LEARNING CONTINUOUSLY WITHOUT FORGETTING FOR CONTINUAL SEMANTIC SEGMENTATION

CVPR 2021

Arthur Douillard
Yifu Chen
Arnaud Dapogny
Matthieu Cord

Machine Learning & Deep Learning for Information Access

The Team

The Team

Arthur Douillard
Sorbonne Université
Heuritech

Yifu Chen
Sorbonne Université

Arnaud Dapogny
Datakalab

Matthieu Cord
Sorbonne Université
Valeo.ai

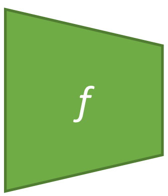
What is Continual Learning?

What

Data **independent and identically distributed** (iid) assumption

What

Data **independent and identically distributed** (iid) assumption



Retrain from scratch

Evaluate on a fixed test set

...

What

Retraining everytime is not always possible:

- **Slow** → companies with ever-growing datasets
- **Privacy** → data is only available for a short time
- **Memory limitation** → poor robot in the wild doesn't have peta of disk storage

What

Real world data is rarely **independent and identically distributed** (i.i.d.)

New classes [1] may appear:

[1]: Lomonaco and Maltoni, CORe50: a New Dataset and Benchmark for Continuous Object Recognition, 2017

Protocol

Protocol

1. Initialize model f^0
2. Train f^0 on $t = 0$

Protocol

Protocol

1. Initialize model f^0
2. Train f^0 on $t = 0$
3. For $t = 1; t < T; t++$
    1. Initialize model: $f^t \leftarrow f^{t-1}$

Protocol

Protocol

1. Initialize model f^0
2. Train f^0 on $t = 0$
3. For $t = 1; t < T; t++$
    1. Initialize model: $f^t \leftarrow f^{t-1}$
    2. Add classifier weights to f^t

Protocol

Protocol

1. Initialize model f^0
2. Train f^0 on $t = 0$
3. For $t = 1; t < T; t++$
   1. Initialize model: $f^t \leftarrow f^{t-1}$
   2. Add classifier weights to f^t
   3. Train f^t on t

Protocol

Protocol

1. Initialize model f^0
2. Train f^0 on $t = 0$
3. For $t = 1; t < T; t++$
   1. Initialize model: $f^t \leftarrow f^{t-1}$
   2. Add classifier weights to f^t
   3. Train f^t on t
   4. Evaluate f^t on $\{1, \ldots, t\}$

Evaluation

Single-head vs **Multi-heads** during evaluation [14]?

Task 1

Task 2

[14]: Chaudhry et al., Riemannian Walk for Incremental Learning: Understanding Forgetting and Intransigence, 2018

Evaluation

Single-head vs **Multi-heads** during evaluation [14]?

Task 1


Task 2


Final Evaluation: 

[14]: Chaudhry et al., Riemannian Walk for Incremental Learning: Understanding Forgetting and Intransigence, 2018

Evaluation

Single-head vs **Multi-heads** during evaluation [14]?

Task 1


Task 2

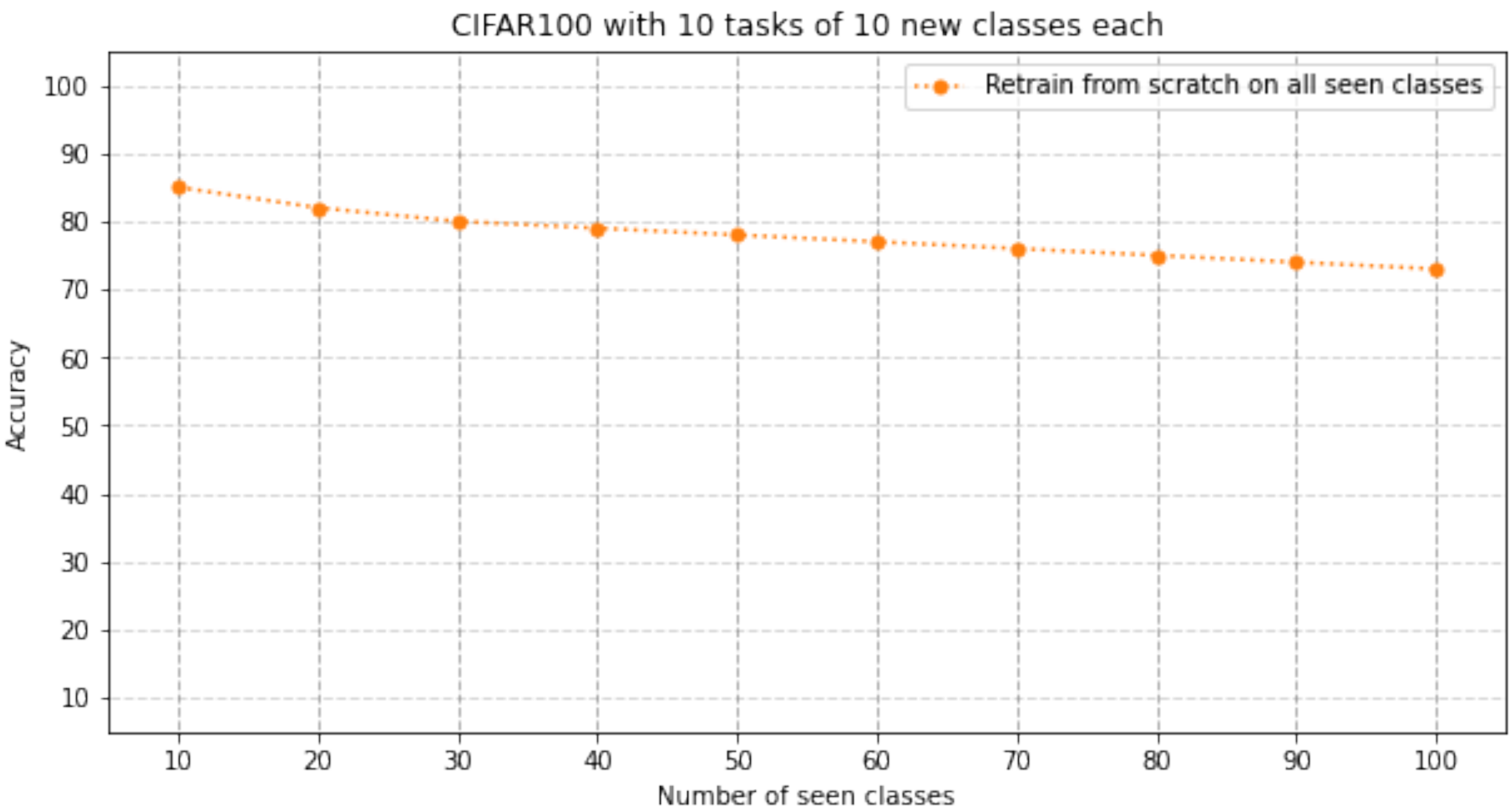

Final Evaluation: 

Single → {dog, cat, boat, plane} ?

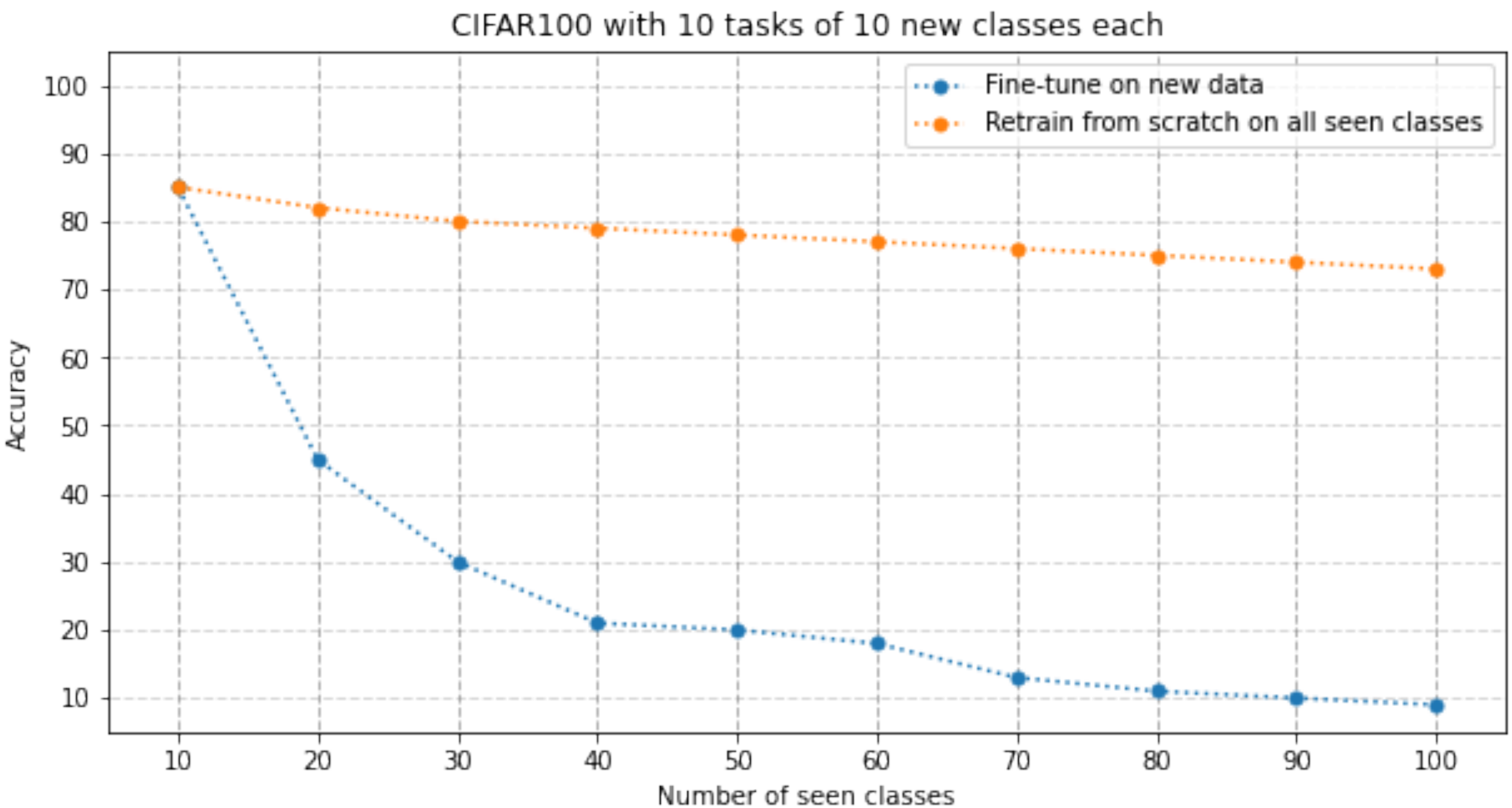
Multi → {dog, cat} ?

[14]: Chaudhry et al., Riemannian Walk for Incremental Learning: Understanding Forgetting and Intransigence, 2018

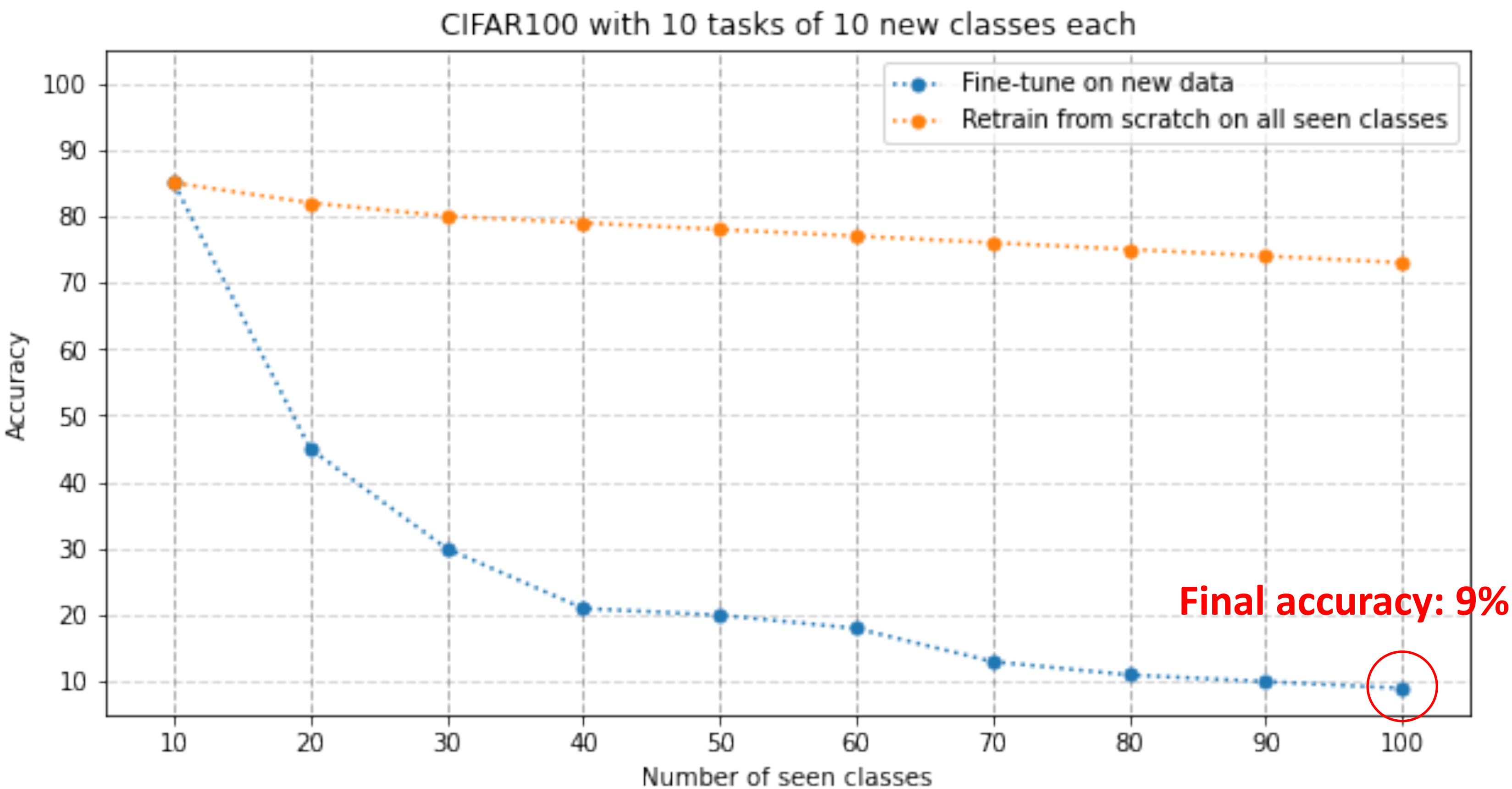
Example


CIFAR100 with 10 tasks of 10 new classes each
Retrain from scratch on all seen classes
Accuracy
100
90
80
70
60
50
40
30
20
10
10
20
30
40
50
60
70
80
90
100
Number of seen classes

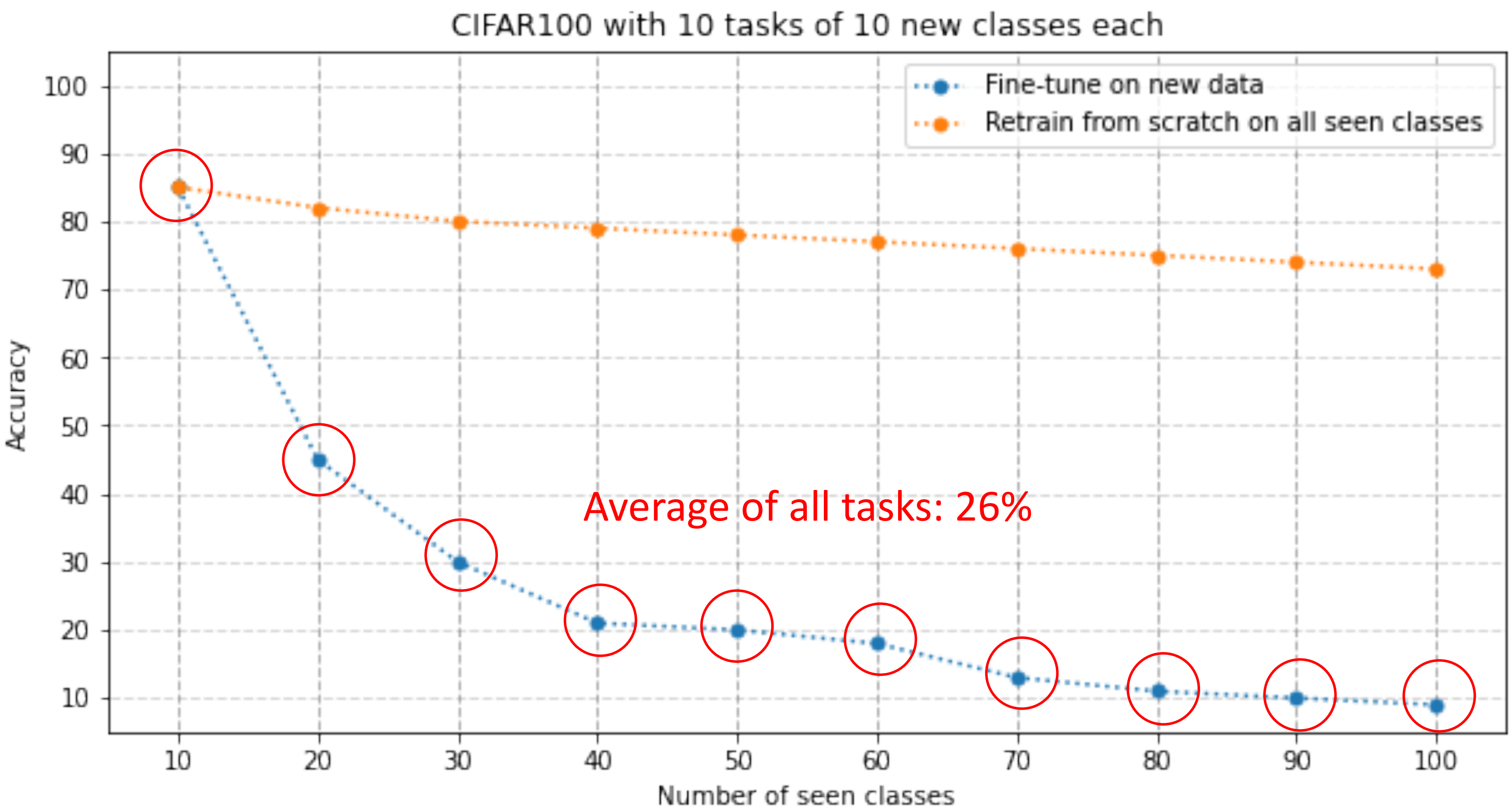
Example


CIFAR100 with 10 tasks of 10 new classes each
Fine-tune on new data
Retrain from scratch on all seen classes
Accuracy
100
90
80
70
60
50
40
30
20
10
10 20 30 40 50 60 70 80 90 100
Number of seen classes

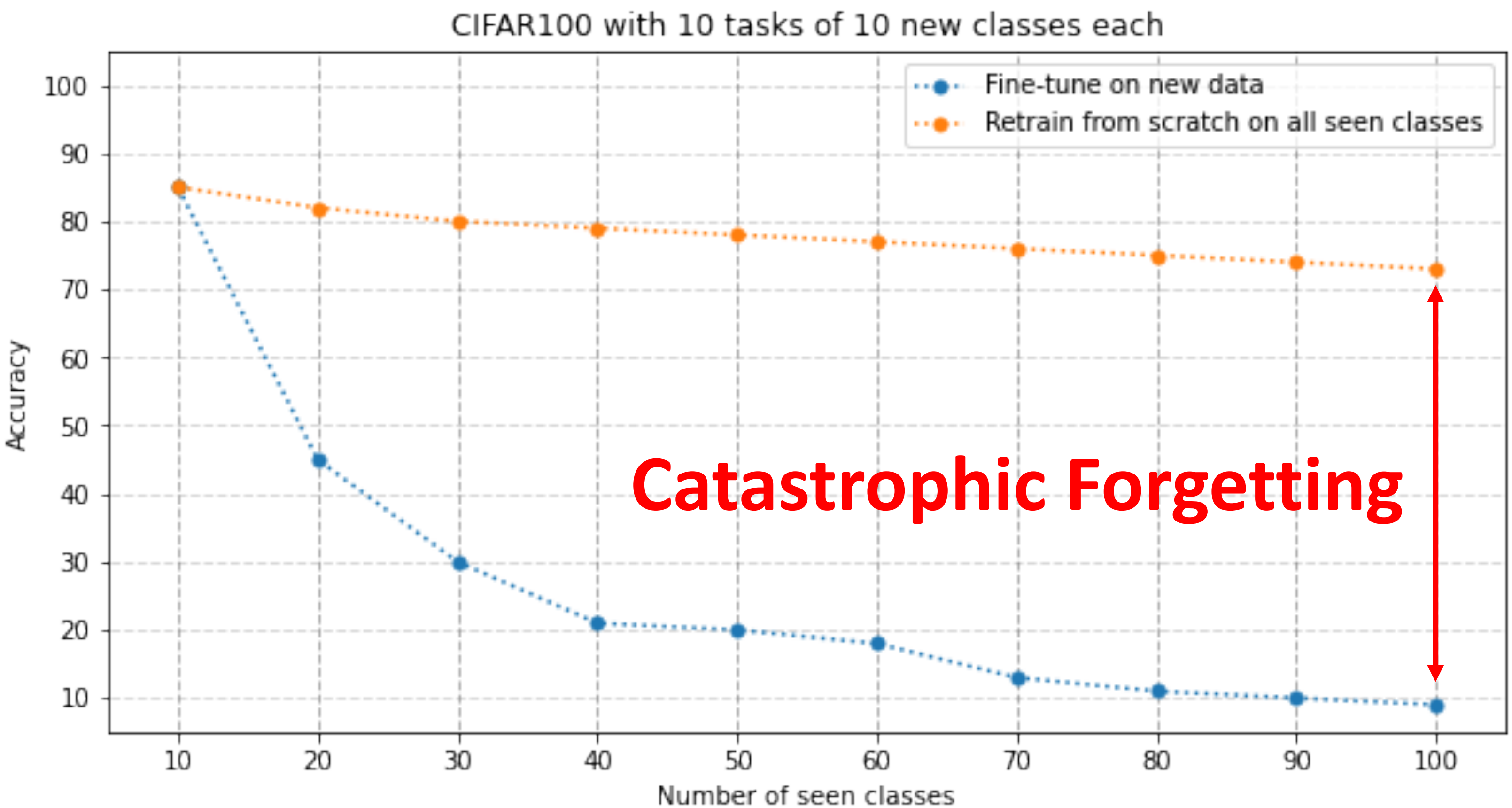
Example


CIFAR100 with 10 tasks of 10 new classes each
Fine-tune on new data
Retrain from scratch on all seen classes
Accuracy
100
90
80
70
60
50
40
30
20
10
Number of seen classes
10
20
30
40
50
60
70
80
90
100
Final accuracy: 9%

Example


CIFAR100 with 10 tasks of 10 new classes each
Fine-tune on new data
Retrain from scratch on all seen classes
Accuracy
100
90
80
70
60
50
40
30
20
10
Average of all tasks: 26%
10
20
30
40
50
60
70
80
90
100
Number of seen classes

Example


CIFAR100 with 10 tasks of 10 new classes each
Fine-tune on new data
Retrain from scratch on all seen classes
Accuracy
100
90
80
70
60
50
40
30
20
10
10
20
30
40
50
60
70
80
90
100
Number of seen classes
Catastrophic Forgetting

How to Solve it?

Broad Strategies

1. Rehearsal
2. Constraints
3. Architecture
4. Classifier Correction

Broad Strategies

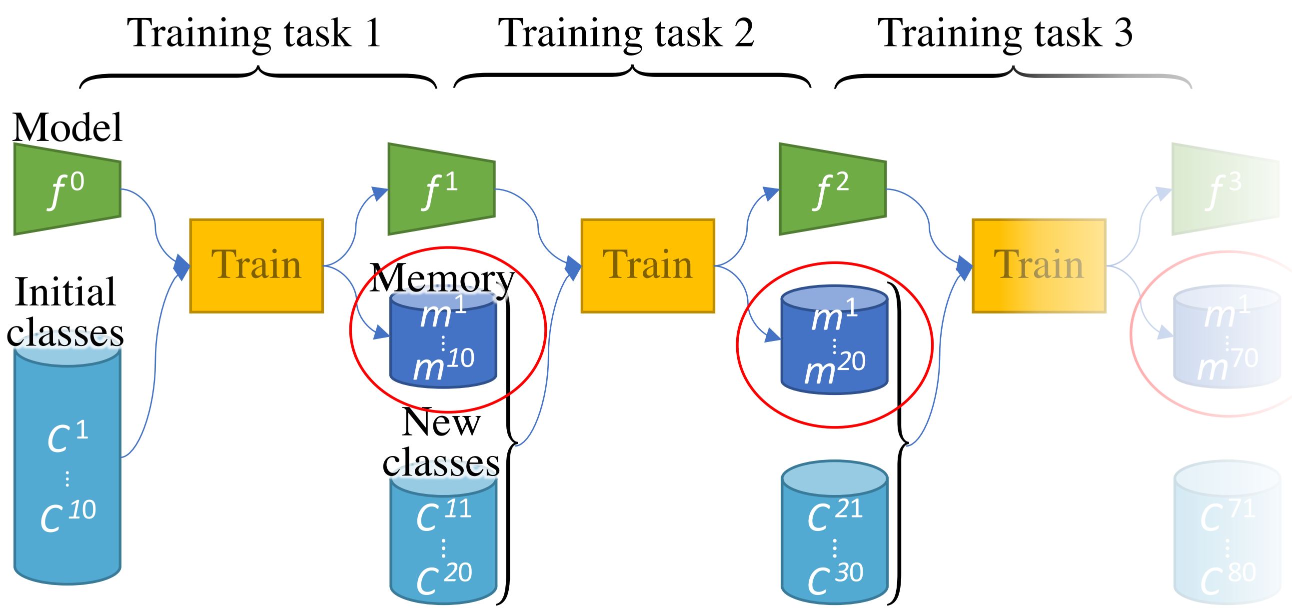
1. **Rehearsal**
2. Constraints
3. Architecture
4. Classifier Correction

1. Rehearsal

heuritech SCIENCES SORBONNE UNIVERSITÉ

Replay a limited amount of previous data

e.g. iCaRL [3]

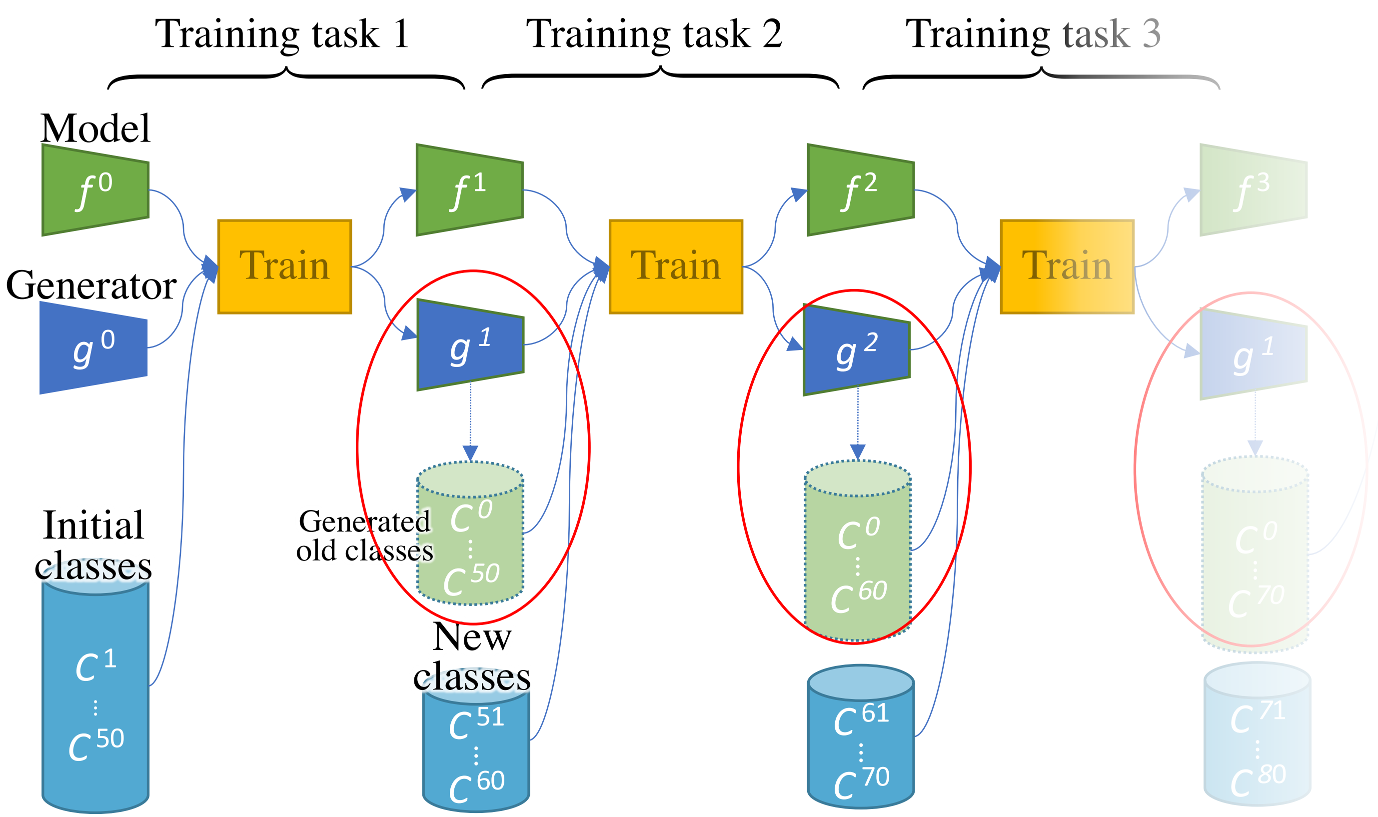


[3]: Rebuffi et al., iCaRL: Incremental Classifier and Representation Learning, 2017

1. Rehearsal

Generate a limited amount of previous data

e.g. DGR [15]



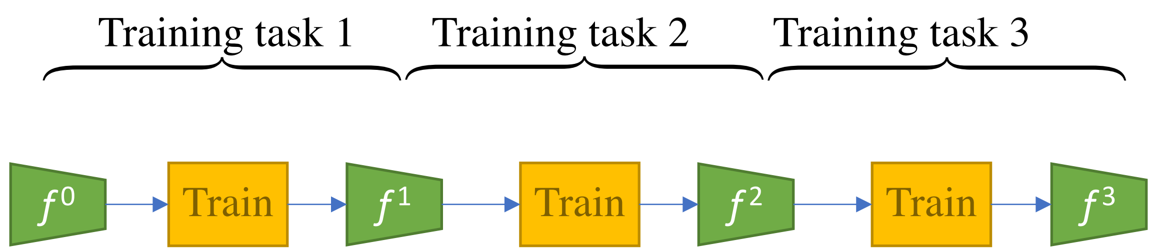
[15]: Shin et al., Continual Learning with Deep Generative Replay, 2017

Broad Strategies

1. Rehearsal
2. **<u>Constraints</u>**
3. Architecture
4. Classifier Correction

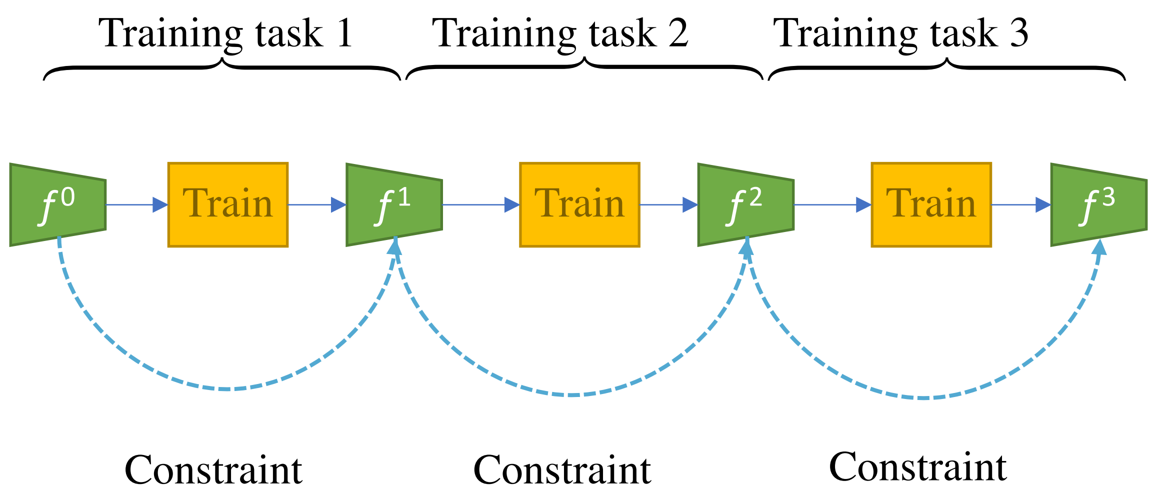
2. Constraints

Constraints between f^{t-1} and f^t:



2. Constraints

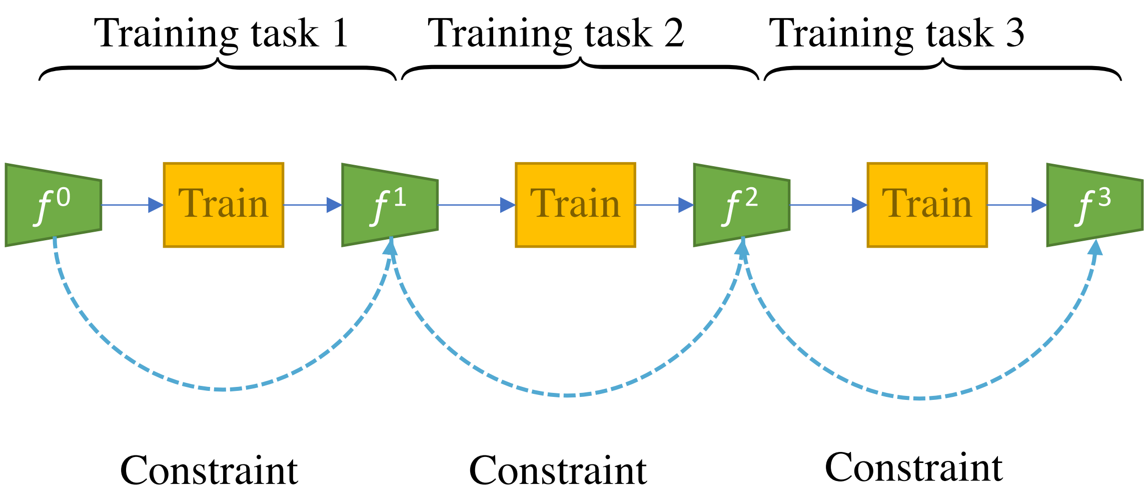
Constraints between f^{t-1} and f^{t}:



2. Constraints

Constraints between f^{t-1} and f^{t}:

On the weights (EWC [4])
On the probabilities (LwF [5])
On the gradients (GEM [6])
On the features (PODNet [7])



[4]: Kirkpatrick et al., Overcoming catastrophic forgetting in neural networks, 2017
[5]: Li and Hoiem, Learning without forgetting, 2016
[6]: Lopez-Paz and Ranzato, Gradient episodic memory for continual learning, 2017
[7]: Douillard et al., PODNet: Pooled Outputs Distillation for small-tasks incremental learning, 2020

Broad Strategies

1. Rehearsal
2. Constraints
3. **<u>Architecture</u>**
4. Classifier Correction

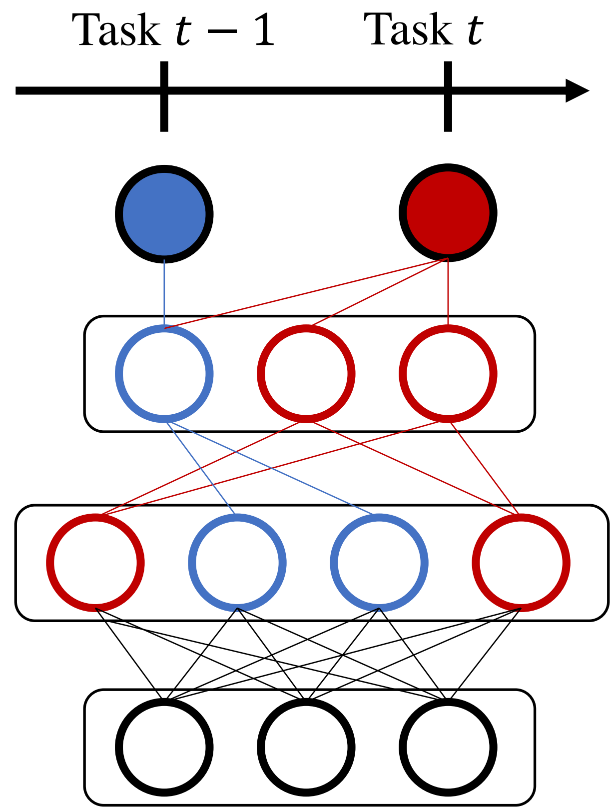
3. Architecture

One **sub-network** per task

Often requires in inference the **task id** to select the task-specific sub-network.

Sub-network can be uncovered via evolutionary algorithms (PathNet [8]), sparsity (Neural Pruning [9]), or learned masks (CPG [10]).

Neurons can also be added (MNTDP-D [16])



Two sub-networks & can co-exist in the same network

[8]: Fernando et al., PathNet: Evolution Channels Gradient Descent in Super Neural Networks , 2017
[9]: Golkar et al., Continual learning via neural pruning, 2019
[10]: Hung et al., Compacting, picking and growing for unforgetting continual learning, 2019
[16] Veniat et al., Efficient Continual Learning with Modular Networks and Task-Drive Priors, 2021

Broad Strategies

1. Rehearsal
2. Constraints
3. Architecture
4. **<u>Classifier Correction</u>**

4. Classifier Correction

Classifier is **biased** towards new classes

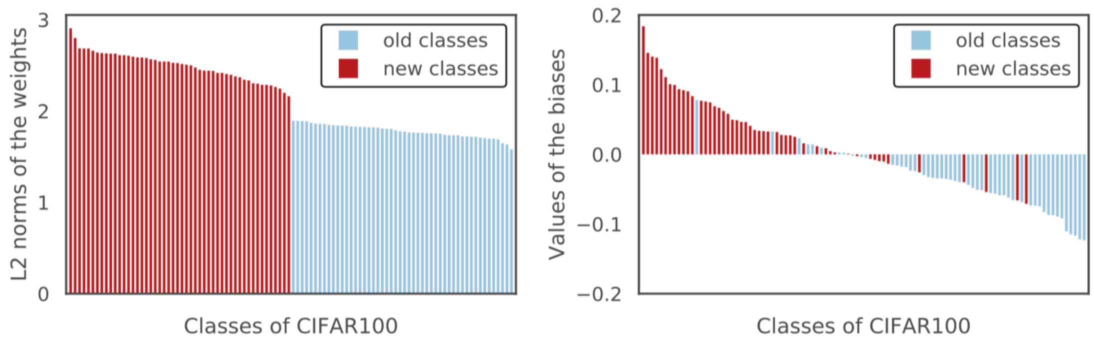
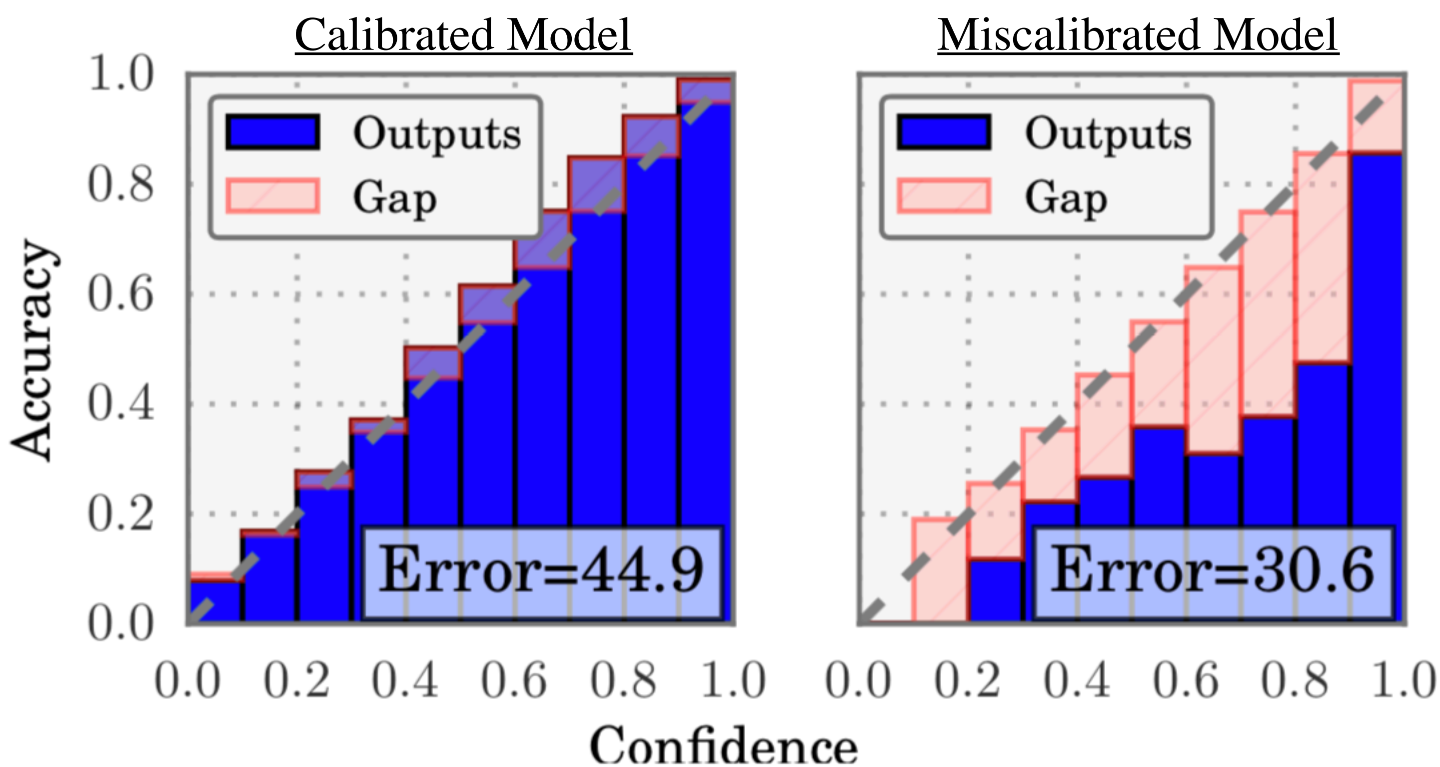


Figure 3. Visualization of the weights and biases in the last layer for old and new classes. The results come from the incremental setting of CIFAR100 (1 phase) by iCaRL [29].

4. Classifier Correction

Classifier is **biased** towards new classes

Can be recalibrated (BiC [11])

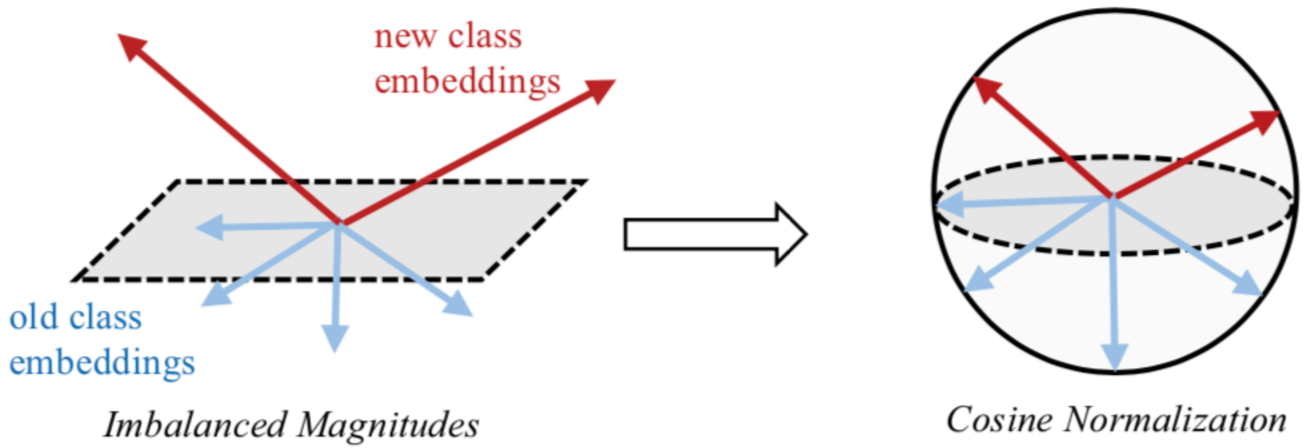


[11]: Wu et al., Large scale incremental learning, 2019

4. Classifier Correction

Classifier is **biased** towards new classes

Or normalized (LUCIR [12])



[11]: Wu et al., Large scale incremental learning, 2019
[12]: Hou et al., Learning an unified classifier incrementally via rebalancing, 2019

Learning without Forgetting
for
Continual Semantic Segmentation

PLOP, CVPR 2021

PLOP: Learning without Forgetting for Continual Semantic Segmentation

Arthur Douillard Yifu Chen Arnaud Dapogny Matthieu Cord

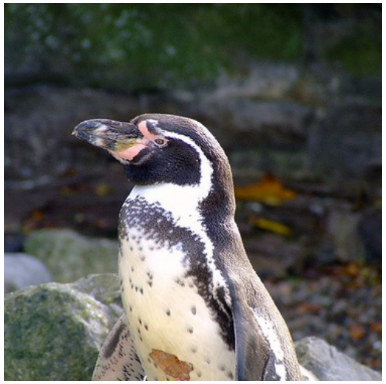
Constraints + Pseudo-labeling

Segmentation

Semantic Segmentation → each pixel is labeled





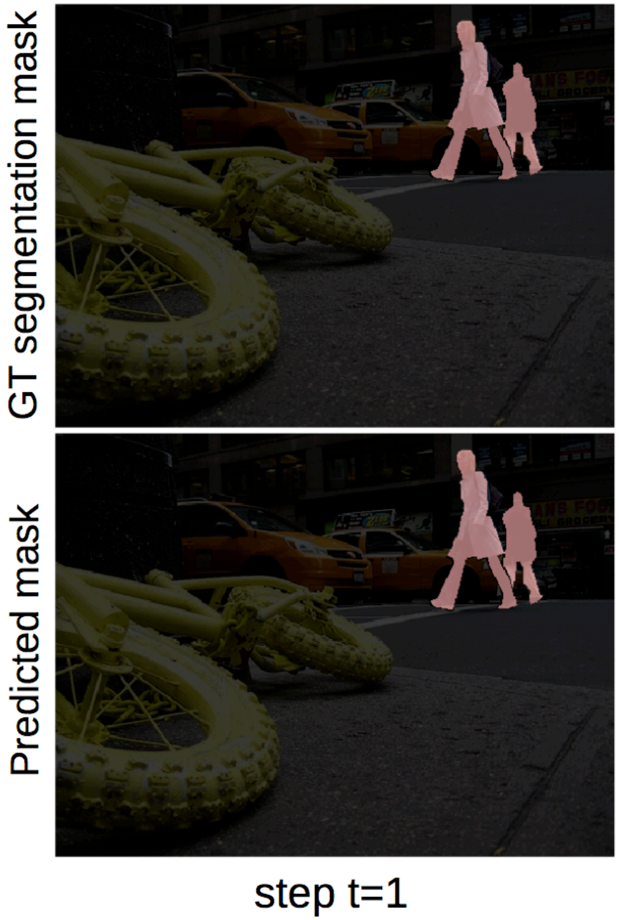



Continual?

Semantic Segmentation → each pixel is labeled

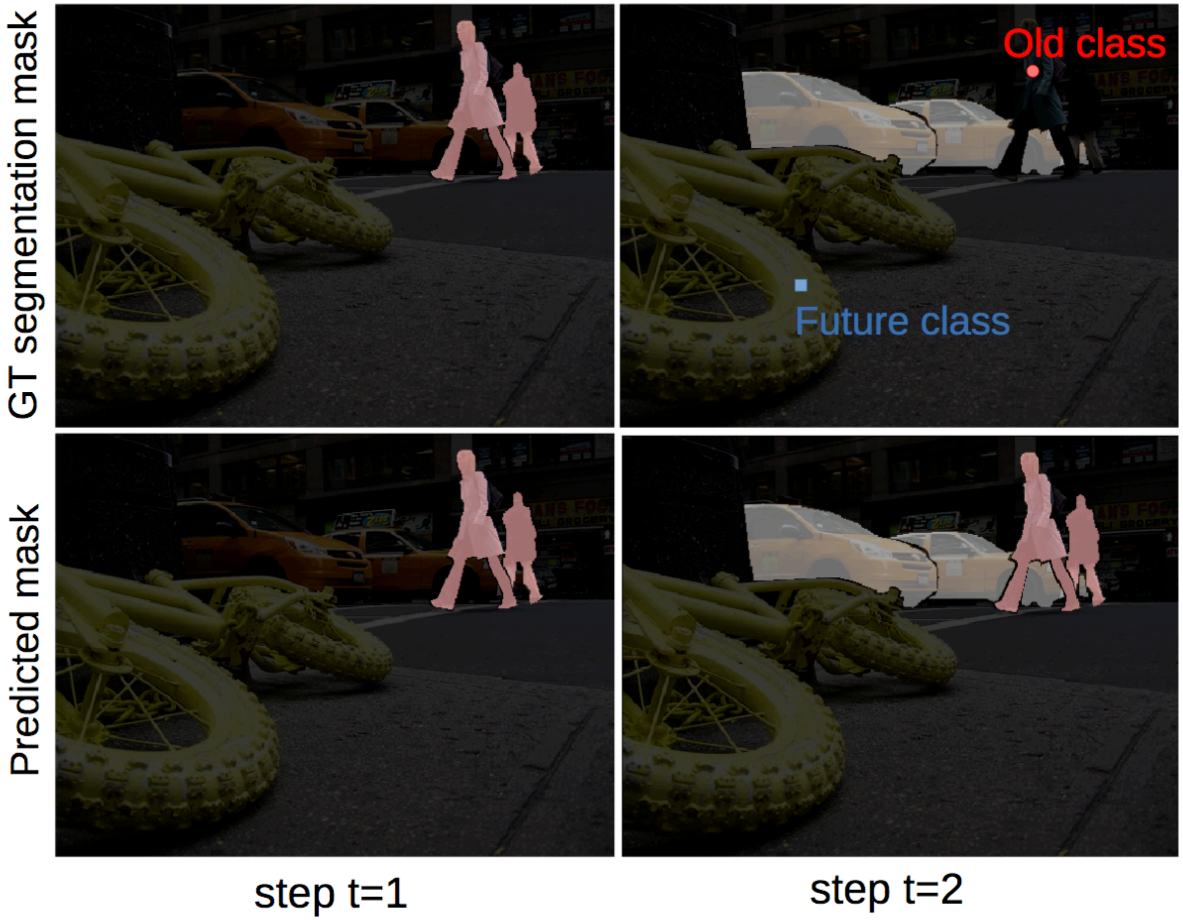
Continual Semantic Segmentation?

Background shift



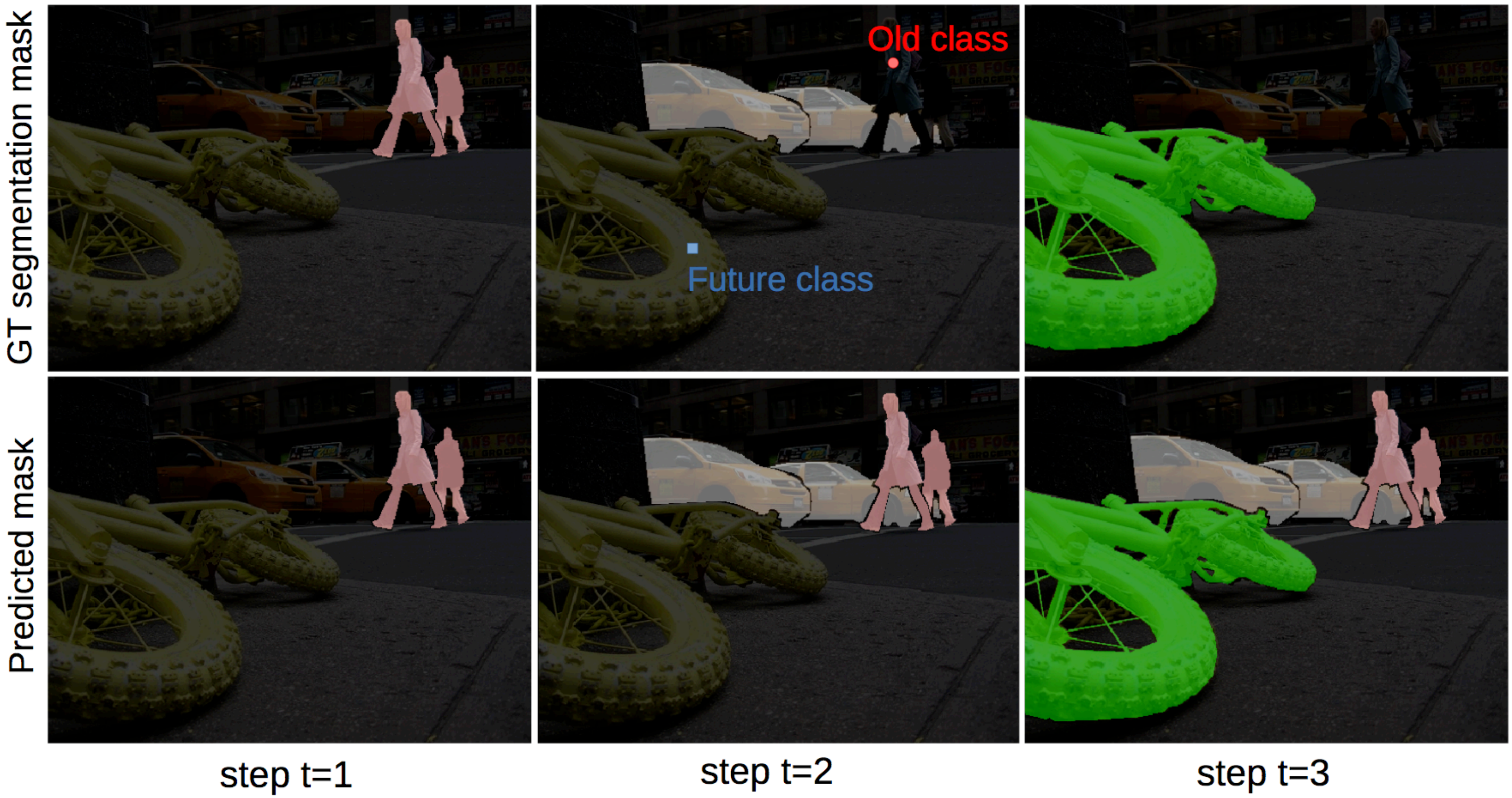
[13]: Cermelli et al., Modeling the Background for Incremental in Semantic Segmentation, 2020

Background shift



[13]: Cermelli et al., Modeling the Background for Incremental in Semantic Segmentation, 2020

Background shift



[13]: Cermelli et al., Modeling the Background for Incremental in Semantic Segmentation, 2020

Problems and weakness

heuritech SCIENCES SORBONNE UNIVERSITÉ

Problems:

- **Forgetting is particularly strong**
  - Previous SotA only constrained final probabilities

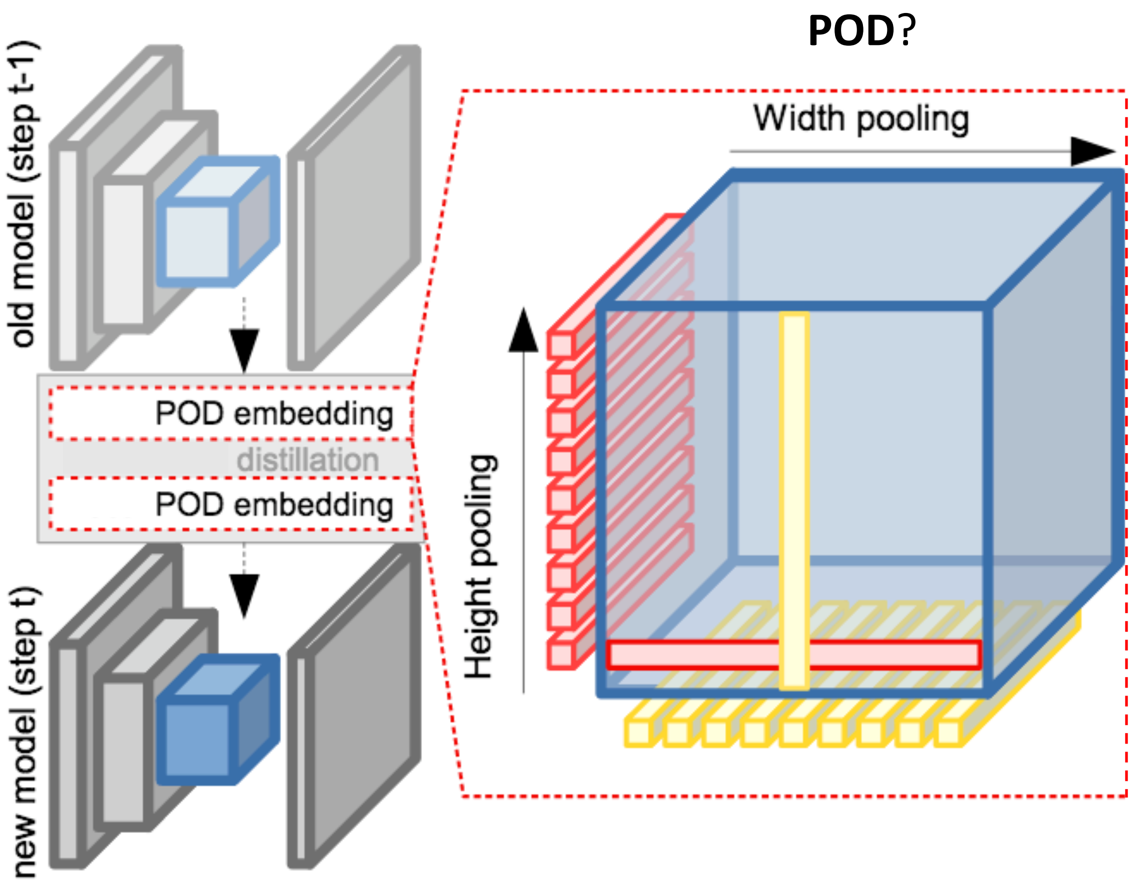
- **Images at task t are partially labeled**
  - Previous SotA maximized the sum of the probabilities of background + old

Problem 1: Forgetting

Problems:

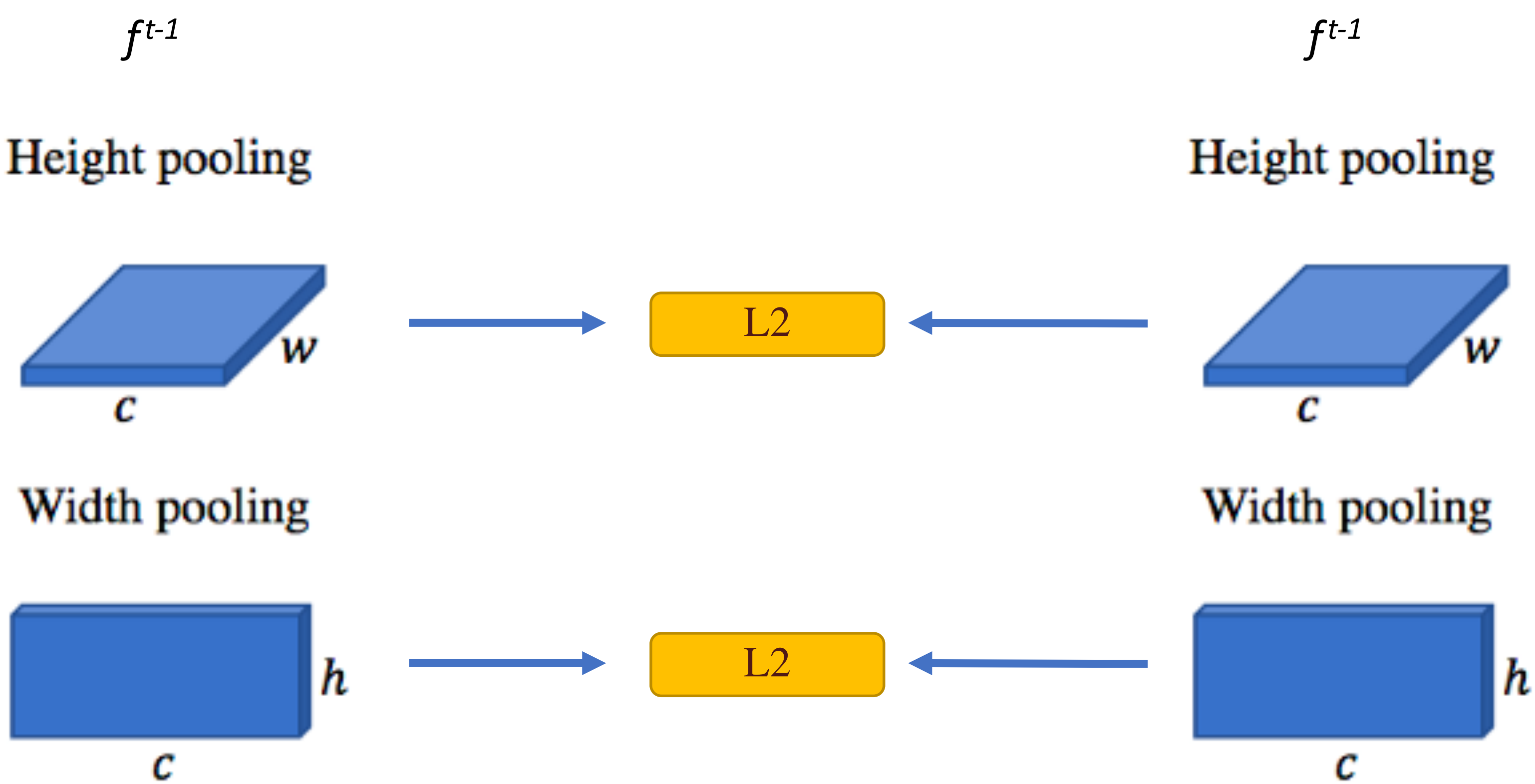
- **Forgetting is particularly strong**
- Images at task t are partially labeled

Problem 1: Forgetting



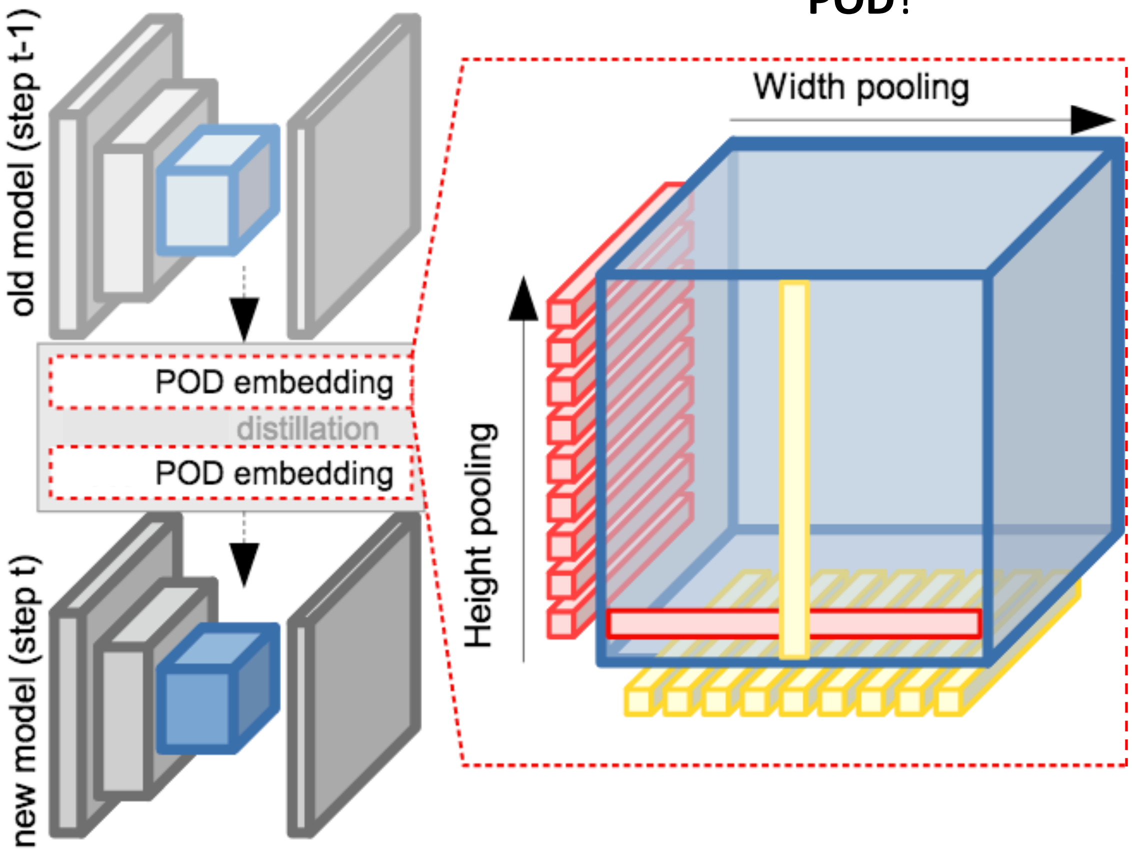
Problem 1: Forgetting

- **Multi-stage features-based distillation loss** (POD)



Problem 1: Forgetting

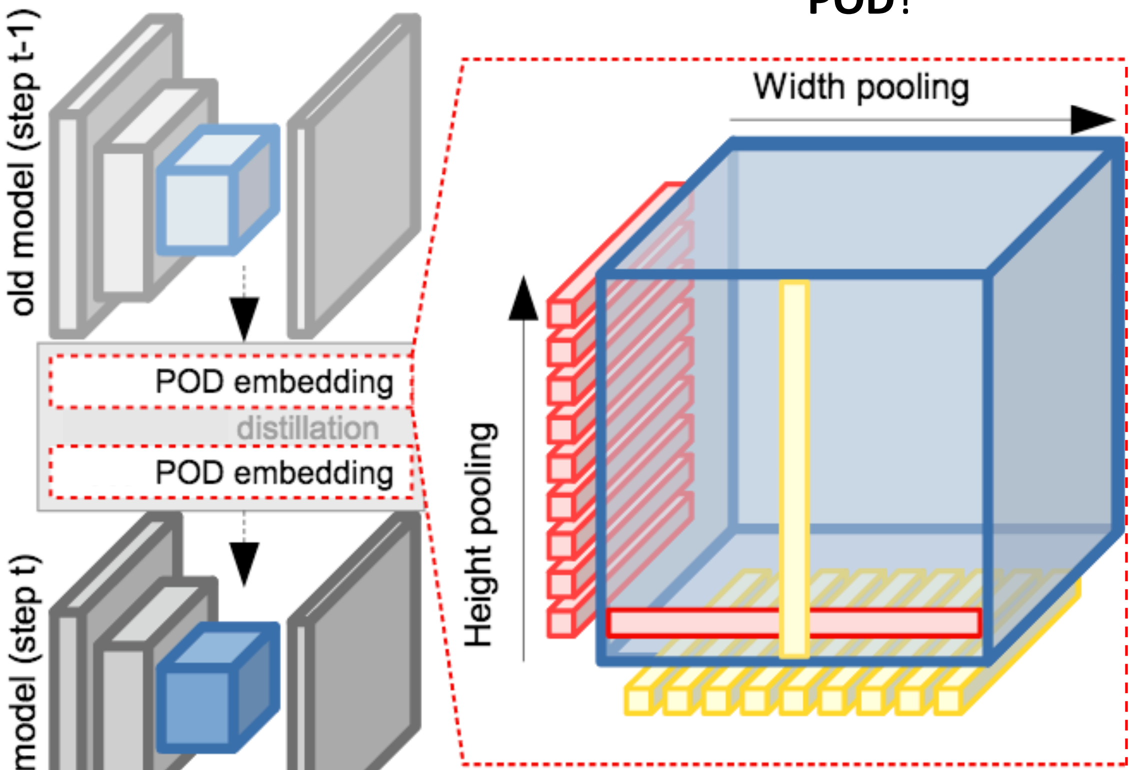
POD?



Problem 1: Forgetting

heuritech SCIENCES SORBONNE UNIVERSITÉ

POD?

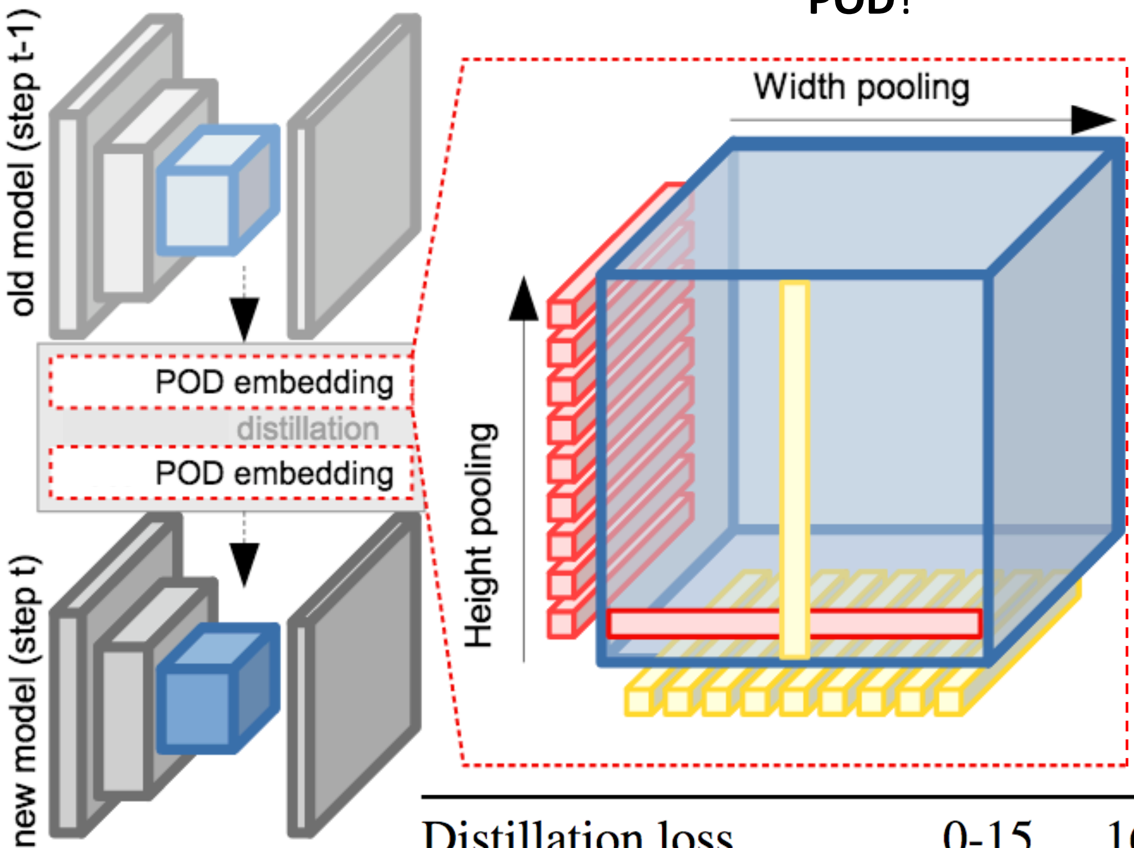


Distillation loss	0-15	16-20	*all*	*avg*
Knowledge Distllation	29.72	4.42	23.69	49.18
UNKD	34.85	5.26	27.80	46.39
POD	43.94	4.82	34.62	53.35

Problem 1: Forgetting

heuritech
SCIENCES SORBONNE UNIVERSITÉ

POD?

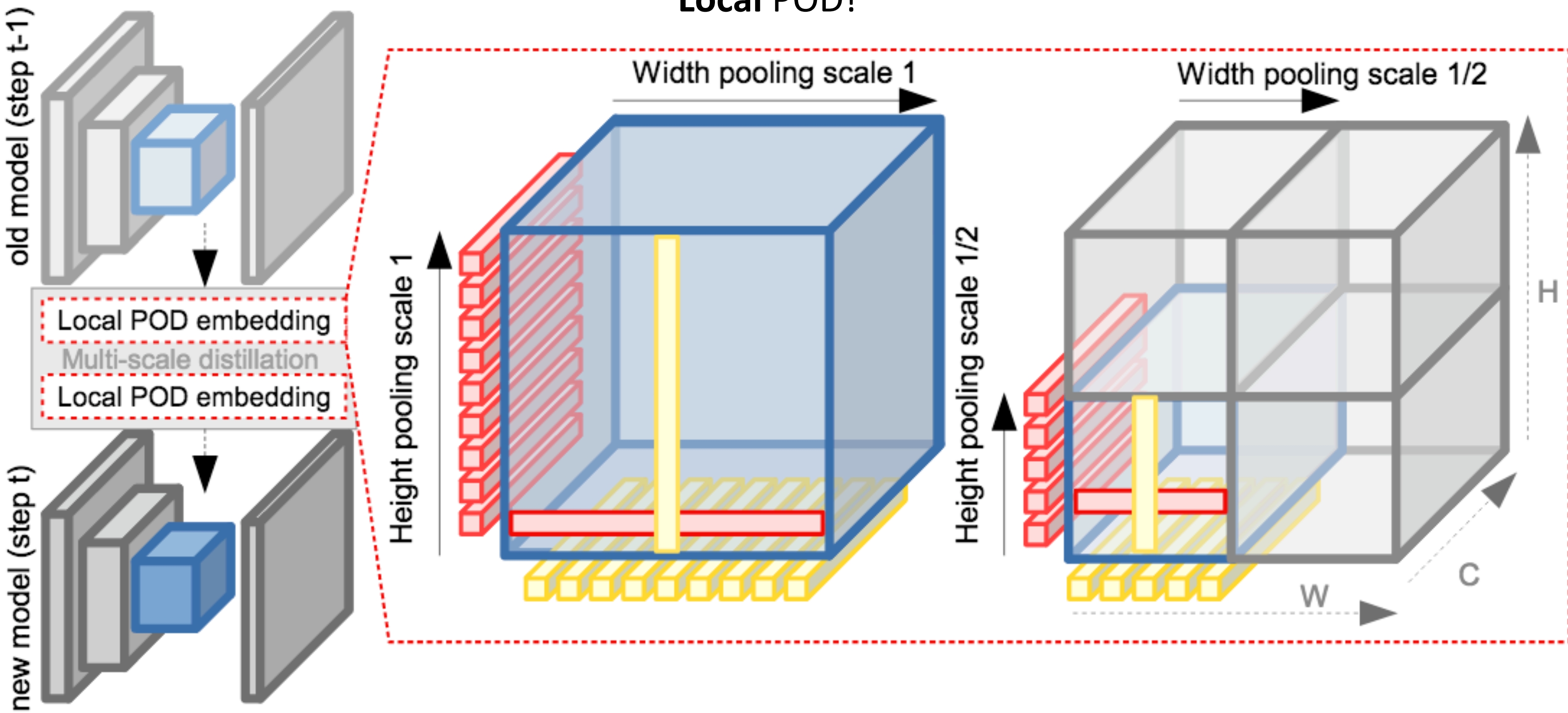


Segmentation
≠
Classification

Distillation loss	0-15	16-20	*all*	*avg*
Knowledge Distllation	29.72	4.42	23.69	49.18
UNKD	34.85	5.26	27.80	46.39
POD	43.94	4.82	34.62	53.35

Problem 1: Forgetting

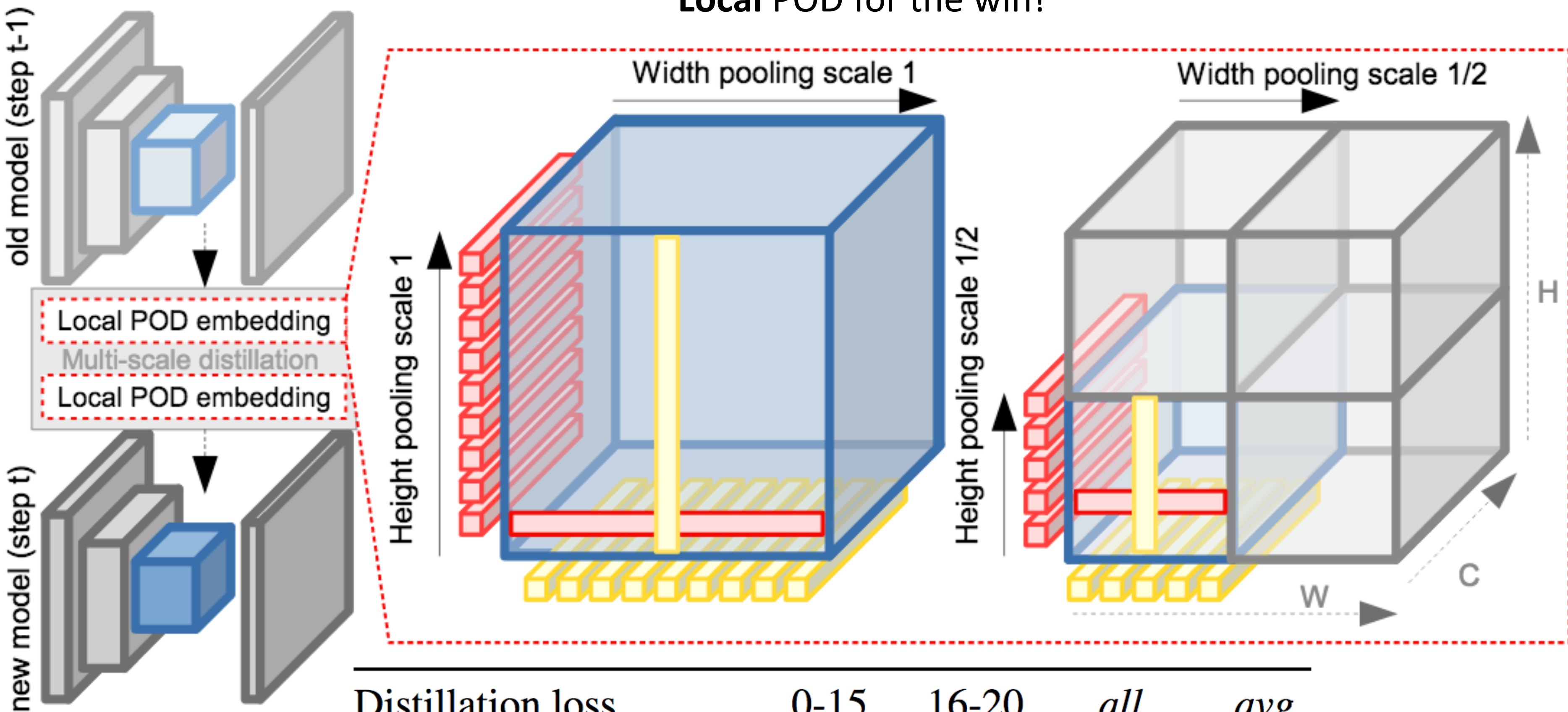
Local POD!



Problem 1: Forgetting

heuritech SCIENCES SORBONNE UNIVERSITÉ

Local POD for the win!



Distillation loss	0-15	16-20	*all*	*avg*
Knowledge Distllation	29.72	4.42	23.69	49.18
UNKD	34.85	5.26	27.80	46.39
POD	43.94	4.82	34.62	53.35
Local POD (Eq. 5)	**63.06**	**17.92**	**52.31**	**65.71**

Problem 1: Background shift

Problems:

- Forgetting is particularly strong

- **Images at task *t* are partially labeled**

Problem 1: Background shift

Step 1

GT

Current Predictions

Problem 1: Background shift

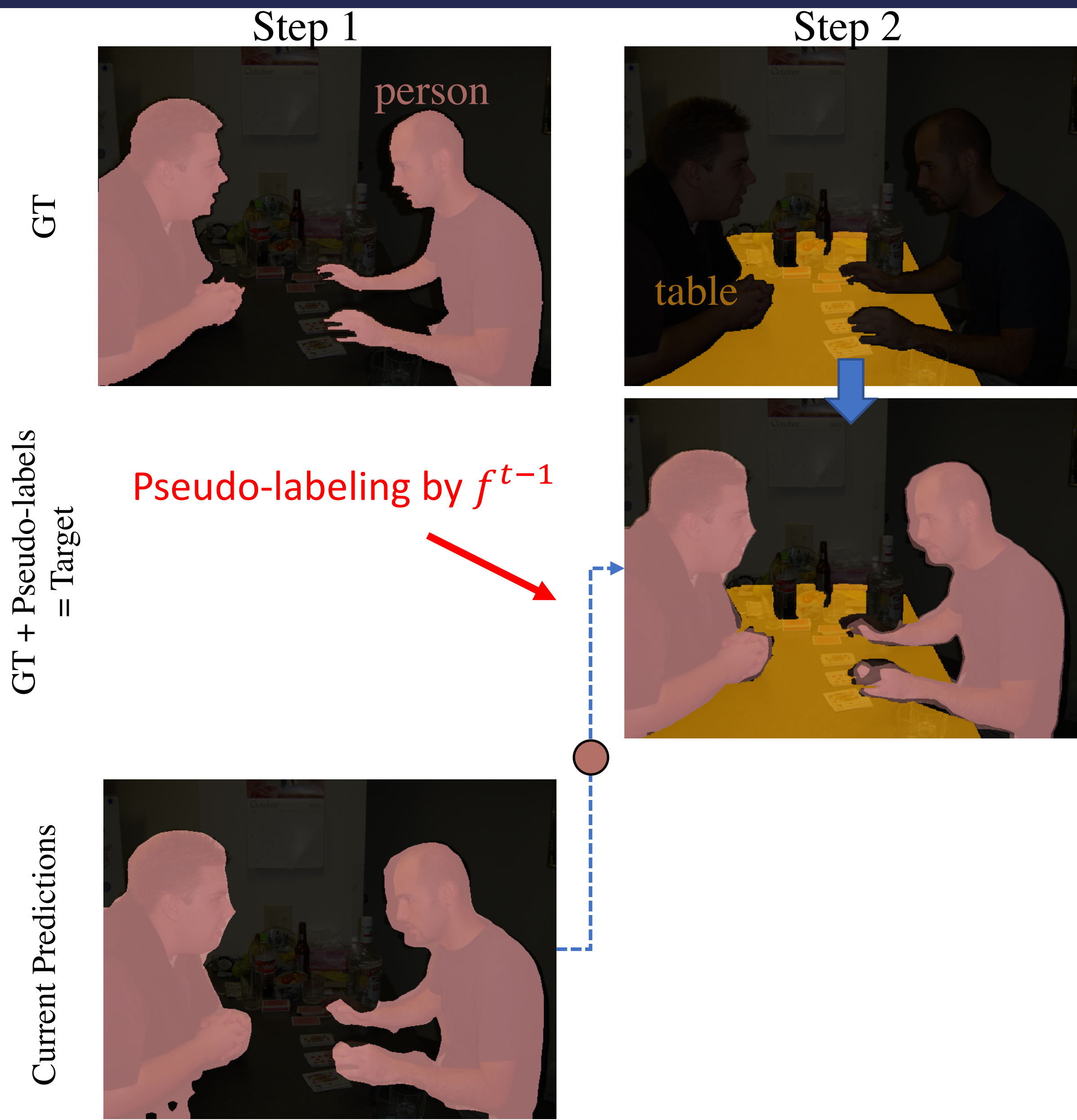
Step 1

Step 2

GT

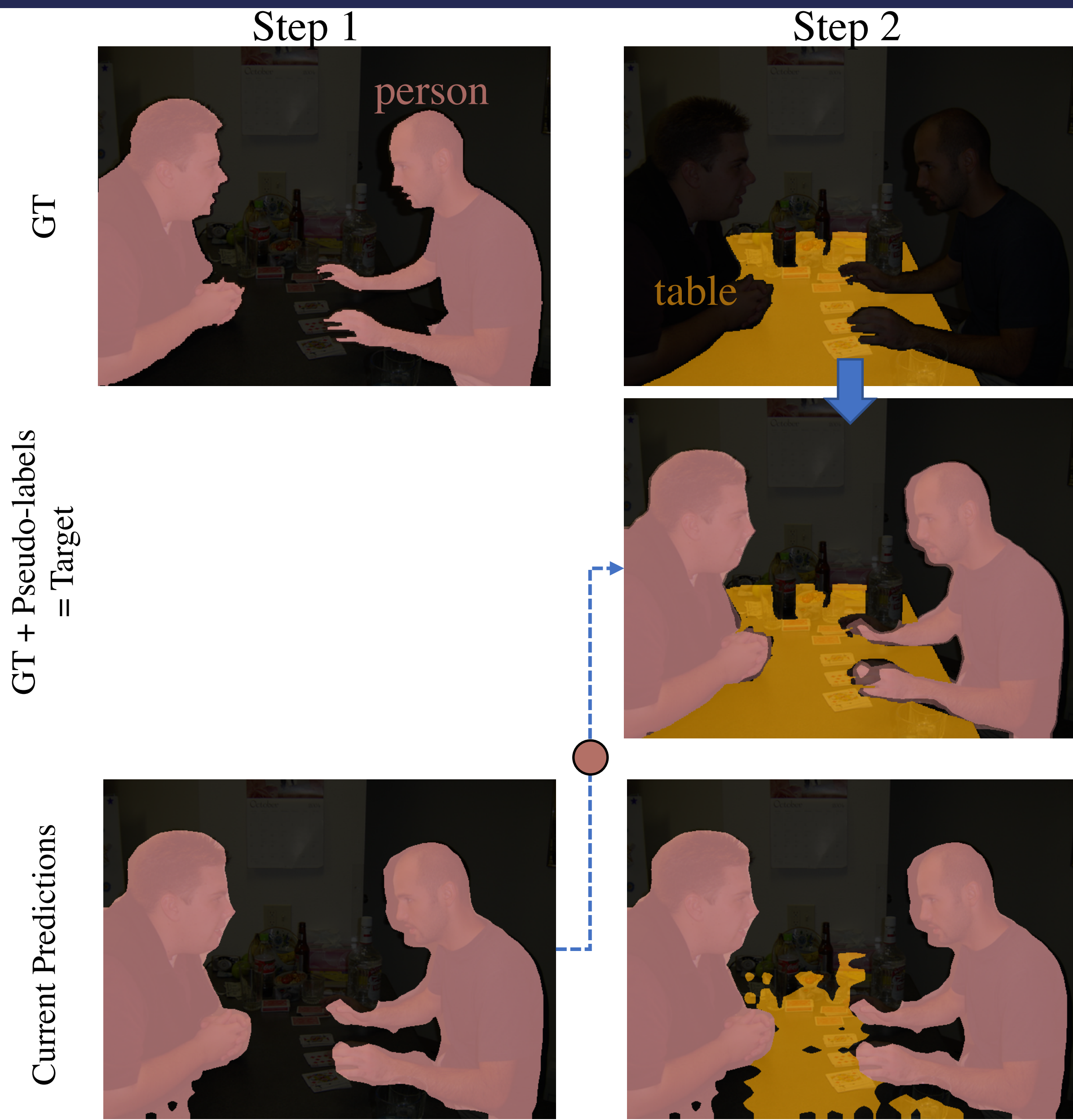
Current Predictions

Problem 1: Background shift



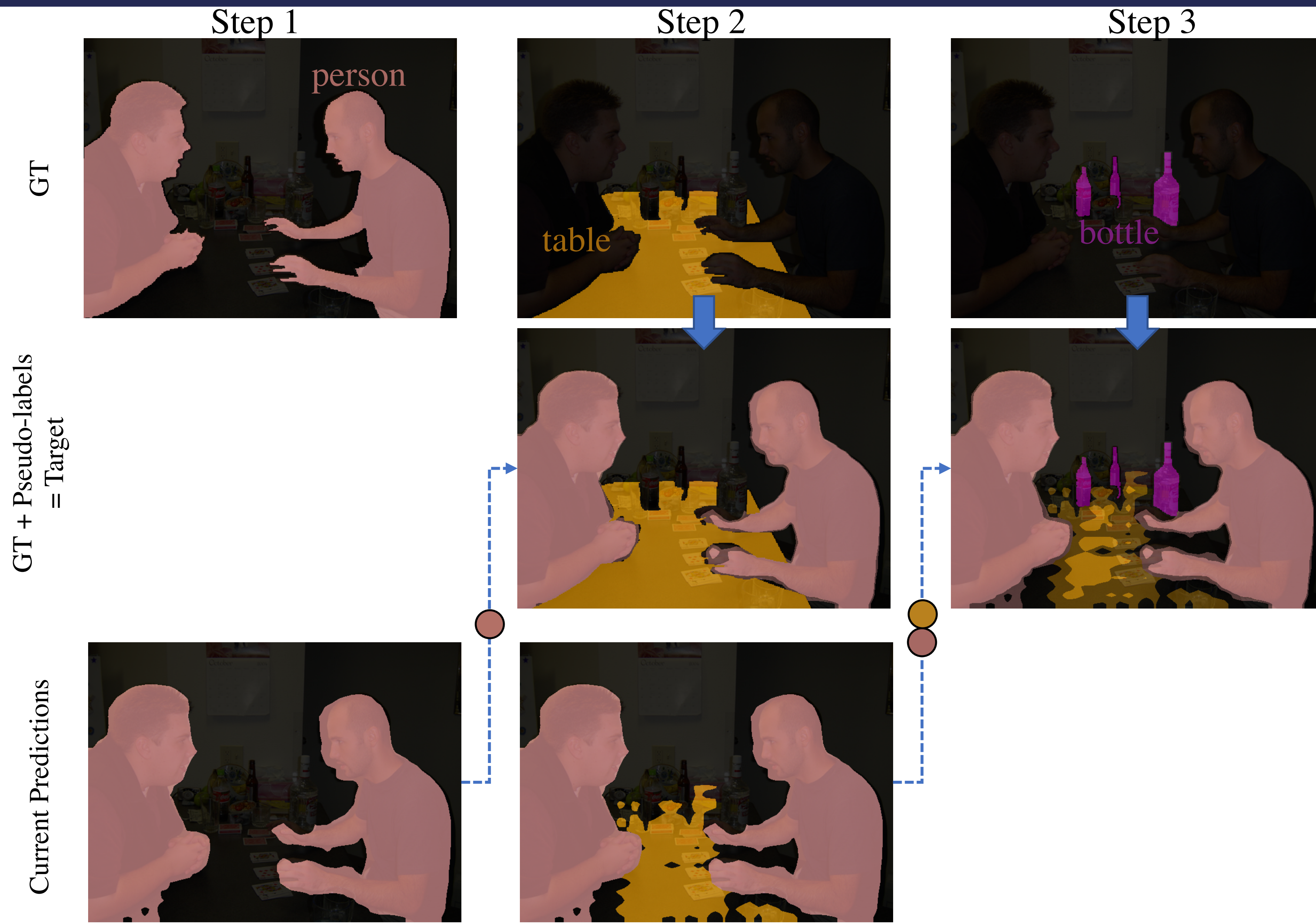
Problem 1: Background shift

heuritech SCIENCES SORBONNE UNIVERSITÉ

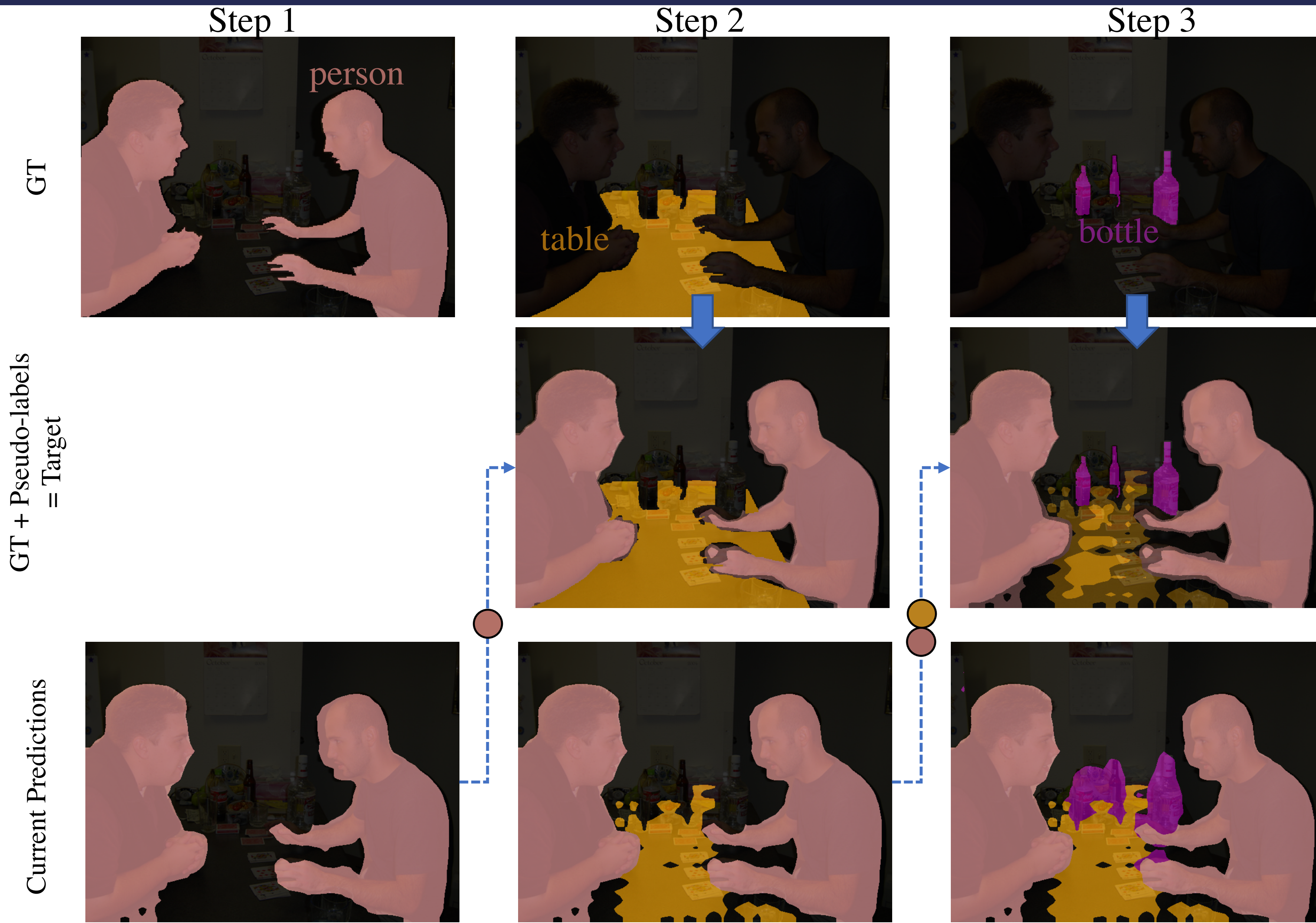


Problem 1: Background shift

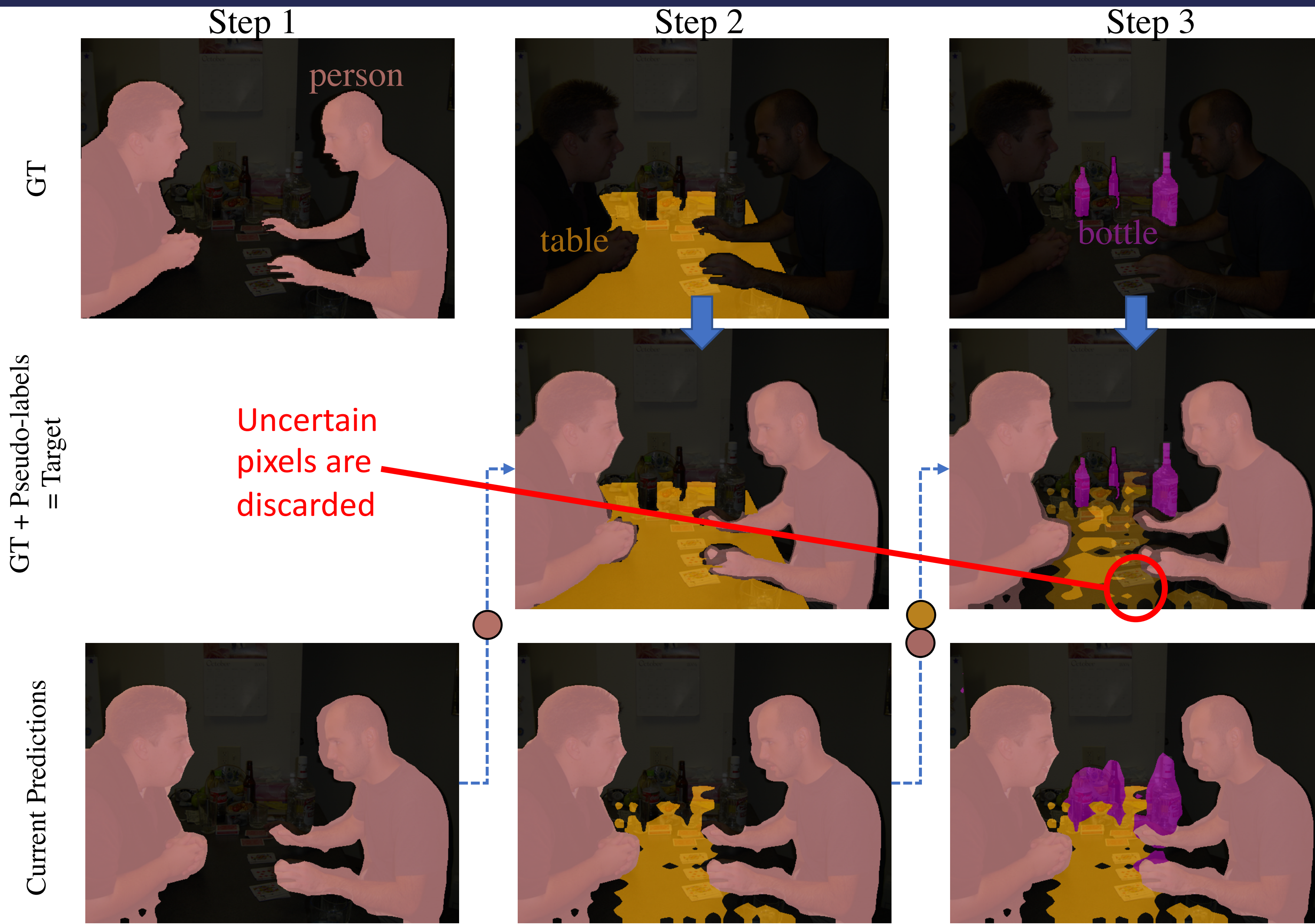
Problem 1: Background shift



Problem 1: Background shift



Problem 1: Background shift



Problem 1: Background shift

heuritech SCIENCES SORBONNE UNIVERSITÉ

UNCE (CVPR 2020) merges predictions of old classes with background

Classification loss	1-15	16-20	*all*	*avg*
CE only on new	12.95	2.54	10.47	47.02
CE	33.80	4.67	26.87	50.79
UNCE	48.46	4.82	38.62	53.19
Pseudo (Eq. 8)	**63.06**	**17.92**	**52.31**	**65.71**
Pseudo-Oracle	*63.69*	*23.35*	*54.09*	*66.05*

Different pseudo-labeling

Pseudo-labeling	*1-15*	*16-20*	*all*	*avg*
Naive	68.28	10.79	54.59	66.77

Pseudo-labelize all pixels that are "**background**"

Different pseudo-labeling

heuritech SCIENCES SORBONNE UNIVERSITÉ

Pseudo-labeling	*1-15*	*16-20*	*all*	*avg*
Naive	68.28	10.79	54.59	66.77
Threshold 0.90	56.63	10.65	54.06	66.43
Median	66.28	11.25	53.18	65.91

Pseudo-labelize all pixels that are "**background**"

And **confident** enough

Different pseudo-labeling

Pseudo-labeling	*1-15*	*16-20*	*all*	*avg*
Naive	68.28	10.79	54.59	66.77
Threshold 0.90	56.63	10.65	54.06	66.43
Median	66.28	11.25	53.18	65.91
Entropy [65]	63.06	17.92	52.31	65.71

Pseudo-labelize all pixels that are ”**background**”

And **entropy** low enough

And **adaptive sample weight**

Experiments

heuritech SCIENCES SORBONNE UNIVERSITÉ

Pascal-VOC (20 classes) experiments

	19-1 (2 tasks)				**15-5** (2 tasks)			
Method	1-19	20	*all*	*avg*	1-15	16-20	*all*	*avg*
EWC† [36]	26.90	14.00	26.30		24.30	35.50	27.10	
LwF-MC† [54]	64.40	13.30	61.90		58.10	35.00	52.30	
ILT† [49]	67.10	12.30	64.40		66.30	40.60	59.90	
ILT [49]	67.75	10.88	65.05	71.23	67.08	39.23	60.45	70.37
MiB† [7]	70.20	22.10	67.80		75.50	49.40	69.00	
MiB [7]	71.43	23.59	69.15	73.28	**76.37**	49.97	**70.08**	**75.12**
PLOP	**75.35**	**37.35**	**73.54**	**75.47**	75.73	**51.71**	**70.09**	**75.19**

Experiments

heuritech SCIENCES SORBONNE UNIVERSITÉ

Pascal-VOC (20 classes) experiments

Method	**19-1** (2 tasks) 1-19	20	*all*	*avg*	**15-5** (2 tasks) 1-15	16-20	*all*	*avg*	**15-1** (6 tasks) 1-15	16-20	*all*	*avg*
EWC† [36]	26.90	14.00	26.30		24.30	35.50	27.10		0.30	4.30	1.30	
LwF-MC† [54]	64.40	13.30	61.90		58.10	35.00	52.30		6.40	8.40	6.90	
ILT† [49]	67.10	12.30	64.40		66.30	40.60	59.90		4.90	7.80	5.70	
ILT [49]	67.75	10.88	65.05	71.23	67.08	39.23	60.45	70.37	8.75	7.99	8.56	40.16
MiB† [7]	70.20	22.10	67.80		75.50	49.40	69.00		35.10	13.50	29.70	
MiB [7]	71.43	23.59	69.15	73.28	**76.37**	49.97	**70.08**	**75.12**	34.22	13.50	29.29	54.19
PLOP	**75.35**	**37.35**	**73.54**	**75.47**	75.73	**51.71**	**70.09**	**75.19**	**65.12**	**21.11**	**54.64**	**67.21**

Experiments

Pascal-VOC (20 classes) experiments

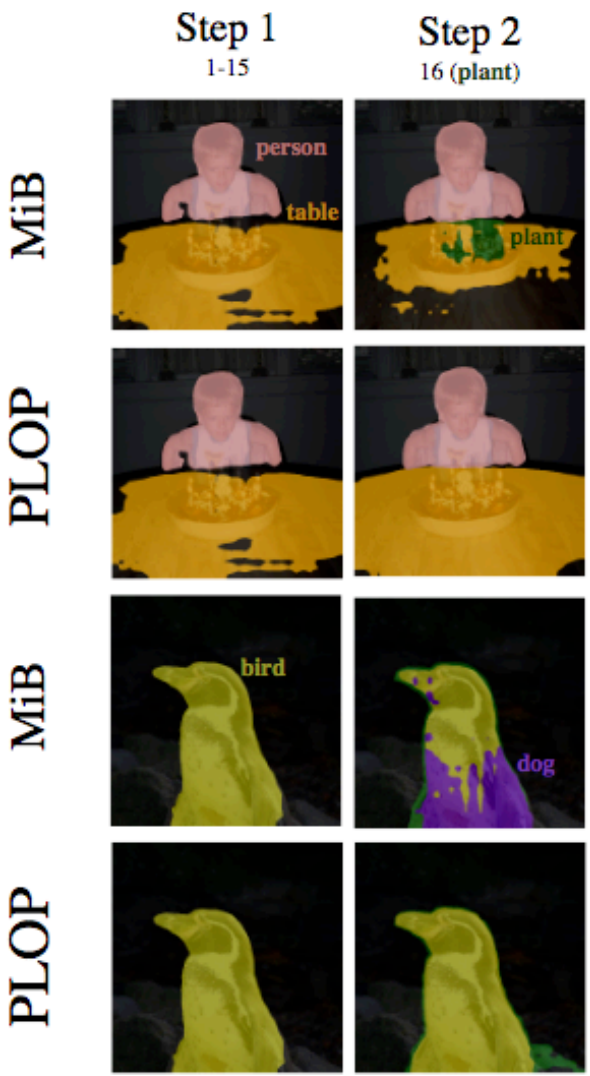
Method	**19-1** (2 tasks) 1-19	20	*all*	*avg*	**15-5** (2 tasks) 1-15	16-20	*all*	*avg*	**15-1** (6 tasks) 1-15	16-20	*all*	*avg*
EWC† [36]	26.90	14.00	26.30		24.30	35.50	27.10		0.30	4.30	1.30	
LwF-MC† [54]	64.40	13.30	61.90		58.10	35.00	52.30		6.40	8.40	6.90	
ILT† [49]	67.10	12.30	64.40		66.30	40.60	59.90		4.90	7.80	5.70	
ILT [49]	67.75	10.88	65.05	71.23	67.08	39.23	60.45	70.37	8.75	7.99	8.56	40.16
MiB† [7]	70.20	22.10	67.80		75.50	49.40	69.00		35.10	13.50	29.70	
MiB [7]	71.43	23.59	69.15	73.28	**76.37**	49.97	**70.08**	**75.12**	34.22	13.50	29.29	54.19
PLOP	**75.35**	**37.35**	**73.54**	**75.47**	75.73	**51.71**	**70.09**	**75.19**	**65.12**	**21.11**	**54.64**	**67.21**

Method	**VOC 10-1** (11 tasks) 1-10	11-20	*all*	*avg*
ILT [55]	7.15	3.67	5.50	25.71
MiB [8]	12.25	13.09	12.65	42.67
PLOP	**44.03**	**15.51**	**30.45**	**52.32**

Visuals

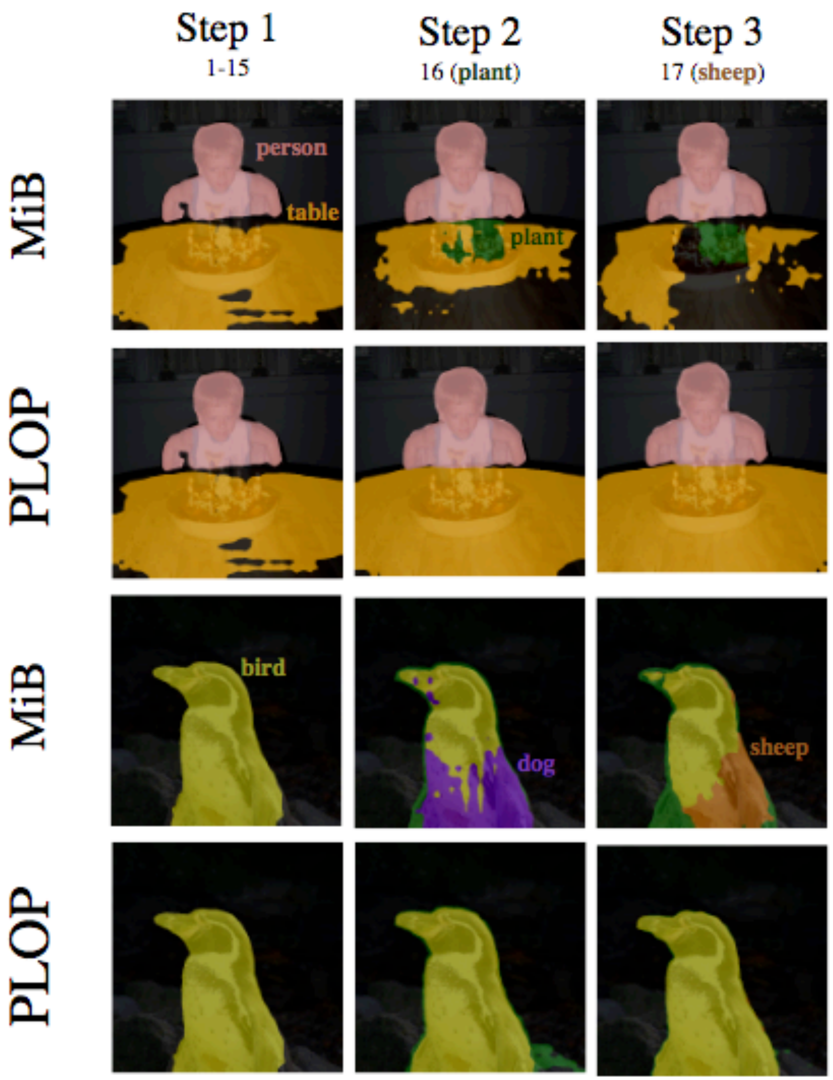
First, learn 15 classes

Visuals



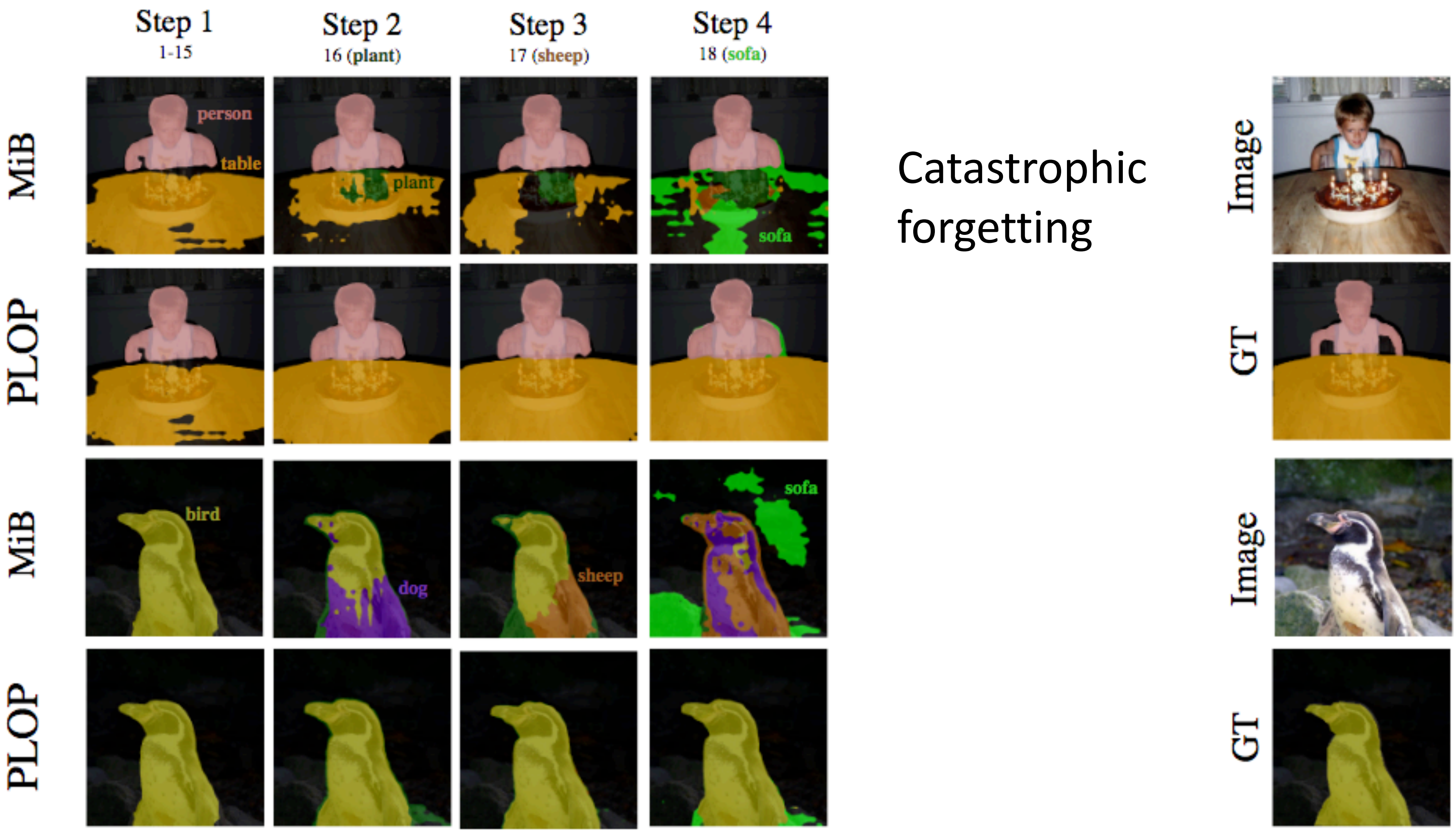
Learn the "plant" class

Visuals



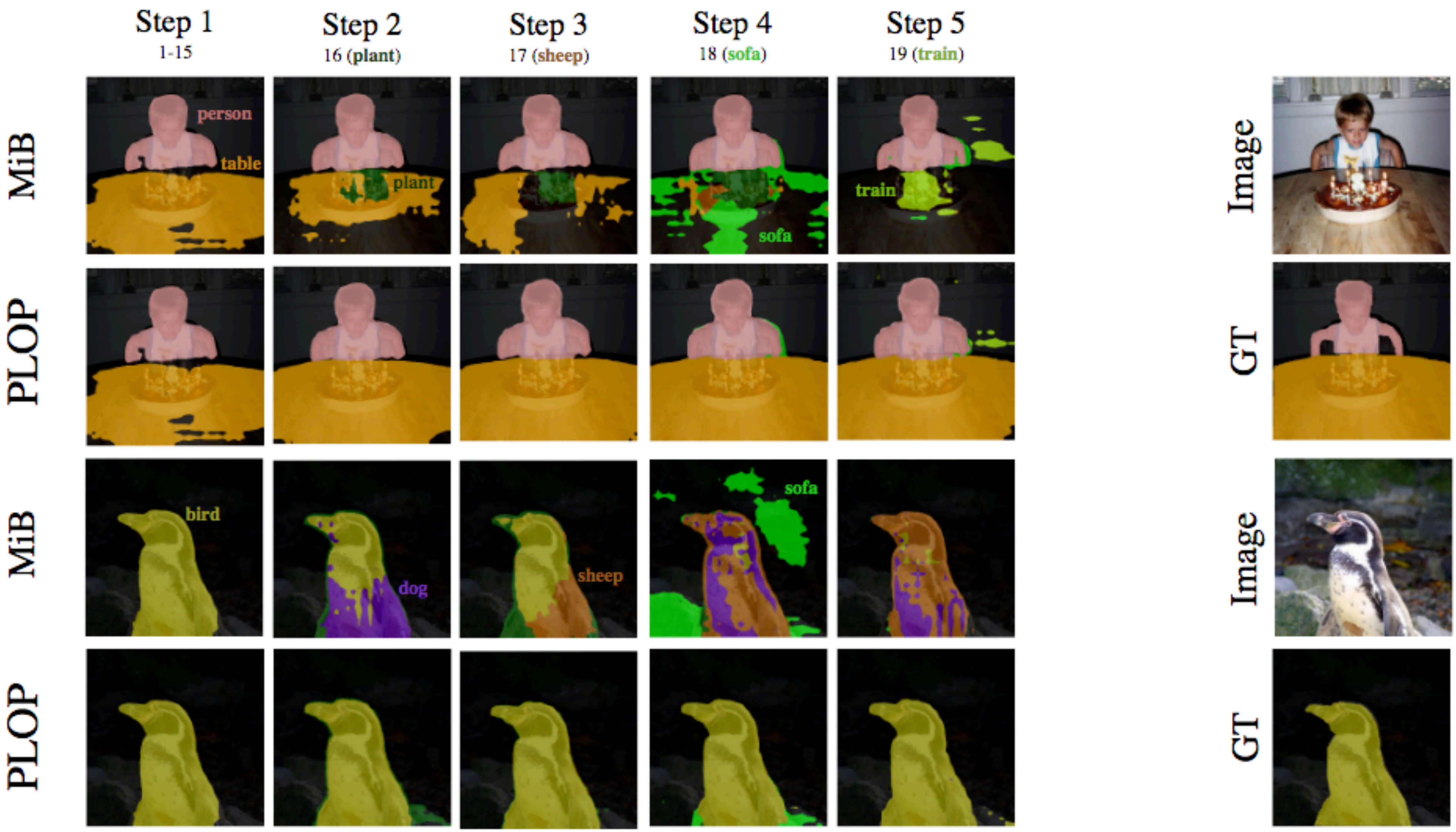
So far, it's still OK

Visuals

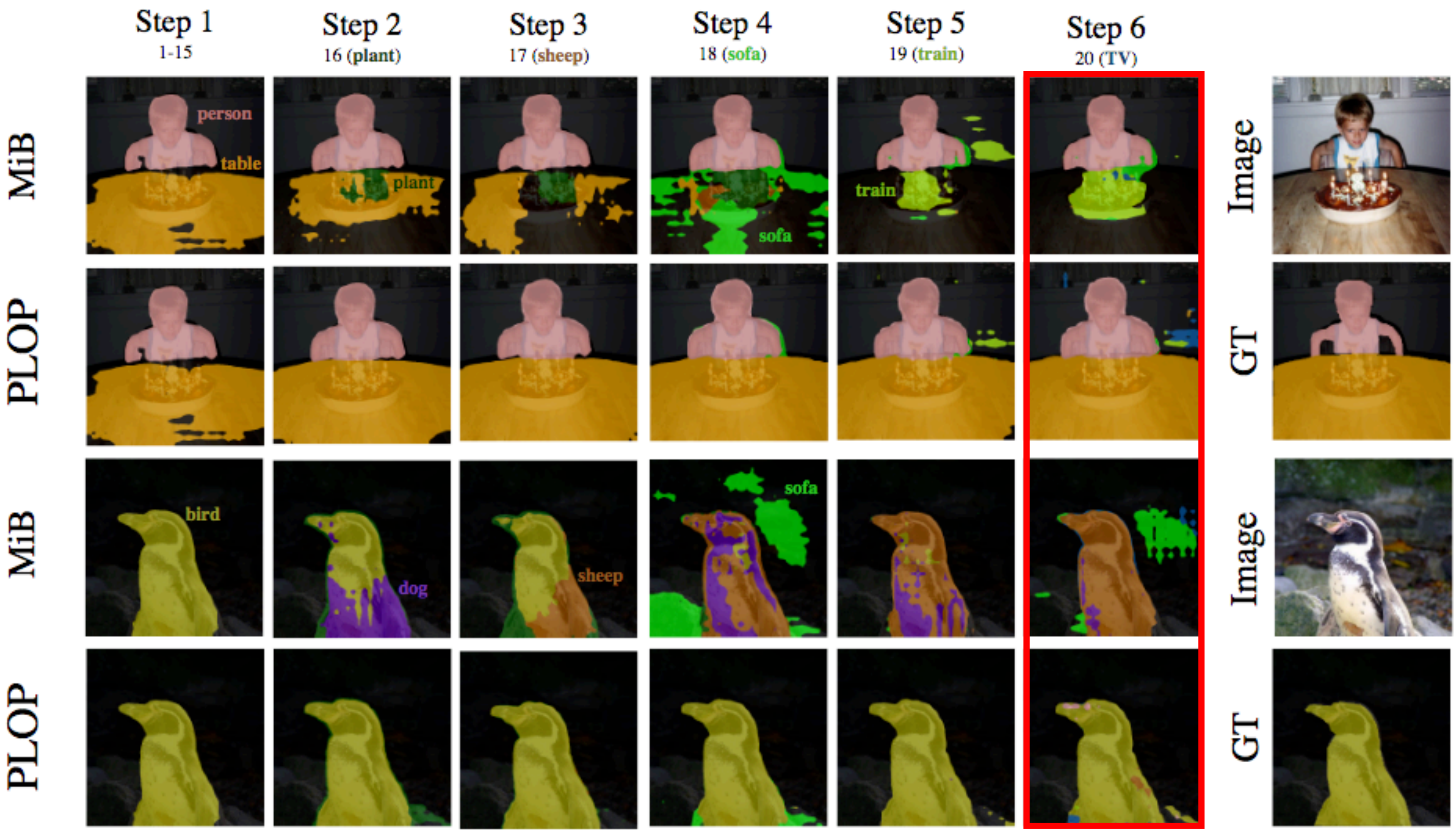


Catastrophic forgetting

Visuals

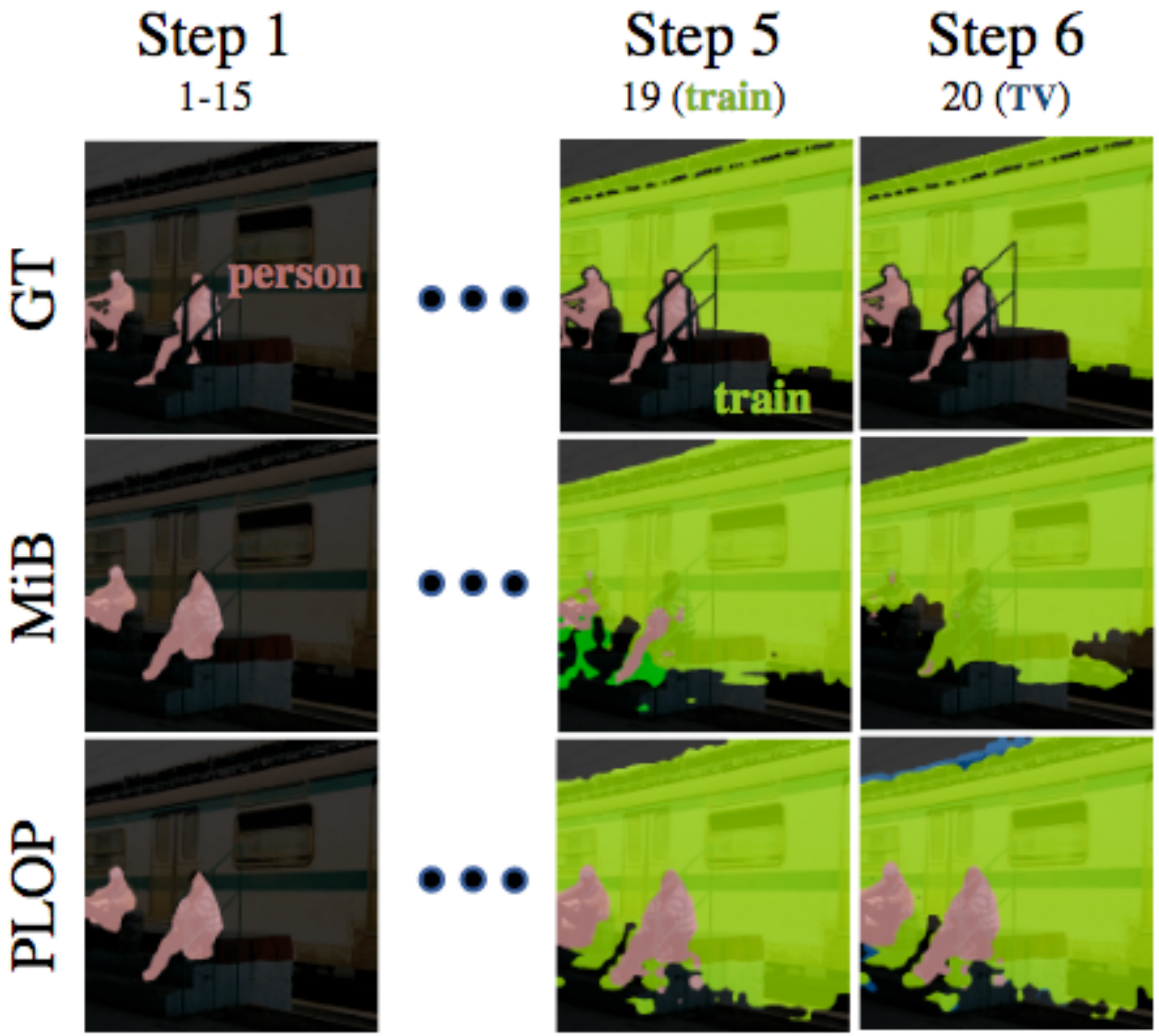

Step 1
1-15
Step 2
16 (plant)
Step 3
17 (sheep)
Step 4
18 (sofa)
Step 5
19 (train)
MiB
PLOP
MiB
PLOP
Image
GT
Image
GT
person
table
plant
sofa
train
bird
dog
sheep
sofa

Visuals


Step 1
1-15
Step 2
16 (plant)
Step 3
17 (sheep)
Step 4
18 (sofa)
Step 5
19 (train)
Step 6
20 (TV)
MiB
PLOP
MiB
PLOP
Image
GT
Image
GT
person
table
plant
sofa
train
bird
dog
sheep
sofa

Visuals

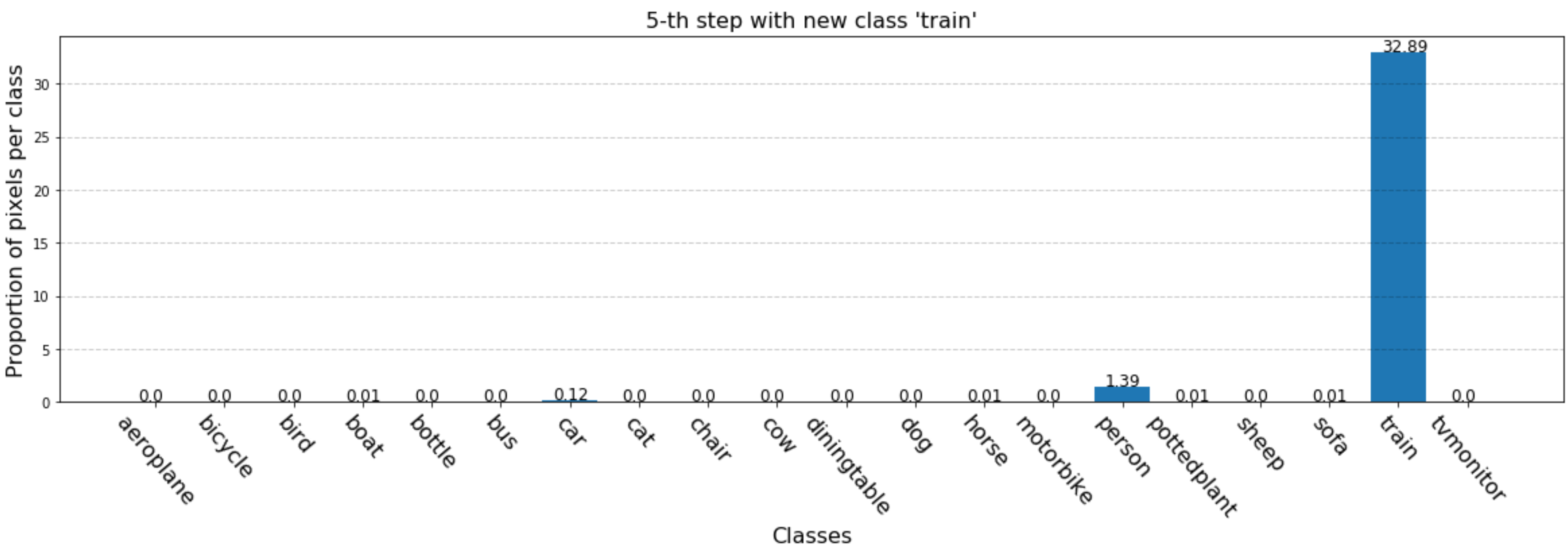
When a class appear only latter in the image



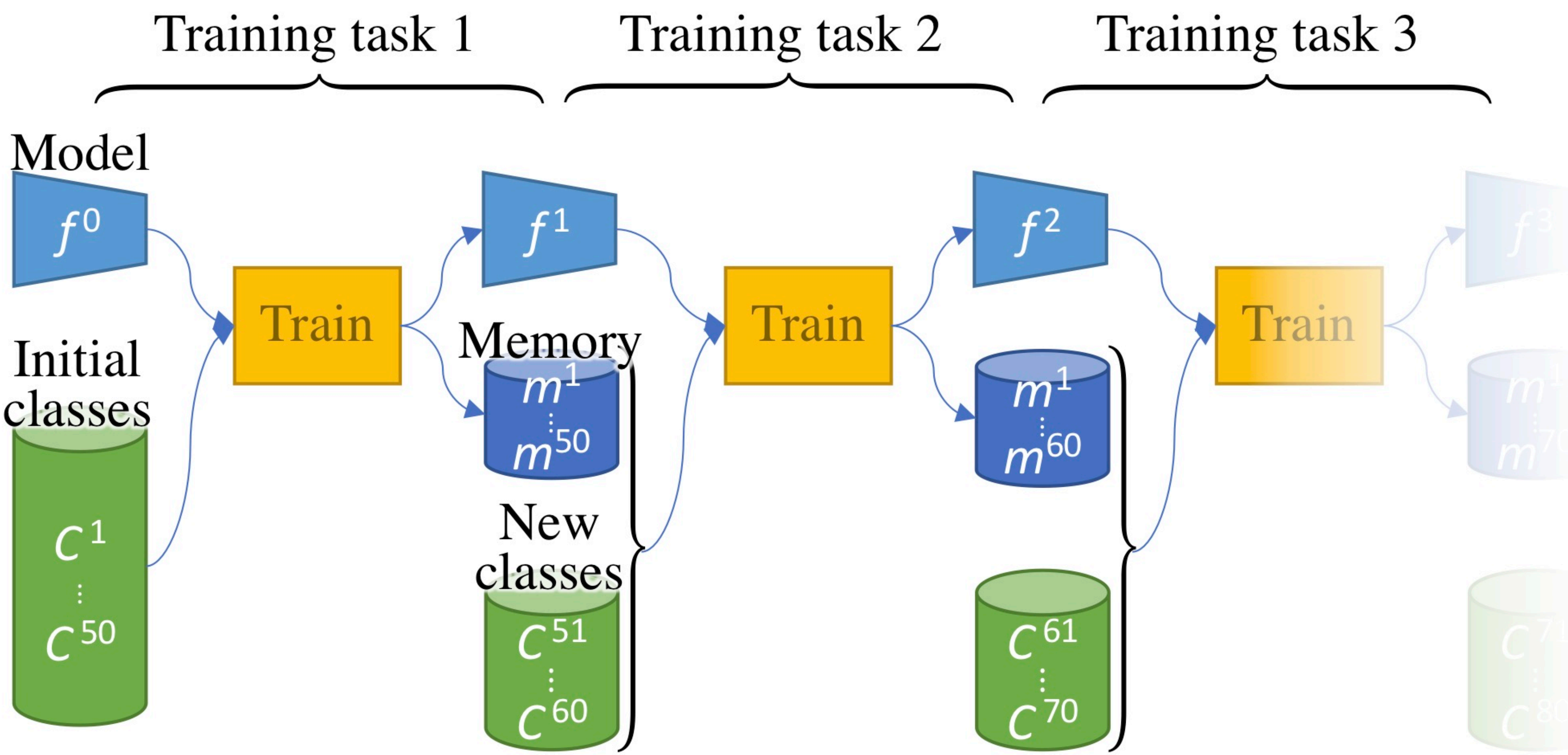
Soon to be released…

Failure Case of Pseudo-Labeling

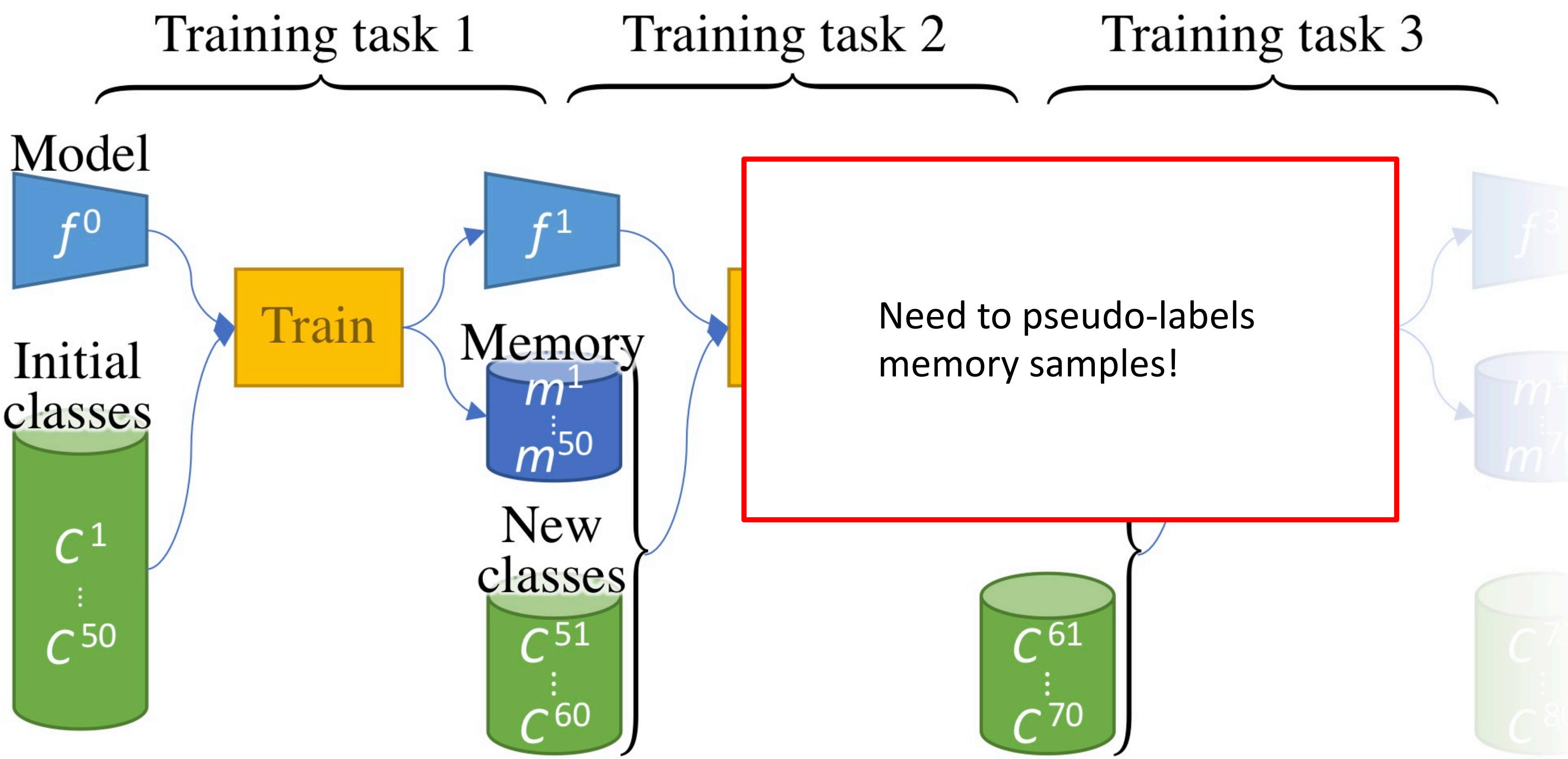
How to pseudo-labels,
when there is nothing to pseudo-labels?



Rehearsal Learning

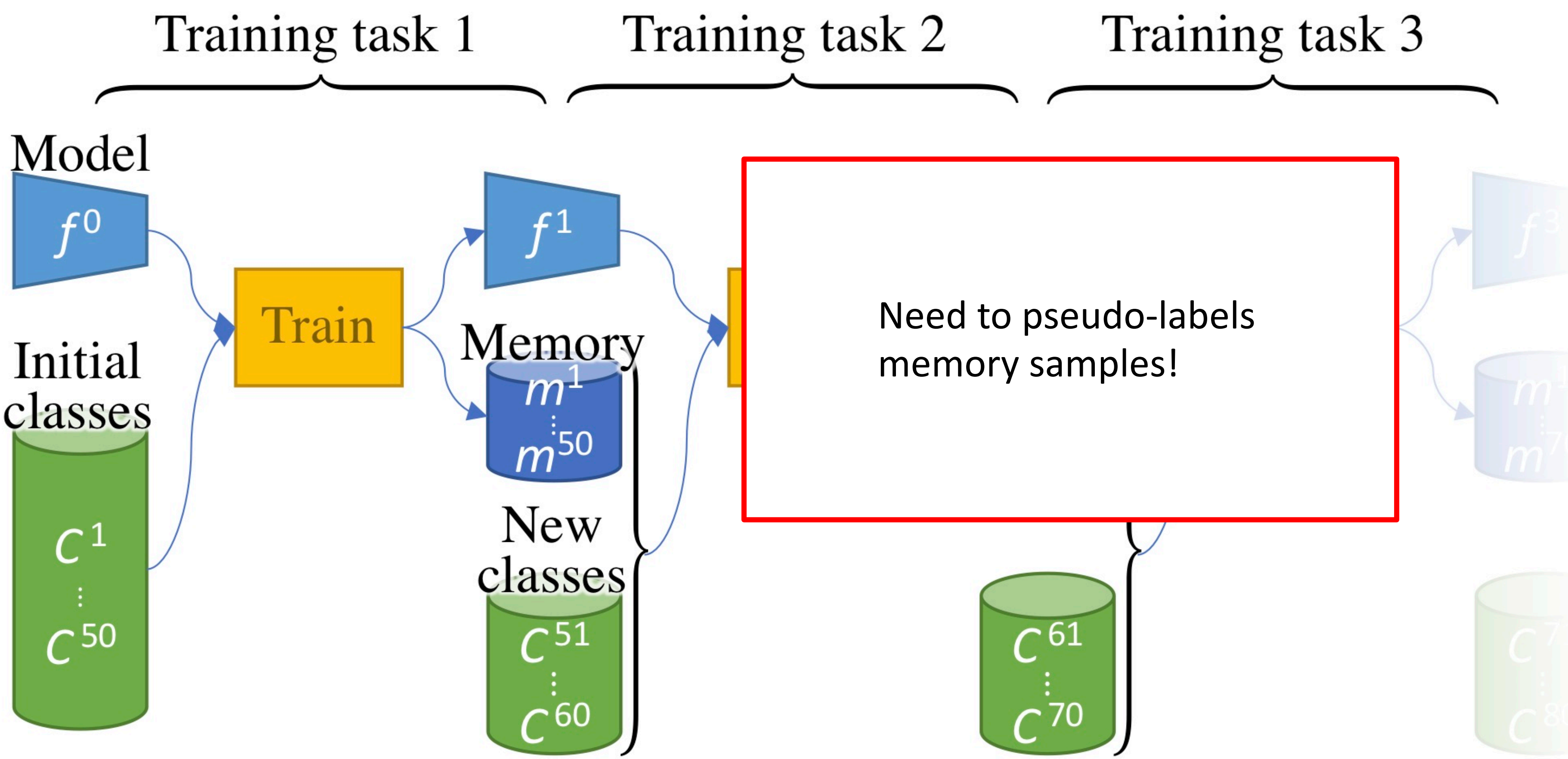


Rehearsal Learning



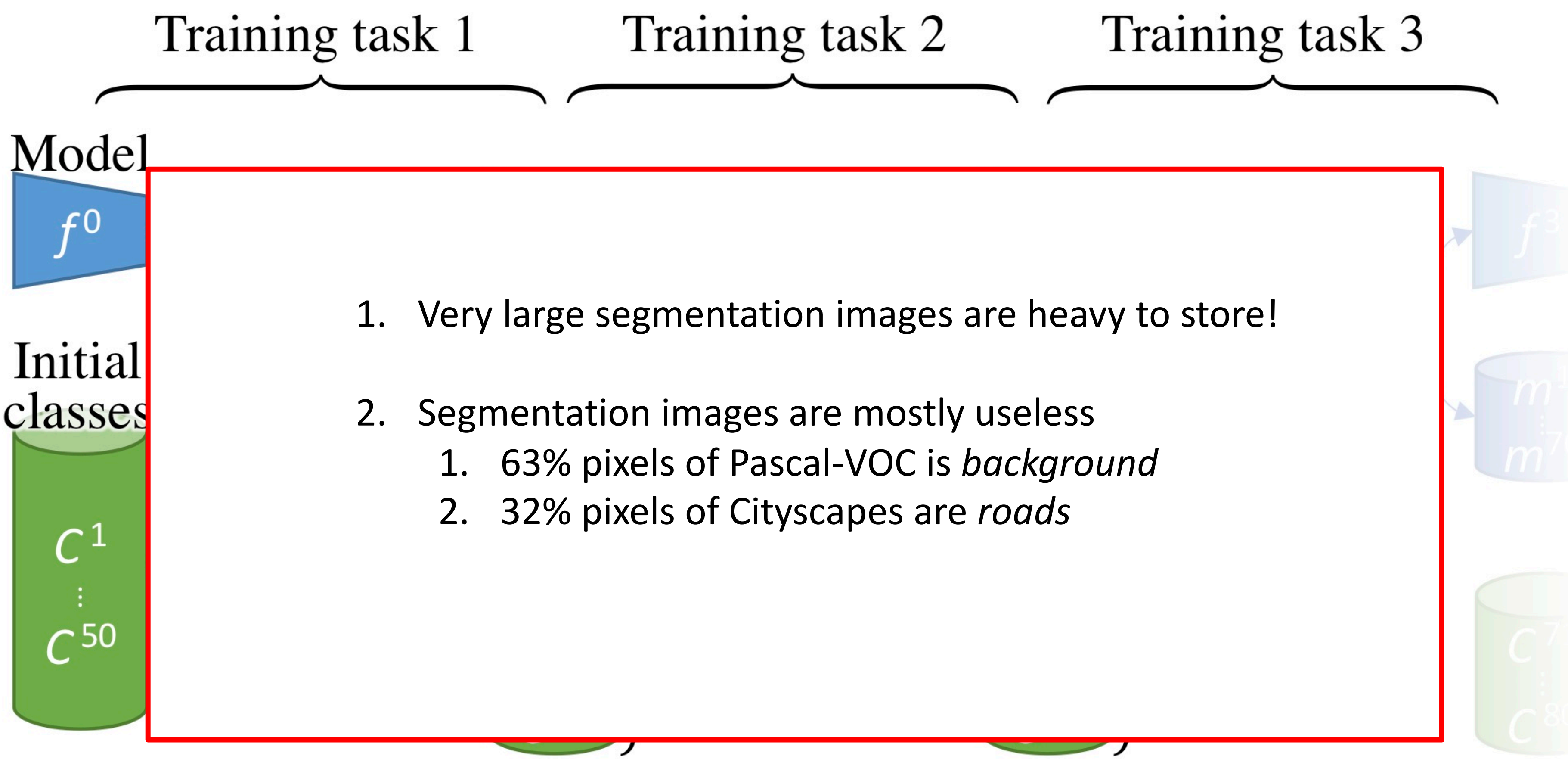
Need to pseudo-labels memory samples!

Rehearsal Learning



Need to pseudo-labels memory samples!

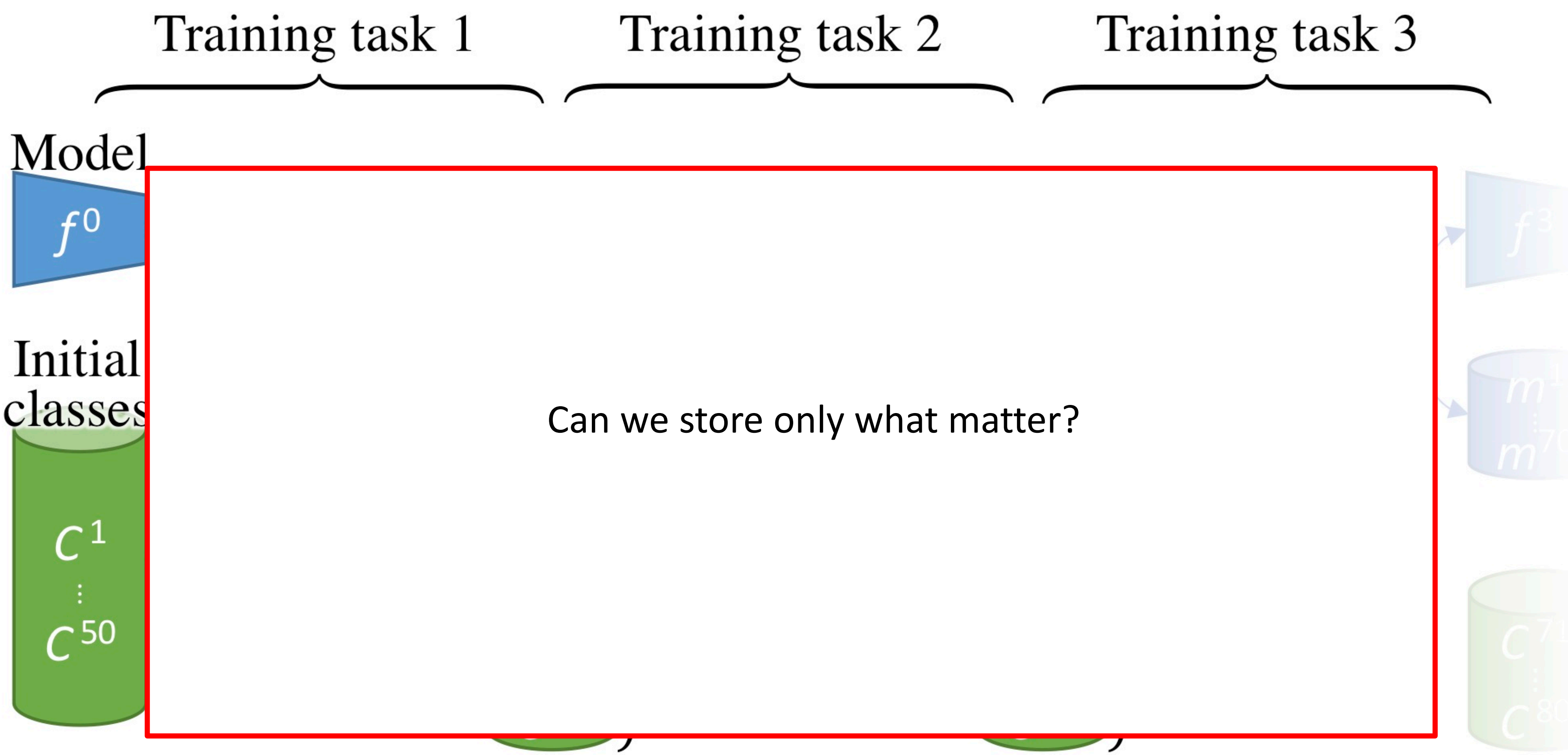
Rehearsal Learning



1. Very large segmentation images are heavy to store!

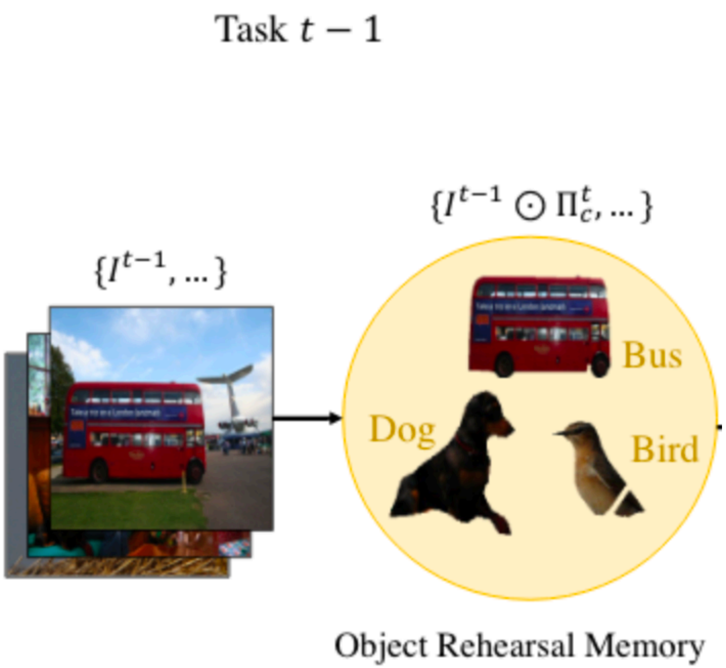
2. Segmentation images are mostly useless
    1. 63% pixels of Pascal-VOC is *background*
    2. 32% pixels of Cityscapes are *roads*

Rehearsal Learning



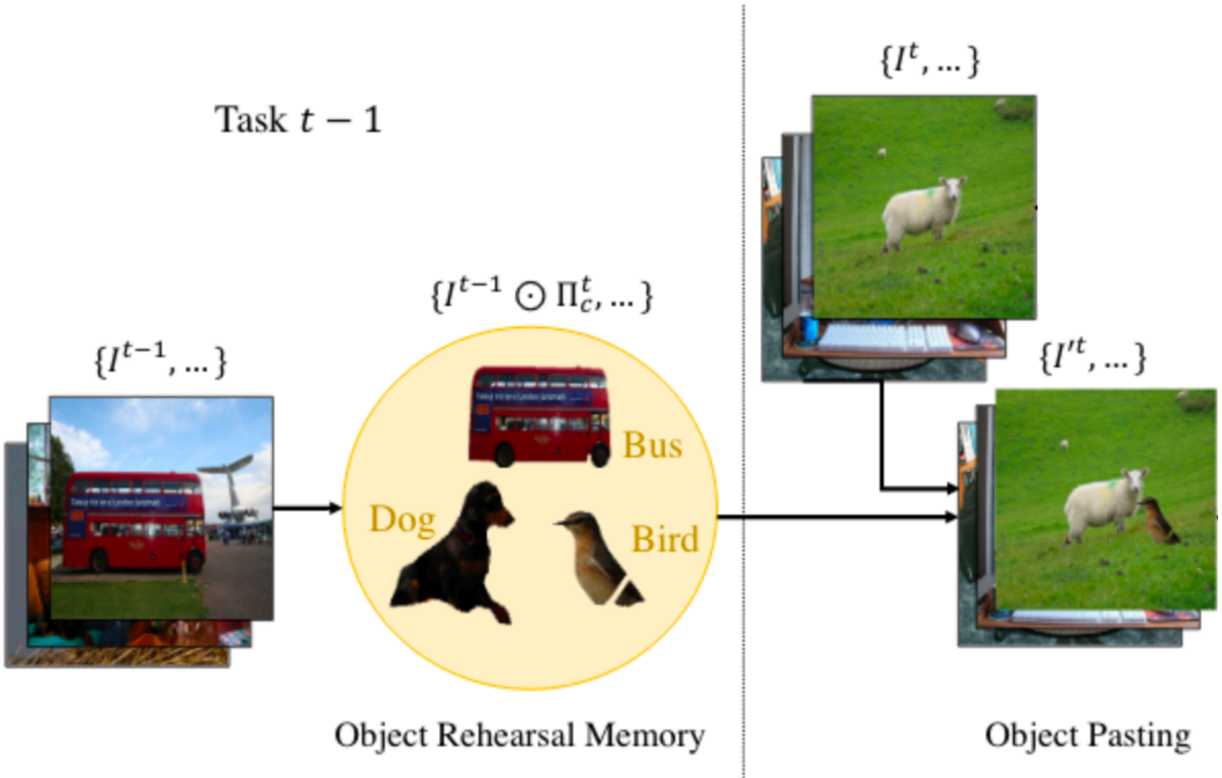
Can we store only what matter?

Object Learning



Storing only the objects

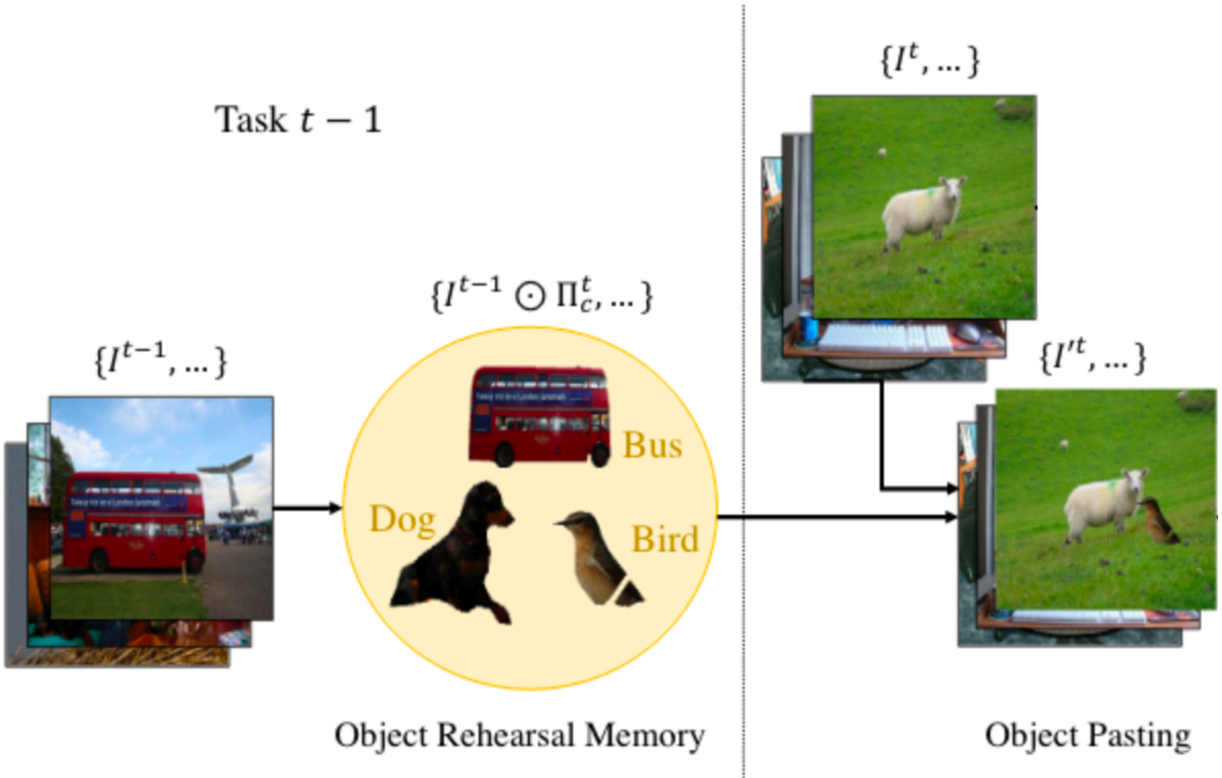
Object Learning



Pasting into current task images

Object Learning

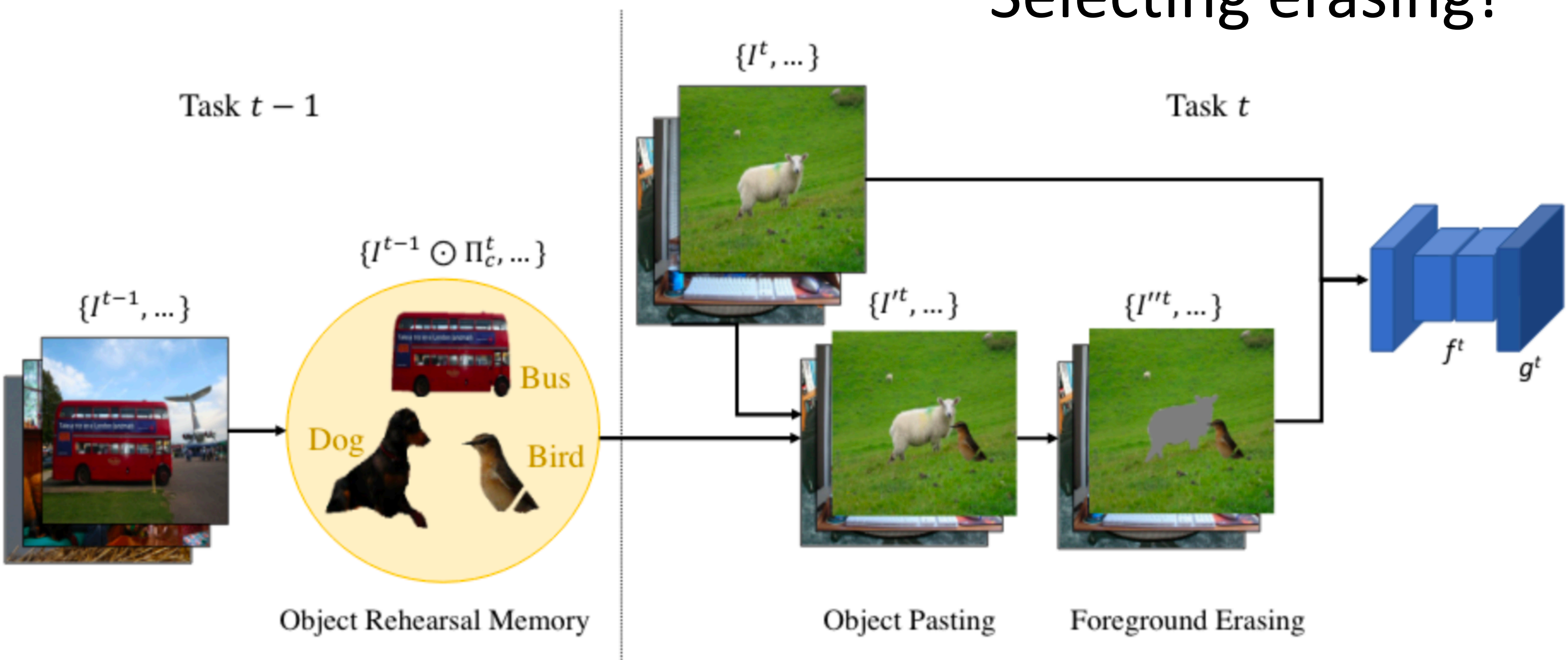
heuritech SCIENCES SORBONNE UNIVERSITÉ



Interference!

Object Learning

Selecting erasing!



Object Learning

Method	Rehearsal	Memory (Mb) ↓	Time (Hours) ↓	15-1 (6 tasks) 0-15	16-20	*all*	*avg*
PLOP	—	0	1.8	65.12	21.11	54.64	67.21
PLOPLong	—	0	1.8	72.00	26.66	61.20	70.02

Object Learning

Method	Rehearsal	Memory (Mb) ↓	Time (Hours) ↓	0-15	16-20	*all*	*avg*
			15-1 (6 tasks)				
PLOP	—	0	1.8	65.12	21.11	54.64	67.21
PLOPLong	—	0	1.8	72.00	26.66	61.20	70.02
Yu et al. [74]	Unlabeled COCO	20,000	7.0	71.40	40.00	63.60	
PLOP	Unlabeled COCO	20,000	1.4	72.57	45.08	66.03	71.85
PLOP	Unlabeled VOC	2,000	1.4	75.32	52.59	69.91	75.21

Object Learning

Method	Rehearsal	**Memory** (Mb) ↓	**Time** (Hours) ↓	**15-1** (6 tasks) 0-15	16-20	*all*	*avg*
PLOP	—	0	1.8	65.12	21.11	54.64	67.21
PLOPLong	—	0	1.8	72.00	26.66	61.20	70.02
Yu et al. [74]	Unlabeled COCO	20,000	7.0	71.40	40.00	63.60	
PLOP	Unlabeled COCO	20,000	1.4	72.57	45.08	66.03	71.85
PLOP	Unlabeled VOC	2,000	1.4	75.32	52.59	69.91	75.21
PLOPLong	Partial VOC	2.2	2.6	74.14	38.87	65.74	72.02
PLOPLong	Partial VOC	22	2.6	**74.18**	43.22	66.81	**72.48**

Object Learning

heuritech SCIENCES SORBONNE UNIVERSITÉ

Method	Rehearsal	Memory (Mb) ↓	Time (Hours) ↓	0-15	16-20	*all*	*avg*
			15-1 (6 tasks)				
PLOP	—	0	1.8	65.12	21.11	54.64	67.21
PLOPLong	—	0	1.8	72.00	26.66	61.20	70.02
Yu et al. [74]	Unlabeled COCO	20,000	7.0	71.40	40.00	63.60	
PLOP	Unlabeled COCO	20,000	1.4	72.57	45.08	66.03	71.85
PLOP	Unlabeled VOC	2,000	1.4	75.32	52.59	69.91	75.21
PLOPLong	Partial VOC	2.2	2.6	74.14	38.87	65.74	72.02
PLOPLong	Partial VOC	22	2.6	**74.18**	43.22	66.81	**72.48**
PLOPLong	Object VOC	0.26	2.7	73.32	42.86	66.07	72.21
PLOPLong	Object VOC	2.6	2.7	73.79	**45.78**	**67.12**	**72.42**
Joint model	—	—	—	79.10	72.60	77.40	—

Object Learning

heuritech SCIENCES SORBONNE UNIVERSITÉ

Method	Rehearsal	Memory (Mb) ↓	Time (Hours) ↓	0-15	16-20	*all*	*avg*
			15-1 (6 tasks)				
PLOP	—	0	1.8	65.12	21.11	54.64	67.21
PLOPLong	—	0	1.8	72.00	26.66	61.20	70.02
Yu et al. [74]	Unlabeled COCO	20,000	7.0	71.40	40.00	63.60	
PLOP	Unlabeled COCO	20,000	1.4	72.57	45.08	66.03	71.85
PLOP	Unlabeled VOC	2,000	1.4	75.32	52.59	69.91	75.21
PLOPLong	Partial VOC	2.2	2.6	74.14	38.87	65.74	72.02
PLOPLong	Partial VOC	22	2.6	**74.18**	43.22	66.81	**72.48**
PLOPLong	Object VOC	0.26	2.7	73.32	42.86	66.07	72.21
PLOPLong	Object VOC	2.6	2.7	73.79	**45.78**	**67.12**	**72.42**
Joint model	—	—	—	79.10	72.60	77.40	—

Object Learning

heuritech SCIENCES SORBONNE UNIVERSITÉ

Type	Mixing	Erase	Memory ↓	**15-1** (6 tasks) *all*	*avg*	
Image	Mixup	—	22.20	61.77	69.88	I
	—	—		66.81	**72.48**	II
Patch	Pasting	All	4.50	55.45	66.35	III
	Pasting	—		63.41	70.75	IV
	Pasting	Foreground		66.28	71.66	V
Object	Mixup	—	**2.60**	63.25	70.91	VI
	Mixup	Foreground		64.45	71.65	VII
	Pasting	All		52.26	65.97	VIII
	Pasting	—		63.12	70.52	IX
	Pasting	Foreground		**67.12**	**72.42**	X

What are your questions?