Modular Embedding Recomposition for Incremental Learning

Aniello Panariello, Emanuele Frascaroli, Pietro Buzzega, Lorenzo Bonicelli, Angelo Porrello, Simone Calderara
{name.surname}@unimore.it

AImage Lab — UNIMORE — UNIVERSITÀ DEGLI STUDI DI MODENA E REGGIO EMILIA

BMVC 2025

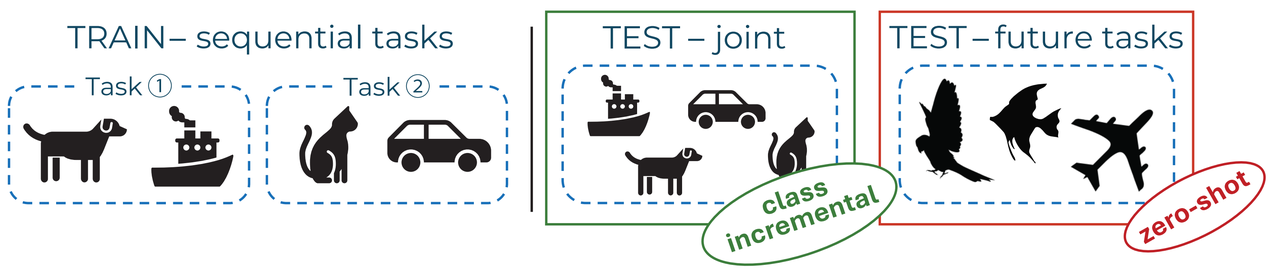
TL;DR

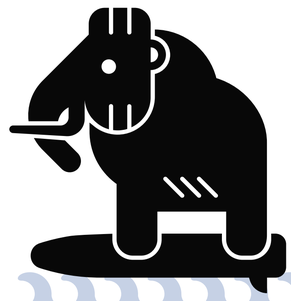
MoDER improves the zero-shot and continual learning abilities of Vision-Language Models. Instead of just preserving zero-shot performance when learning new tasks, it enhances it through **modular compositionality**.

- MoDER trains **textual expert modules**, each specialized on a single class, using parameter-efficient fine-tuning.
- These experts are stored in a foundational hub, a library of **reusable knowledge**.
- When facing new classes, MoDER **composes relevant experts** from the hub to generate new prototypes on the fly.

Motivation

Vision-Language Models excel at zero-shot recognition, but they forget when fine-tuned and cannot adapt to new tasks without losing generality. We need a way to continually learn new knowledge while enhancing, not degrading, their zero-shot ability.



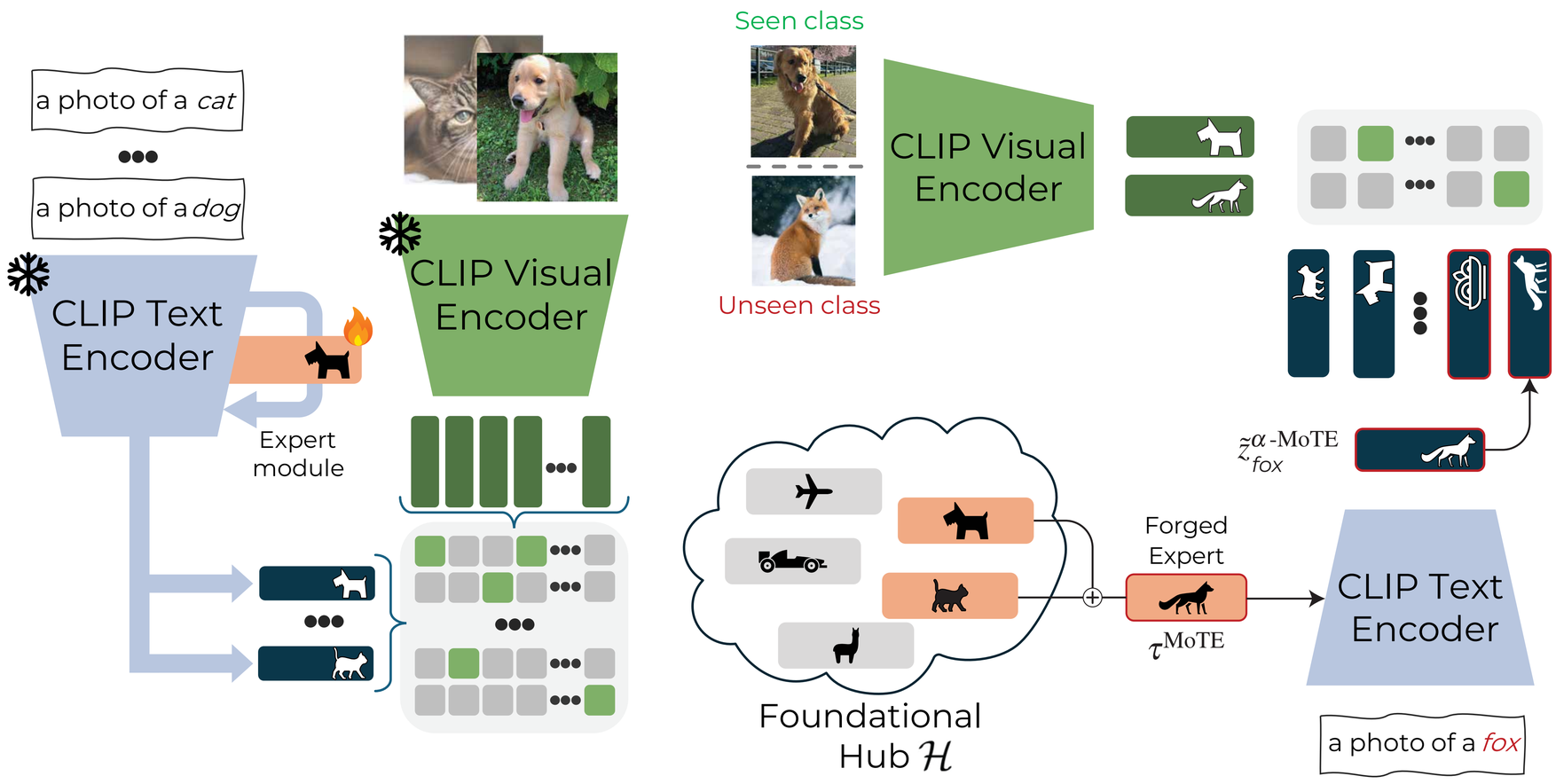


MAMMOTH

A PyTorch benchmark library for Continual Learning with 20+ datasets and 50+ methods.
aimagelab/mammoth

Phase 1 - Textual Alignment

Phase 2 - Forging the Unseen Classes Experts



MoDER — Training Phase

for each dataset $\mathcal{D}_{t_c}$, $t_c \in 1, \ldots, T$ **do**
  Fit g_{t_c} on $\{E^{\text{vis}}(x) \mid x \in \mathcal{D}_{t_c}\}$
  $\mathcal{D}_{\text{SYN}} = \{z^{\text{vis}} \sim g_i, \forall\, i \in 1, \ldots, t_c\}$
  # Textual Alignment
  for it: $= 1, \ldots$ **do**
    Sample random prompt $p_i \; \forall\, i \in \mathcal{Y}$
    $z_i^{\text{txt}} = E^{\text{txt}}(p_i; \theta_0 + \tau_i)$
    Sample a batch from $\mathcal{D}_{\text{SYN}}$
    Update expert τ_i with $\mathcal{L}_\sigma$

MoDER — Forging

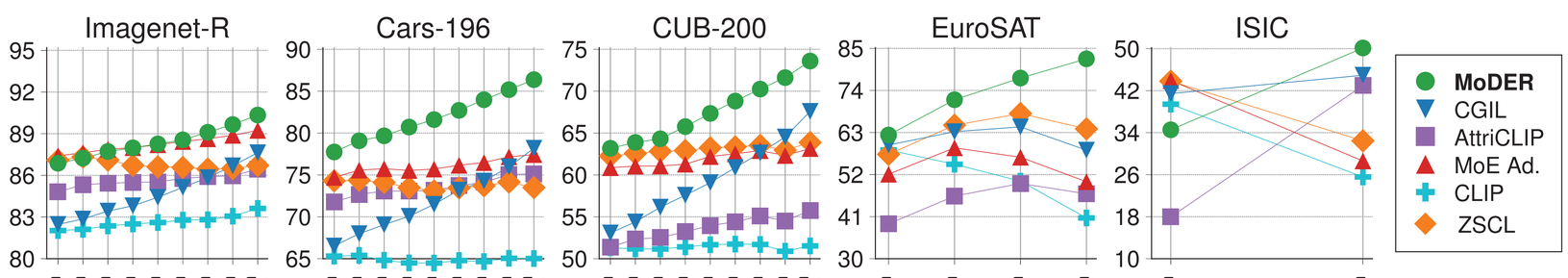
for each class i in $\mathcal{Y}_{\text{seen}}$ **do**
  $\mathcal{Z}[i] \leftarrow E^{\text{txt}}(p_i; \theta_0 + \alpha\tau_i)$
  # Mixture of Textual Experts
  for each class j in $\mathcal{Y}_{\text{unseen}}$ **do**
    $\text{top}_K(p_j) \leftarrow$ top-K experts from $\mathcal{H}$
    $\mathcal{Z}[j] \leftarrow$ Combine their weights
    $z_j^{\alpha\text{-MoTE}} \leftarrow E^{\text{txt}}(p_j; \theta_0 + \alpha\boldsymbol{\tau}_j^{\text{MoTE}})$
$z^{\text{vis}} \leftarrow E^{\text{vis}}(x_{\text{new}})$ ▷ ***# classification***
$y_{\text{pred}} \leftarrow \arg\max_{y \in \mathcal{Y}} \langle z^{\text{vis}}, \mathcal{Z}_y \rangle$

Class Incremental Transfer

Model	IN-R	Cars	CUB	ESAT	ISIC	Avg.
CLIP	82.1	66.2	50.9	55.0	22.4	55.3
AttriCLIP	85.7	74.0	54.1	59.7	24.1	61.0
MoE Ad.	88.2	75.8	61.7	55.8	21.1	60.5
ZSCL	85.3	72.5	62.8	69.7	25.3	63.1
CGIL	86.7	78.8	66.3	71.5	48.2	70.3
MoDER (LoRA)	**89.7**	**87.0**	**73.2**	74.0	**52.8**	**75.3**
MoDER (VeRA)	89.6	86.9	72.9	**74.7**	51.7	75.2

To evaluate the zero-shot capabilities of our approach we measure the CI-Transfer, which evaluates the accuracy on unseen tasks.

$$\frac{1}{T-1}\sum_{t=1}^{T-1}\left(\frac{1}{T-t}\sum_{i=t+1}^{T} A_t^i\right)$$



Class Incremental Learning

Model	IN-R	Cars	CUB	ESAT	ISIC	Avg.
CLIP	81.9	65.0	50.5	53.3	26.6	55.5
CODA-P.	78.9	45.2	72.2	63.7	47.4	61.5
AttriCLIP	87.4	75.6	58.3	72.3	28.3	64.4
SLCA	85.5	73.5	**87.8**	93.6	63.8	80.8
TMC	63.2	39.9	63.4	65.0	48.9	56.1
Inf-LoRA	84.4	58.0	80.4	79.7	56.1	71.7
MoE Ad.	**90.7**	77.8	65.0	80.6	34.5	69.7
STAR-P.	89.2	86.5	85.2	94.2	67.4	84.5
ZSCL	89.1	77.7	62.4	79.1	34.1	68.5
CGIL	89.4	89.3	83.1	96.2	73.0	86.2
MoDER (LoRA)	89.7	**90.1**	83.7	**96.4**	76.3	**87.2**
MoDER (VeRA)	89.4	89.6	82.7	96.3	**76.4**	86.9

Experimental Results

Our method, MoDER, achieves state-of-the-art performance across all benchmarks.

- **Class-Incremental Transfer**: Strongest zero-shot transfer to unseen classes.
- **Class-Incremental Learning**: Best overall retention and adaptation balance.
- **MTIL**: Improves zero-shot accuracy while minimizing forgetting across domains. MoDER effectively reuses textual experts to enhance generalization and maintain efficiency, outperforming all prior CLIP-based approaches.

Multi Task Incremental Learning (MTIL)

Model	Transf.	Δ	Avg.	Δ	Last	Δ
CLIP	69.4	0.0	65.3	0.0	65.3	0.0
Cont. FT	44.6	-24.8	55.9	-9.4	77.3	+12.0
LwF	58.9	-10.5	64.7	-0.6	74.6	+9.3
Wise-FT	52.3	-17.1	60.7	-4.6	77.7	+12.4
ZSCL	68.1	-1.3	75.4	+10.1	83.6	+18.3
MoE Ad.	68.9	-0.5	76.7	+11.4	85.0	+19.7
MoDER	**69.7**	+0.3	**76.9**	+11.6	**85.8**	+20.5

	Train. Params	GPU (MiB)
LwF	149.6 M (×7.1)	32 172 (×6.7)
ZSCL	149.6 M (×7.1)	26 290 (×5.5)
MoE Ad.	59.8 M (×2.9)	22 358 (×4.7)
MoDER	20.9 M (×1)	4793 (×1)