.76	3.45	2.91	5.79	5.09	3.68
pink!20	**Acc.**	44.28	68.73	84.12	63.68	24.14	88.37	91.21	67.89	44.16	64.91	64.15
pink!20 -2***TPT + CoCoOp + A-TPT**	**ECE**	3.52	3.08	1.91	2.74	4.79	2.09	2.67	2.85	3.49	4.95	**3.22**

Table: Top-1 Accuracy (Acc.) and Expected Calibration Error (ECE) across CoOp and CoCoOp.

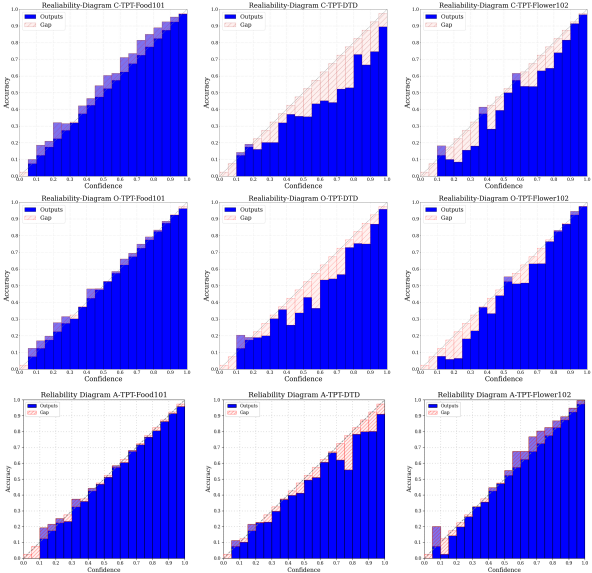
Calibration Results

Expected Calibration Error: (lower is better)

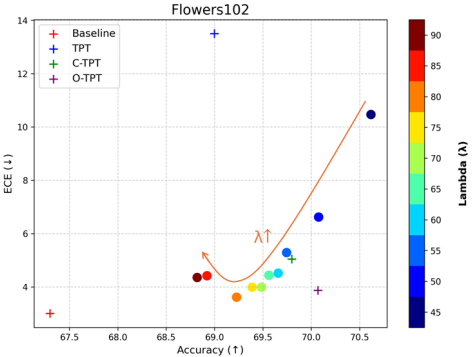
$$\mathrm{ECE} = \sum_{n=1}^{N} \frac{|B_n|}{M} \left|\mathrm{acc}\left(B_n\right) - \mathrm{conf}\left(B_n\right)\right|,$$



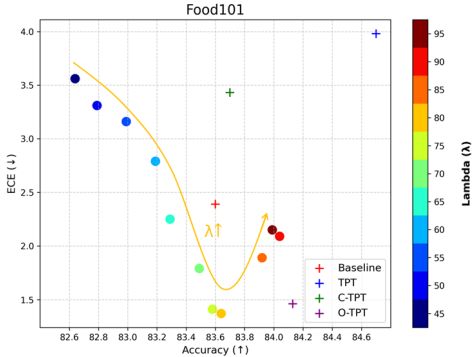
Reliability Diagrams



Pareto Front Analysis



(a) Flower 102



(b) Food 101

Conclusion

- Motivated by the Tammes' problem, we propose Angular Diversity (A-TPT) as an efficient calibration method.
- We theoretically justify that A-TPT is capable of finding optimal angular separation, thereby improving calibration.
- Extensive experiments on the benchmark datasets demonstrate that A-TPT achieves better calibration performance among the compared TPT methods on various backbones and different datasets.

ArXiv: `https://arxiv.org/abs/2510.26441`
Code: `https://github.com/MB-Shihab-Aaqil-Ahamed/A-TPT/`