Prototype Augmented Hypernetworks for Continual Learning

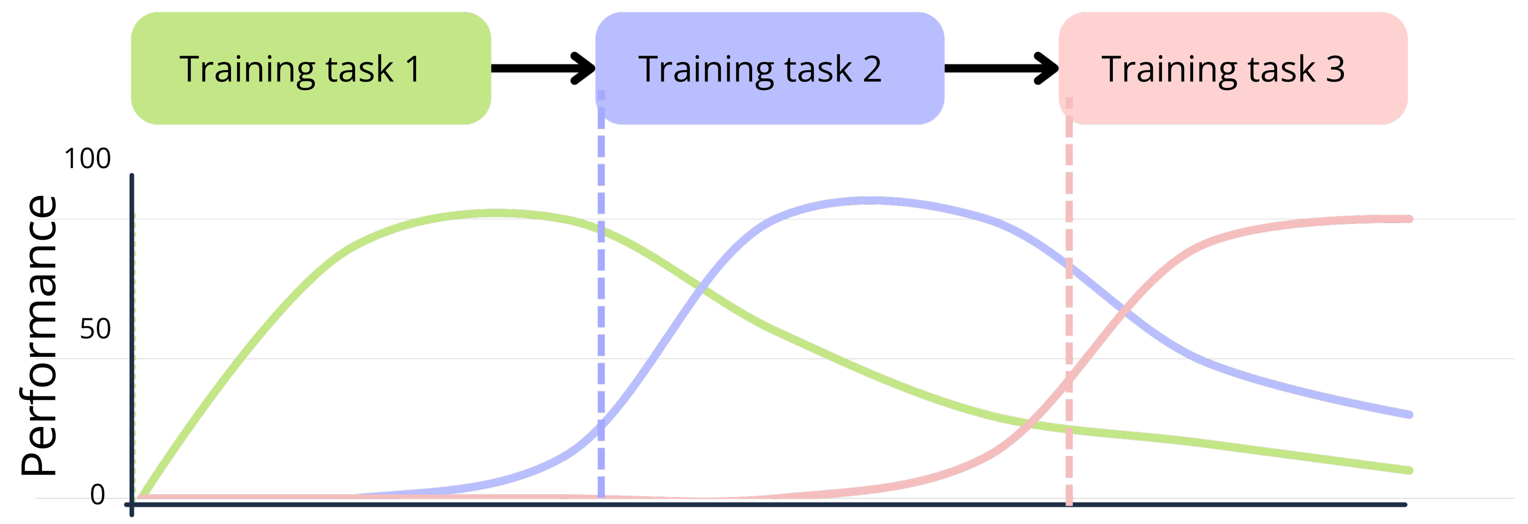
Neil De La Fuente*, Maria Pilligua*, Daniel Vidal, Albin Soutif, Cecilia Curreli, Daniel Cremers, Andrey Barsky

Computer Vision Center

TLDR: PAH trains one hypernet that turns learnable class prototypes into task head parameters on the fly, achieving SOTA continual learning without storing per-task heads.

Continual Task Learning

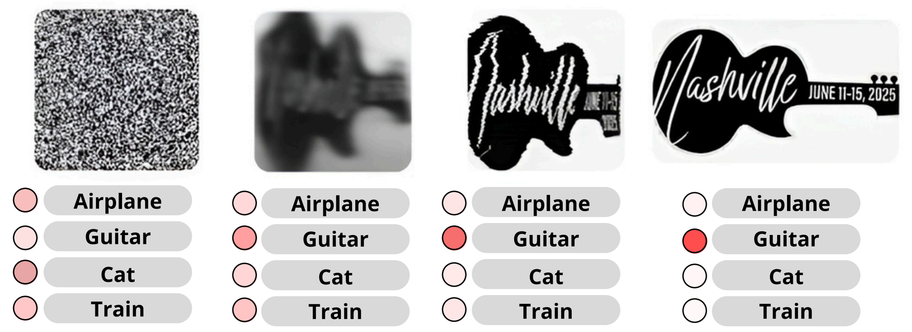
In real-world scenarios, tasks often arrive **sequentially**. Neural networks often **forget** previous tasks when learning new ones **leading to catastrophic forgetting.**



Contributions

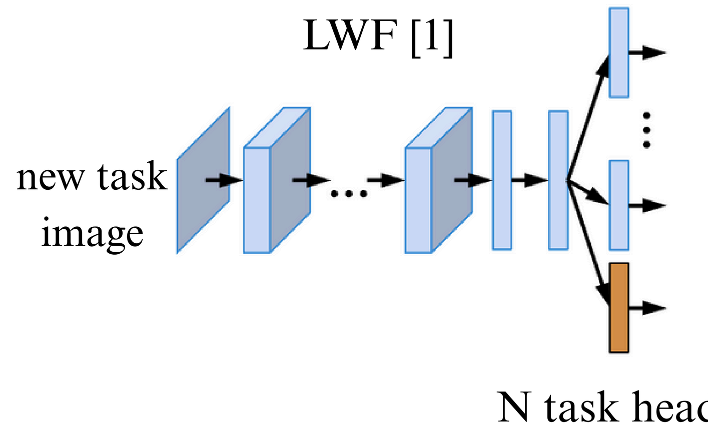
We adress catastrophic forgetting introducing:

- **Prototypes:** small, learnable feature maps (e.g. 10×10 tensors) that summarize the key visual characteristics of each class within a task.

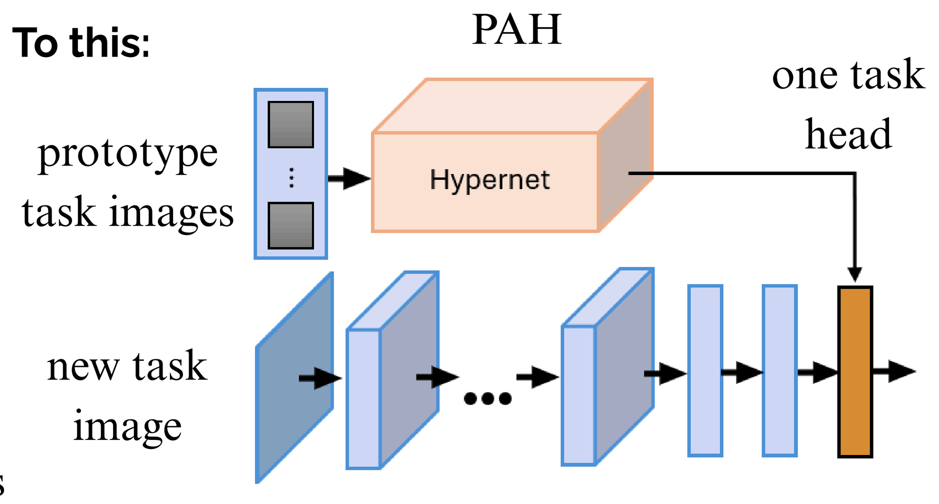


- **Hypernetworks:** No need of storing k classifier heads for k task. Just learn a single unified hypernet that generates the classifier head parameters on demand.

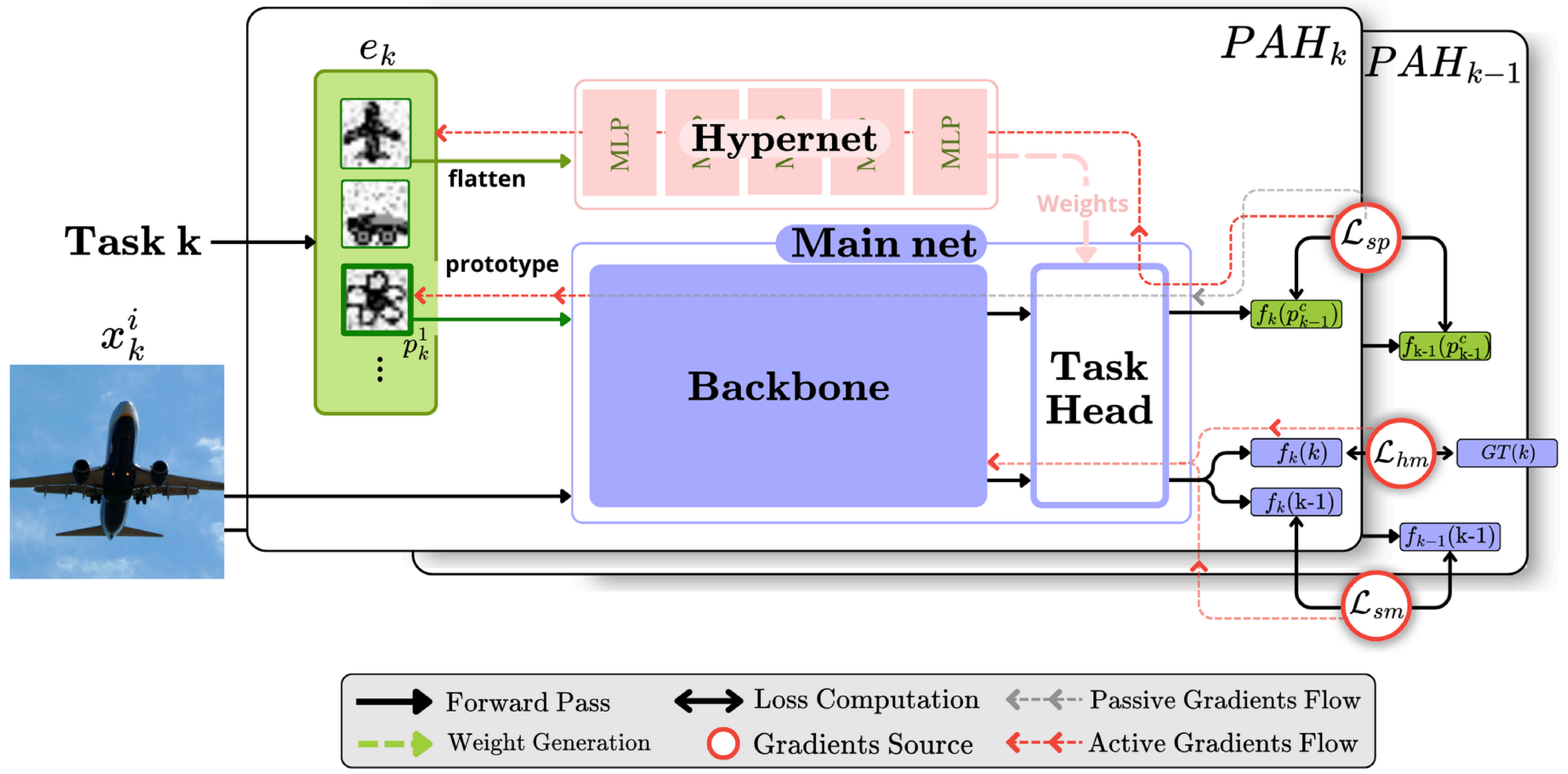
From this:



To this:



Methodology

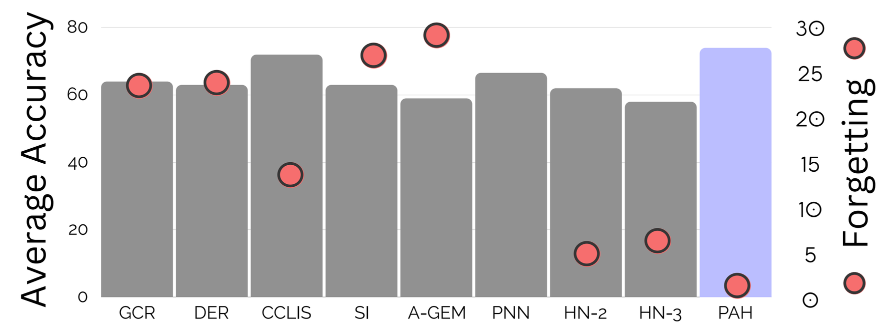


To learn a new task k:

1. **Freeze the current model** before task k starts to use as a reference for old skills.
2. Create a **learnable prototype for each** new **class** and join them into a single vector embedding that represents the task.
3. **Pass the embedding through the hypernet** to generate the task's classifier head weights on the fly; no heads stored.
4. Train on task k with cross-entropy and a distillation loss to **stay close to the frozen model.**
5. **Adjust earlier prototypes** so they still match the feature space.

Results

PAH showcases an increase in average accuracy thanks to an incredibly low forgetting measure between tasks.



Buffer	Methods	Split-CIFAR100	
		ACCURACY ↑	*FORGETTING* ↓
200	GCR	64.24±0.83	24.12±1.17
200	DER	63.09±1.09	25.98±1.55
200	CCLIS	72.93±0.46	14.17±0.20
—	SI	63.58±0.37	27.98±0.34
200-250	A-GEM	59.81±1.07	30.08±0.91
—	PNN [2]	66.58±1.00	—
200-400	HN-2	62.80±1.60	4.10±0.50
200-400	HN-3	58.80±1.00	7.40±0.90
—	*PAH* (Ours)	**74.46±0.08**	**1.71±0.02**

Losses

Classification Loss

$$L_{hm} = -\sum_{c=1}^{C} y_c \log \hat{y}_c$$

Cross-entropy loss over the real input images to enforce correct classification in new task

Previous Tasks Consistency Loss

Avoid forgetting by forcing the model to perform well in previous tasks

$$L_{sm} = \frac{1}{k\text{-}1} \sum_{j=1}^{k-1} \mathrm{KL}(f_{k-1}(x_k \mid j) \parallel f_k(x_k \mid j))$$

Prototype Alignment Loss

$$L_{sp} = \frac{1}{k\text{-}1} \sum_{j=1}^{k-1} \sum_{c=1}^{C} \mathrm{KL}(f_{k-1}(p_j^c \mid j) \parallel f_k(p_j^c \mid j))$$

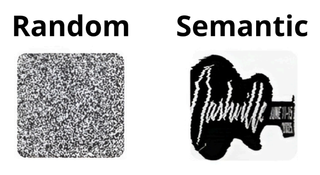
Align prototypes to evolving backbone and hypernet

Ablations

Prototype Size

	5x5	10x10	16x16	20x20
Accuracy	73.72	**74.46**	73.52	71.95
Forgetting	1.89	**1.71**	1.76	2.66

Prototype Initialization



	Random	Semantic
Accuracy	72.54	**74.46**
Forgetting	2.32	**1.71**

Summary

PAH **reduces catastrophic forgetting** by training one hypernetwork that receives a small set of learnable task prototypes and produces the task's classifier weights on the fly. PAH **outperforms** replay, regularization, and earlier hypernetwork **baselines without storing task heads or replay data**. PAH opens a clear path towards scalable and forgetting-free Continual Learning.

References

[1] Li, Z., & Hoiem, D. (2017). Learning without forgetting. IEEE transactions on pattern analysis and machine intelligence, 40(12), 2935-2947.
[2] Von Oswald, J., Henning, C., Grewe, B. F., & Sacramento, J. (2019). Continual learning with hypernetworks. arXiv preprint arXiv:1906.00695.