Incremental Learning with Repetition via Pseudo-Feature Projection

28th Computer Vision Winter Workshop 2025, Graz

Benedikt Tscheschner (btscheschner@know-center.at)
Eduardo Veas (eveas@know-center.at)
Marc Masana (mmasana@tugraz.at)

TU Graz

SCIENCE
PASSION
TECHNOLOGY

KNOW Center

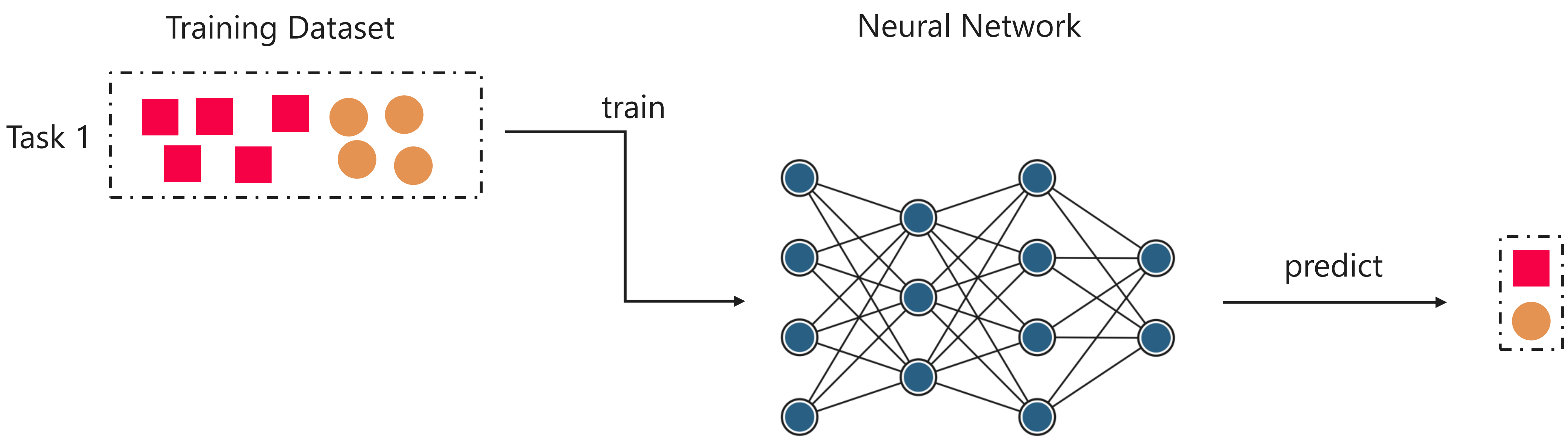
INSTITUTE OF VISUAL COMPUTING

Motivation Incremental Learning

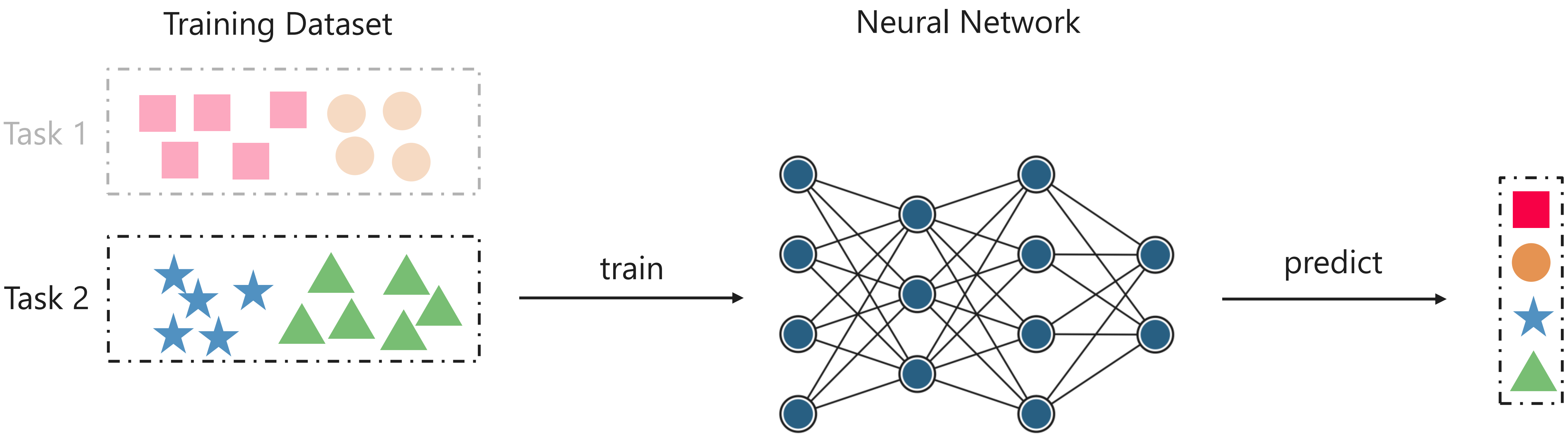
- **Goal**: Learn Model over time as information can be constantly changing
- Training models from scratch is not cheap
- Restriction in:
  - Training Time / Frequent Updates
  - Computational Cost
  - Data Access (Data Privacy / Storage)

© generated by Fotor.com

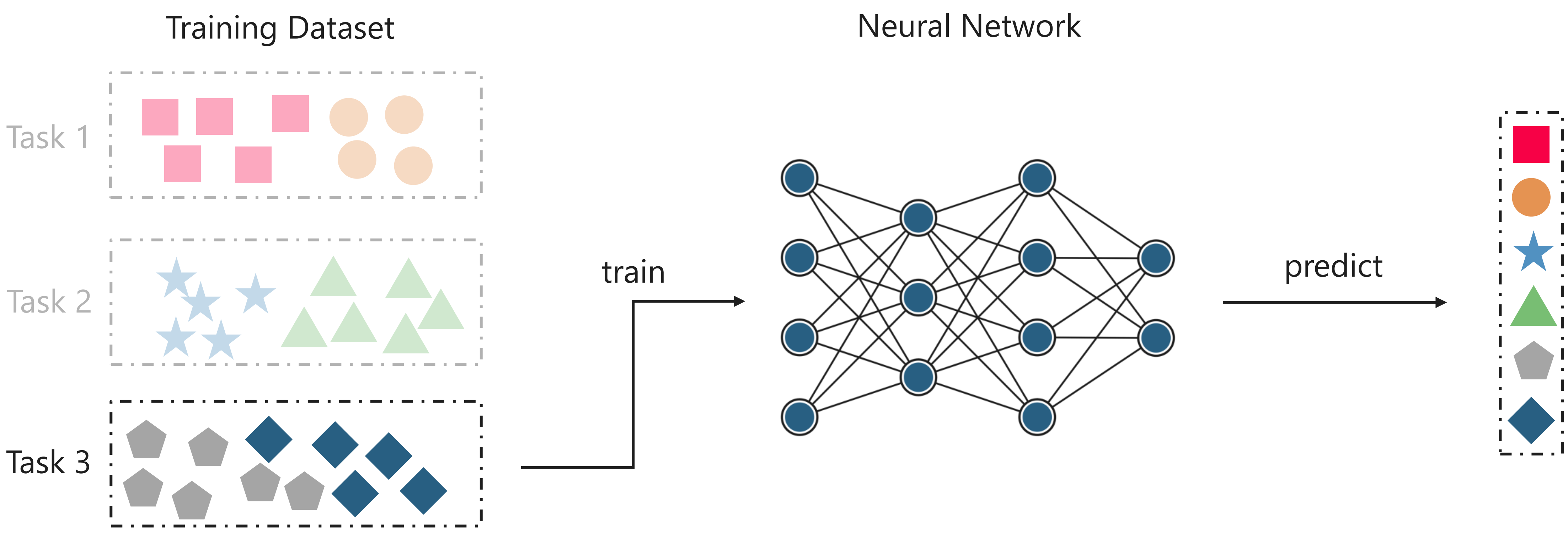
Class Incremental Learning



Class Incremental Learning



Class Incremental Learning



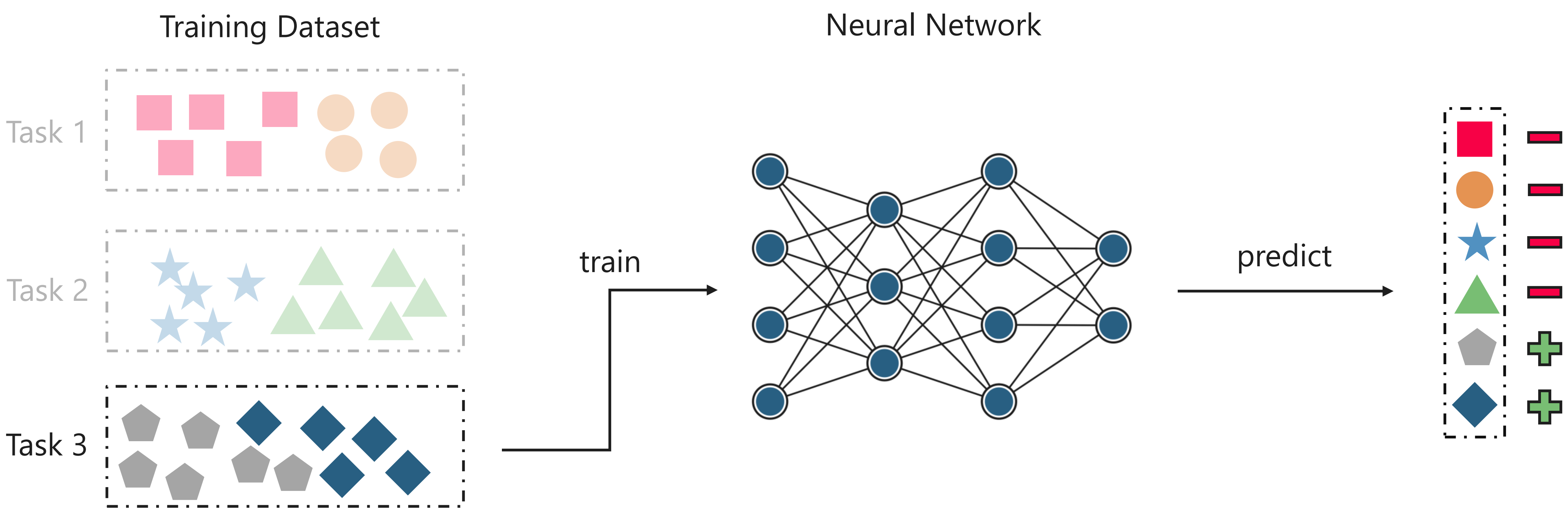
- No access to classes of previous tasks
- Evaluated over all known classes

M. Masana et. al., “Class-incremental learning: survey and performance evaluation on image classification”, TPAMI (2022)

Catastrophic Forgetting

Low Performance

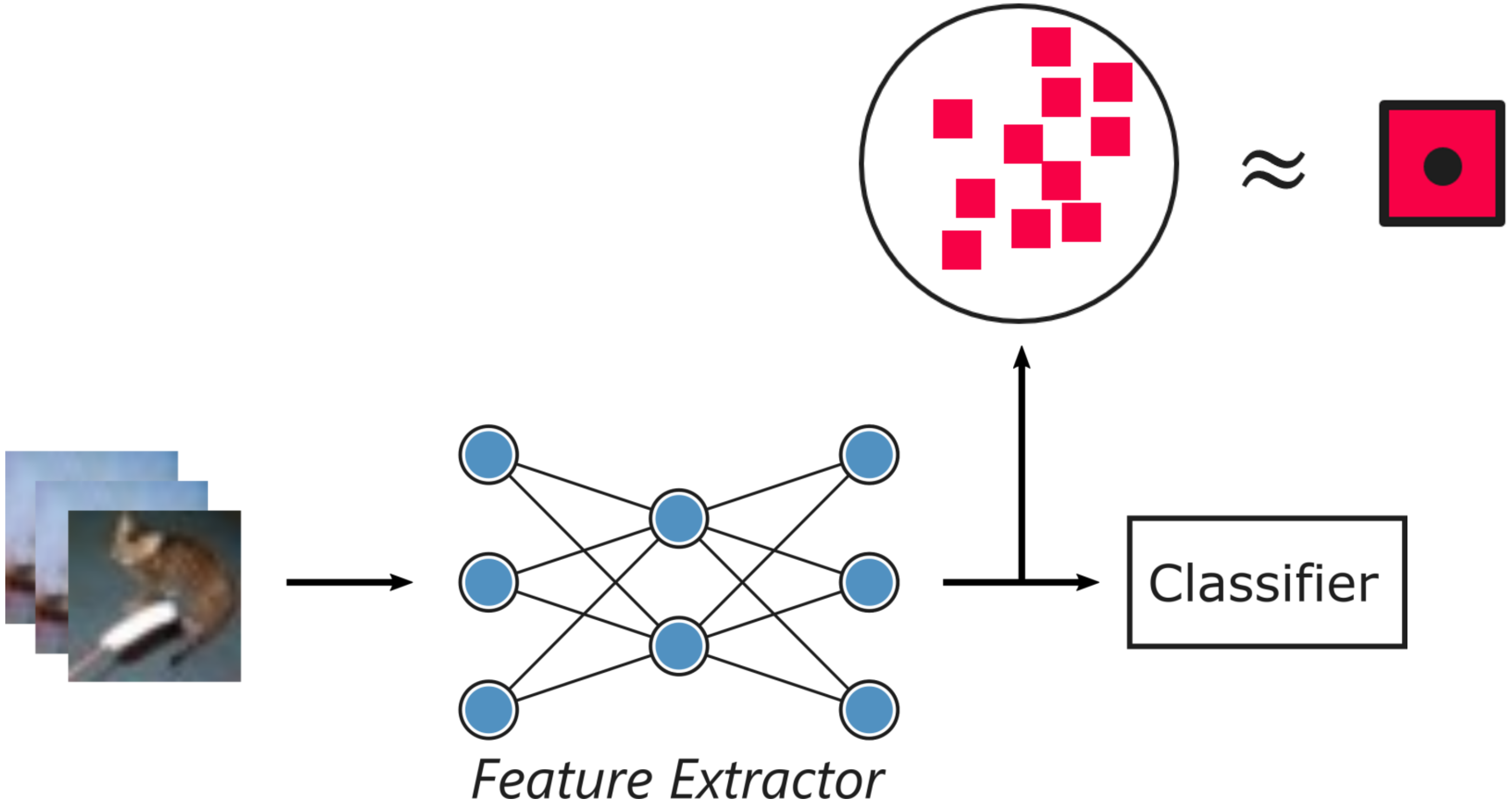
High Performance



M. Mermillod, et. al "The stability-plasticity dilemma: Investigating the continuum from catastrophic forgetting to age-limited learning effects," Frontiers in psychology, 2013.

Class Prototypes

- **Class Prototype (■)** represent the distribution of a class at a **layer**



Class Prototypes

- Two ways of **rehearsal** of **unavailable classes**:

  1. **Sample distribution**[1]
  2. **Feature translation**[2]

$$\widehat{F_c} = f(x_i;\theta) - \mu_{y_i} + \mu_c$$

- Requires **frozen** or **regularized** feature extractor

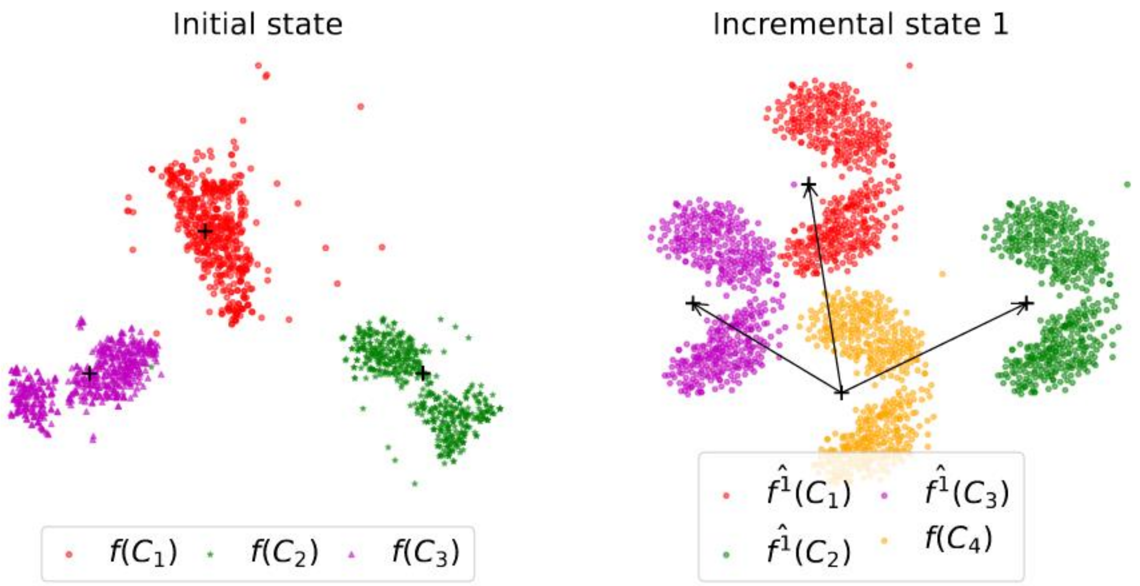


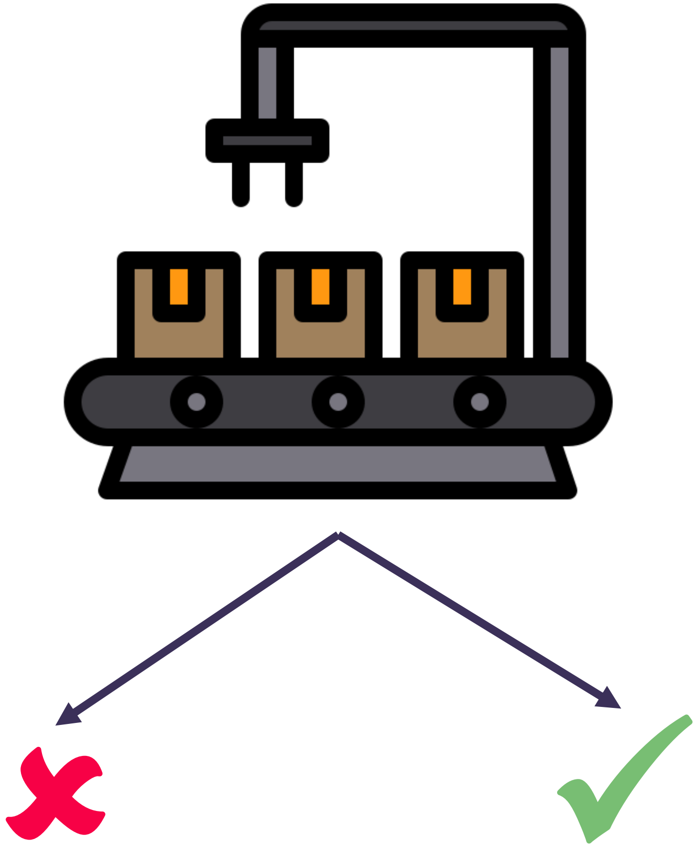
Figure from [2]

[1] F. Zhu et. al., "Prototype Augmentation and Self-Supervision for Incremental Learning", CVPR (2021)
[2] G. Petit et. al., "FeTrIL: Feature Translation for Exemplar-Free Class-Incremental Learning", WACV (2023)

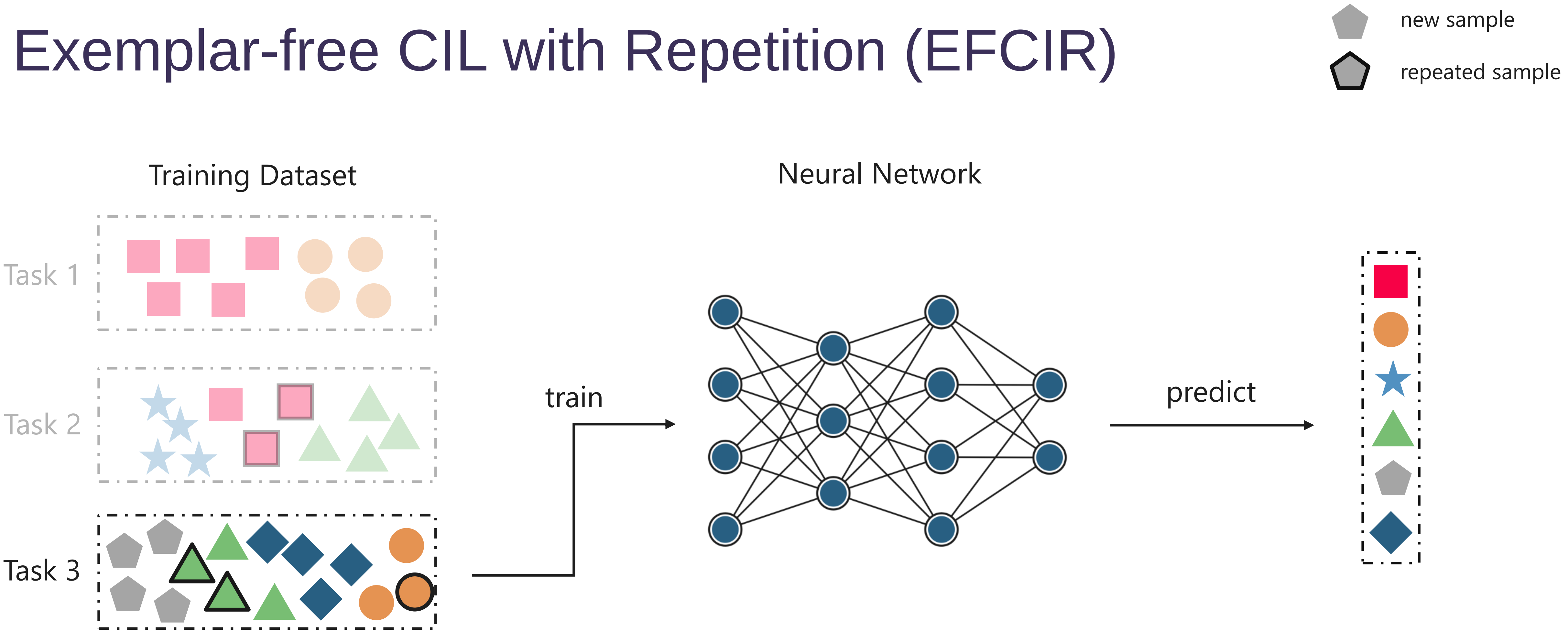
Why Repetition?

1. *If we incrementally add classes to a system is the complete distribution for training known / available?*

2. *Is it reasonable that classes never repeat in a sequence of tasks?*

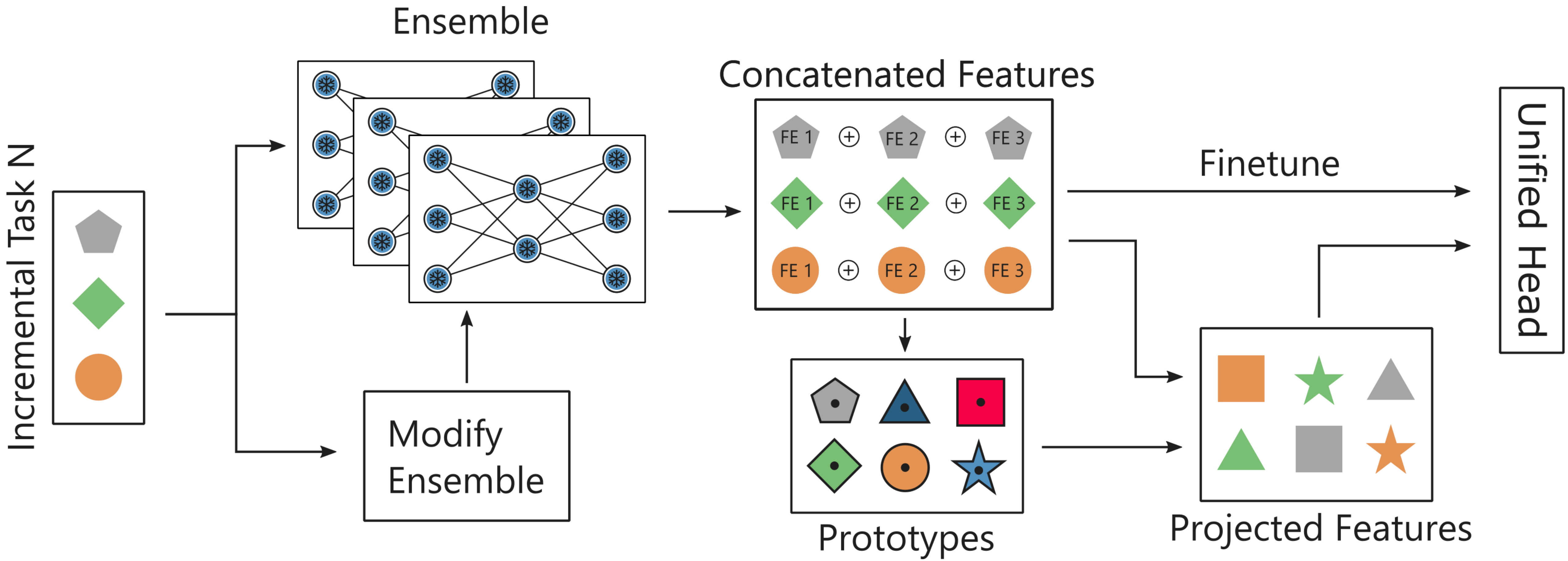


Exemplar-free CIL with Repetition (EFCIR)



- Repetition is modelled in the scenario creation
- Complete training data unavailable in any task

Method Overview

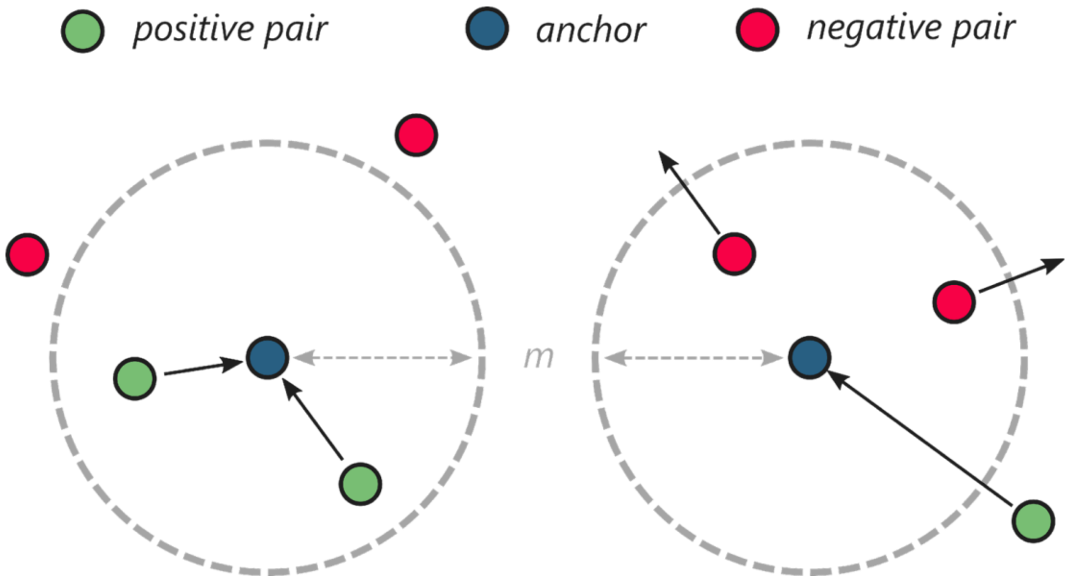


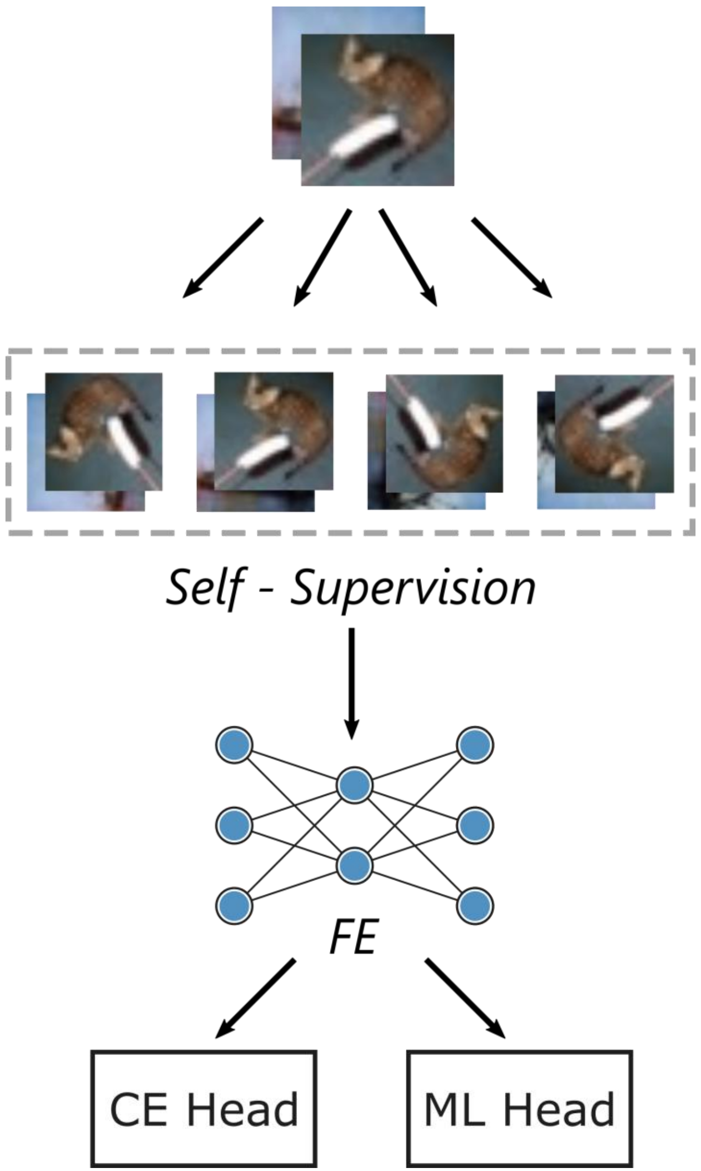
Ensemble Modification

- **Goals:**
  - Identify **beneficial** incremental tasks
  - **Flexibility** in network **architecture & size**
  - Limit the **growth** of the ensemble to a **predefined budget**
- Two baseline heuristics:
  1. **Class Diversity:** maximize number of represented classes
  2. **Error Rate:** on the incremental data (before training)

Feature Extractor Training

Metric Learning[1] **regularizes shape** of class prototypes



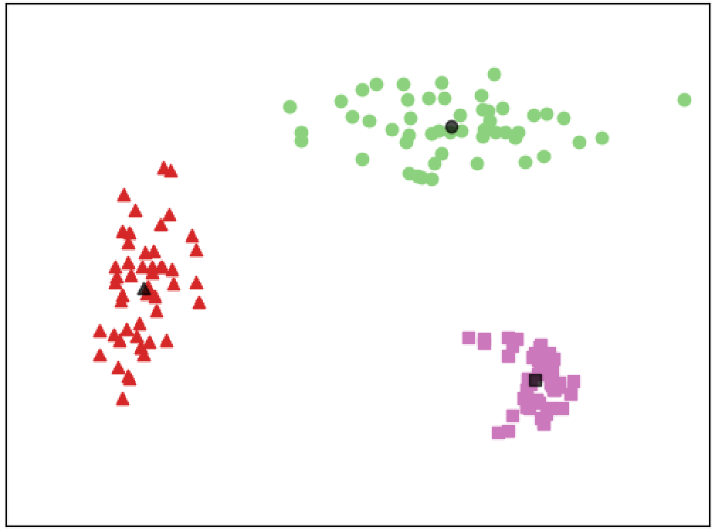


[1] R. Hadsell et. al. “Dimensionality reduction by learning an invariant mapping.” in IEEE CVPR 2006

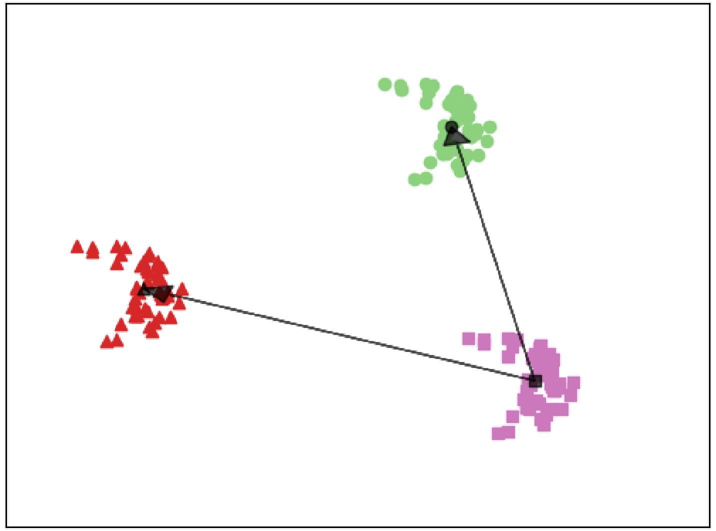
Pseudo-feature Projection (PFP)

- Extend **feature translation**[1] with **standard deviation**

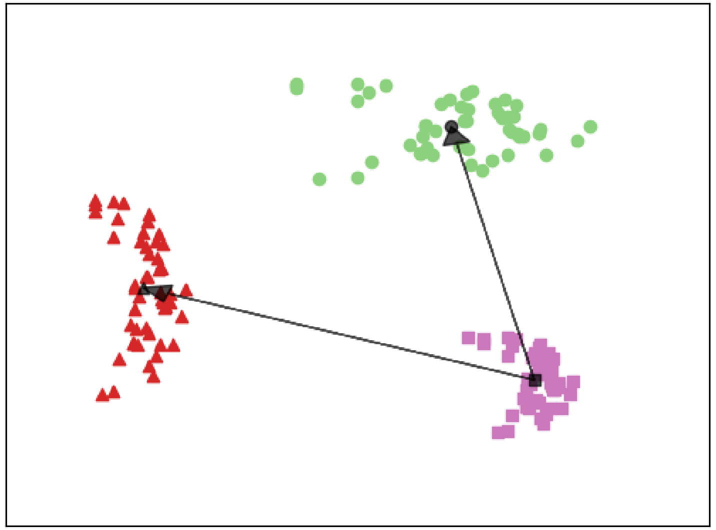
$$\widehat{F}_c = \frac{f(x_i;\theta) - \mu_{y_i}}{\boldsymbol{\sigma}_{\boldsymbol{y_i}}} \cdot \boldsymbol{\sigma}_{\boldsymbol{c}} + \mu_c$$



True Distribution



Feature Translation



Pseudo Feature Projection

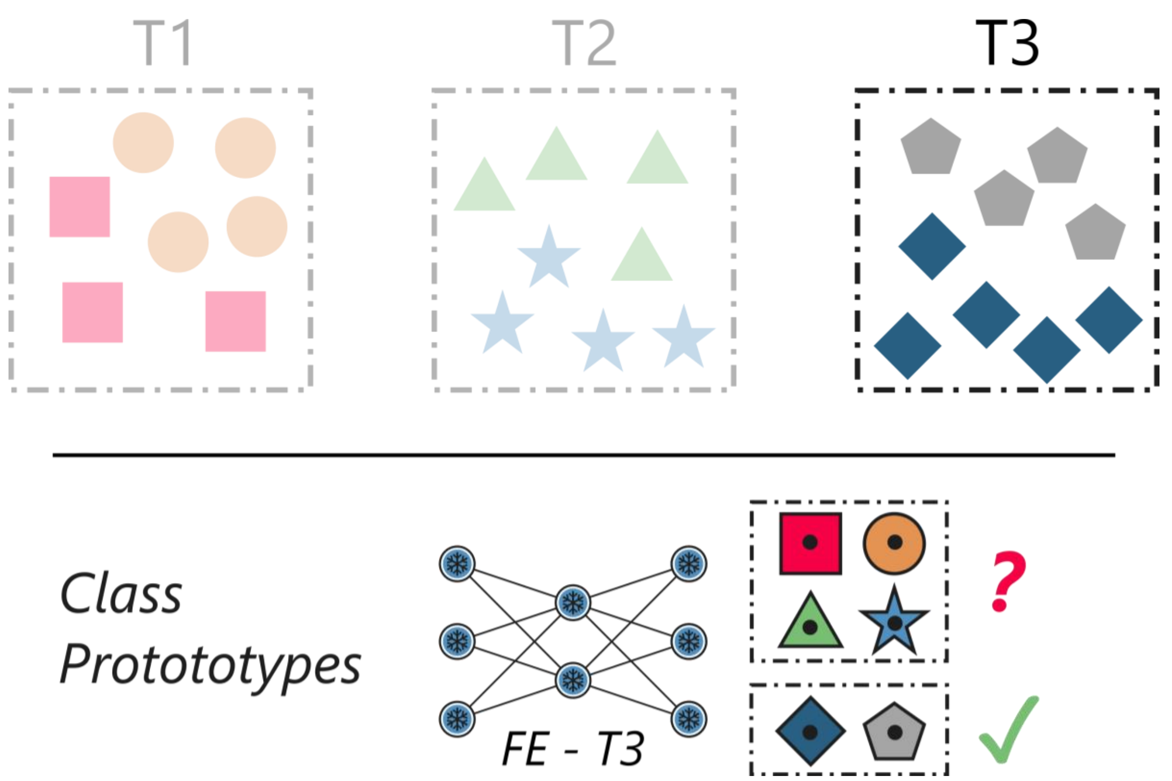
[1] G. Petit et. al. “FeTrIL: Feature Translation for Exemplar-Free Class-Incremental Learning”, WACV 2023.

PFP - Prototype Estimation

- Past class prototypes $\boldsymbol{c_{old}}$ **incomplete** when ensemble is modified

$$\boldsymbol{\mu_c} = (\mu_{c,f_1}, \dots, \mu_{c,f_N})$$
$$\boldsymbol{\sigma_c} = (\sigma_{c,f_1}, \dots, \sigma_{c,f_N})$$

- **Estimate** for f_{new}
  - $\mu_{c_{old}, f_{new}} = f_{new}(x_i)$
  - $\sigma_{c_{old}, f_{new}} = \mathbf{1}$
  - Artificial sample placed near $\boldsymbol{x_i}$



Scenario (a) – Class Incremental Learning

- Classic CIL Scenario
  - No Repetition
  - 50 initial classes
  - 10 inc. tasks with 5 classes
  - CIFAR-100

(a) CIL 50/10

	Method	**Avg. $A \uparrow$**
Regularization	FT	14.2 ± 1.0
	FZ	52.6 ± 1.4
	EWC	45.9 ± 2.9
	MAS	45.9 ± 2.9
	LwF	47.9 ± 1.8
Prototype Sampling	PASS	62.1 ± 1.9
	PRAKA	$\mathbf{63.1 \pm 2.5}$ ●
	IL2A	54.2 ± 1.4
	SSRE	53.0 ± 2.7
Feature Translate	FeTrIL	61.4 ± 0.4
	$Horde_m$	$\mathbf{62.9 \pm 1.2}$ ●
	$Horde_c$	$\mathbf{62.9 \pm 1.2}$ ●

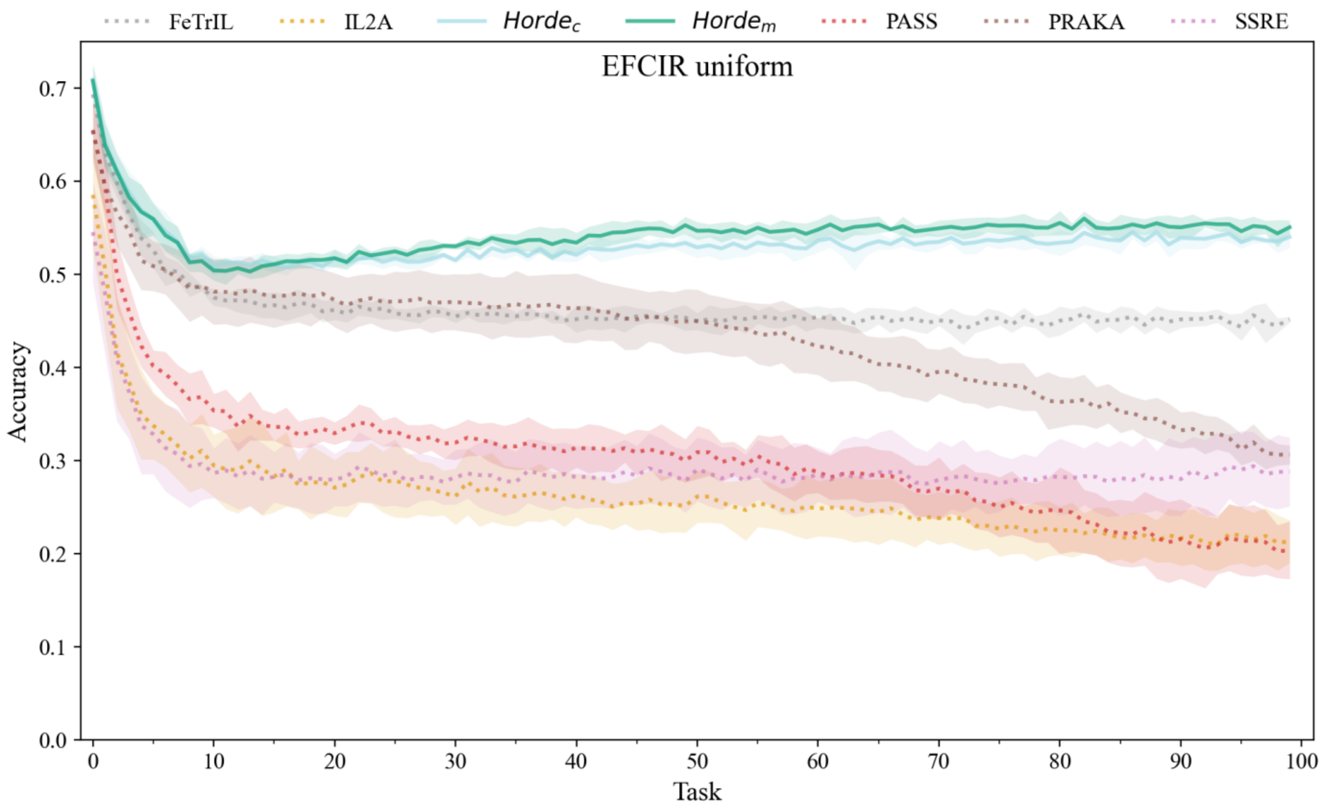
Scenario (b) – with Repetition

- Adapted CIL with Repetition
  - 99 small incremental tasks
  - 15% random repetition chance
  - 50% of training data for initial training

		(a) CIL 50/10	(b) EFCIR-U 50/100
	Method	**Avg. $A \uparrow$**	**Avg. $A \uparrow$**
Regularization	FT	14.2 ± 1.0	36.2 ± 2.1
	FZ	52.6 ± 1.4	40.2 ± 3.9
	EWC	45.9 ± 2.9	47.7 ± 3.2
	MAS	45.9 ± 2.9	$\mathbf{49.3} \pm \mathbf{2.6}$ ●
	LwF	47.9 ± 1.8	45.7 ± 1.9
Prototype Sampling	PASS	62.1 ± 1.9	35.3 ± 2.1
	PRAKA	$\mathbf{63.1} \pm \mathbf{2.5}$ ●	43.1 ± 2.1
	IL2A	54.2 ± 1.4	26.3 ± 3.0
	SSRE	53.0 ± 2.7	29.2 ± 3.5
Feature Translate	FeTrIL	61.4 ± 0.4	46.5 ± 0.7
	$Horde_m$	$\mathbf{62.9} \pm \mathbf{1.2}$ ●	$\mathbf{54.4} \pm \mathbf{0.7}$ ●
	$Horde_c$	$\mathbf{62.9} \pm \mathbf{1.2}$ ●	$\mathbf{53.4} \pm \mathbf{0.7}$ ●

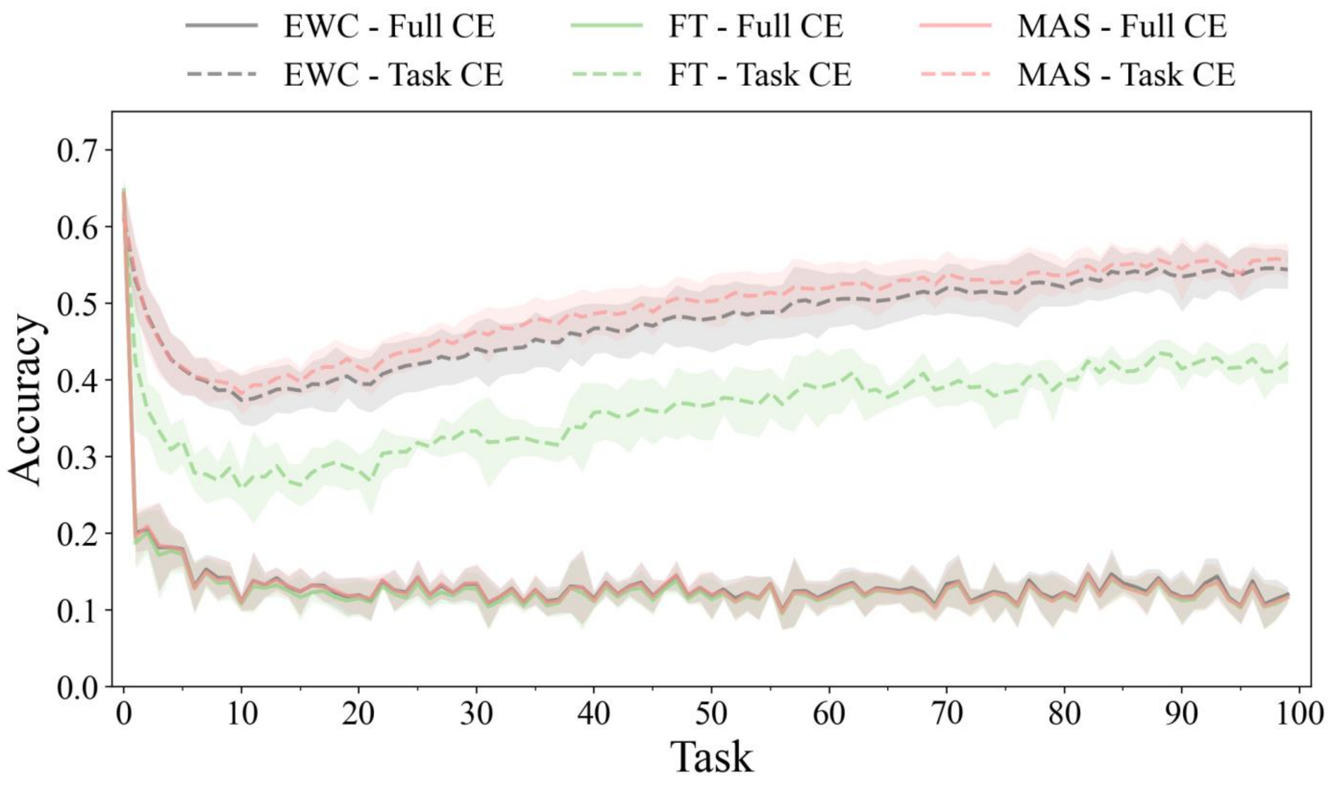
Scenario (b) – Class Incremental with Repetition

- In the **initial 15 tasks** new classes are added → test set increases
- **Dotted Lines** (Prototype Sampling) **degrade** over time in accuracy
- **Frozen FE** with feature translation stable



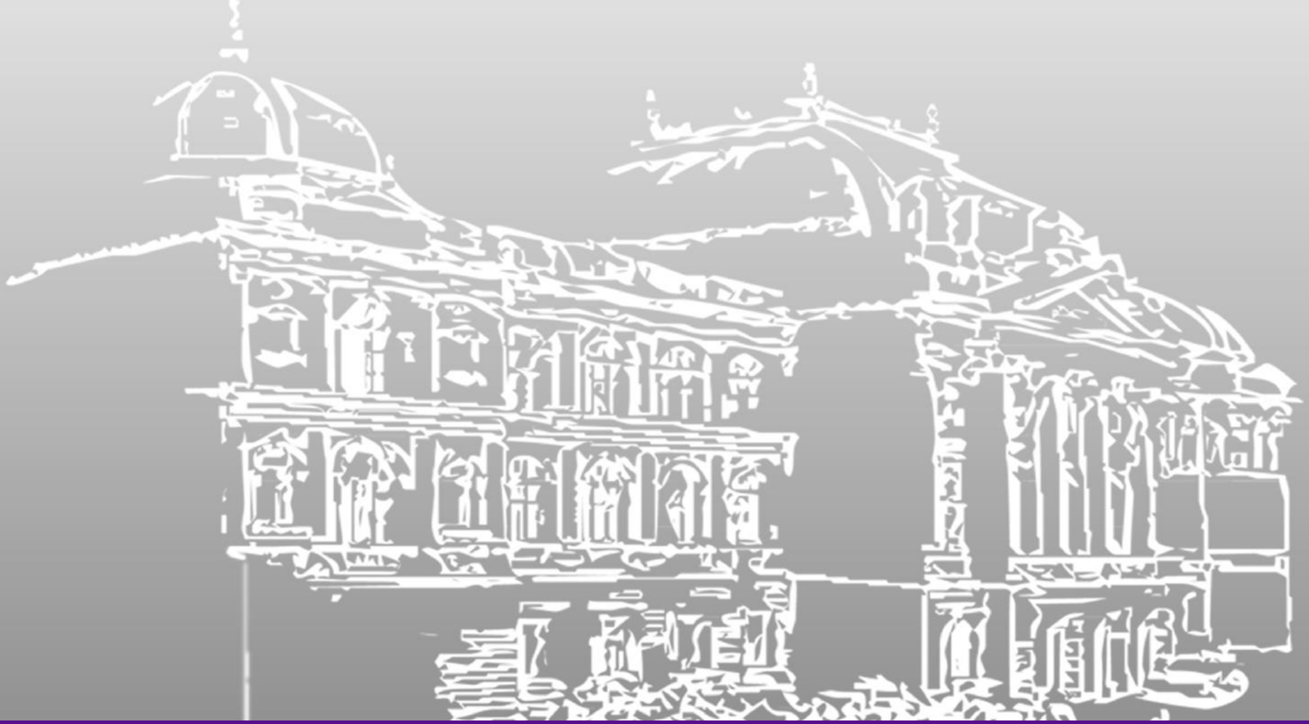
Scenario (b) – Class Incremental with Repetition

- FT, EWC, MAS **benefit** from repetition:
  - Accuracy gains over task sequence
  - **Task-aware** like training
  - Convergence over time?
    - “Limited” new samples for longer sequences



Summary

- Repetition has a significant impact on Incremental Learning
  - **complete class distribution unavailable** at any single task
  - Weight-regularized methods **benefit**
  - Class Prototype usage challenging
- Evaluate resiliency against repetition frequency bias → similar results



Thank you!

Code available soon: https://github.com/Tsebeb/cvww_cir_horde