heuritech

SCIENCES SORBONNE UNIVERSITÉ

LEARNING CONTINUOUSLY WITHOUT FORGETTING FOR CONTINUAL SEMANTIC SEGMENTATION

CVPR 2021

Arthur Douillard
Yifu Chen
Arnaud Dapogny
Matthieu Cord

LIP6

MLIA Machine Learning & Deep Learning for Information Access

What is Continual Learning?

What

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

f

Train once

Evaluate on a fixed test set

…

What

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

f

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

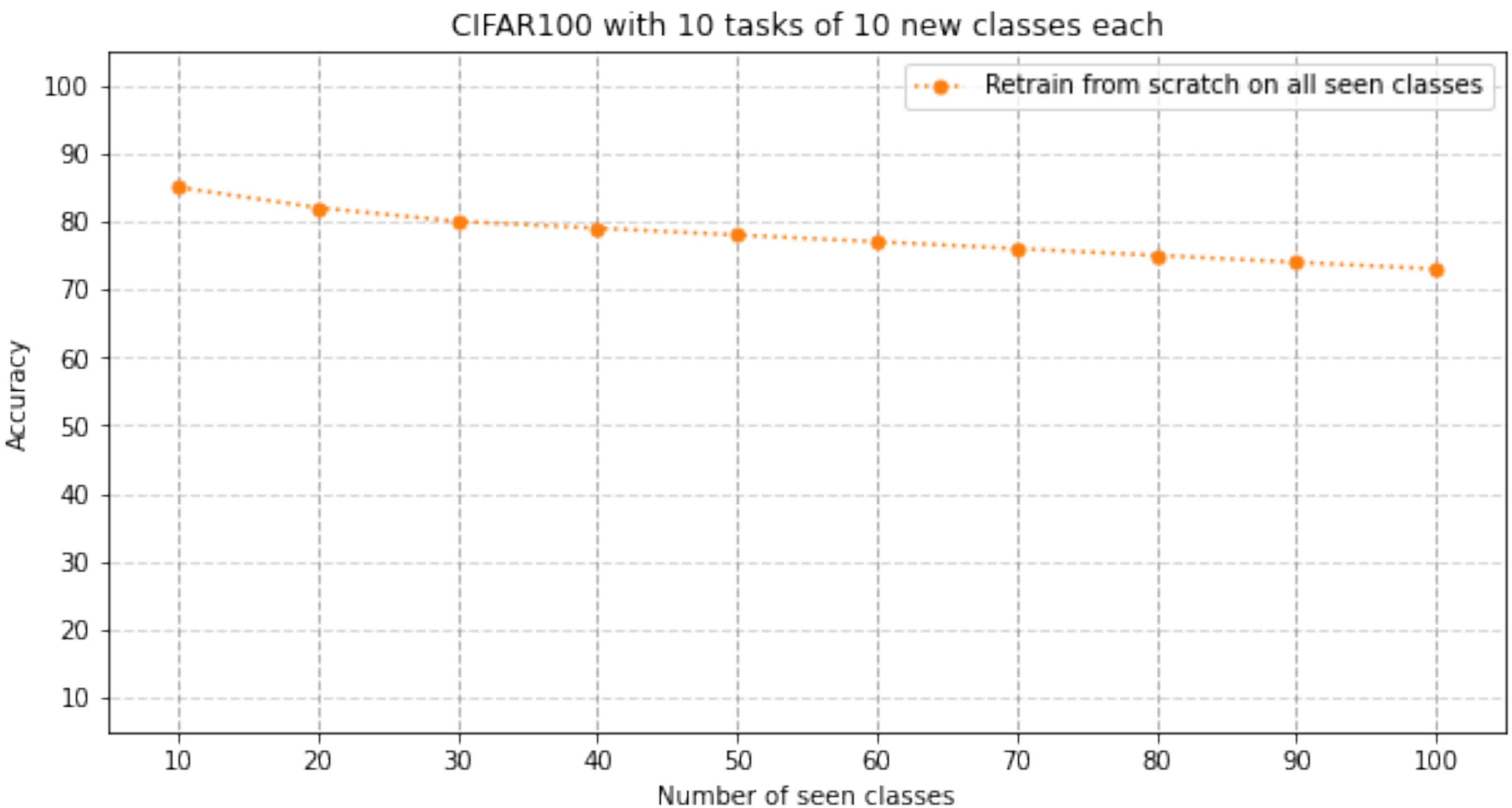
Final Evaluation:

Single {dog, cat, boat, plane} ?

Multi {dog, cat} ?

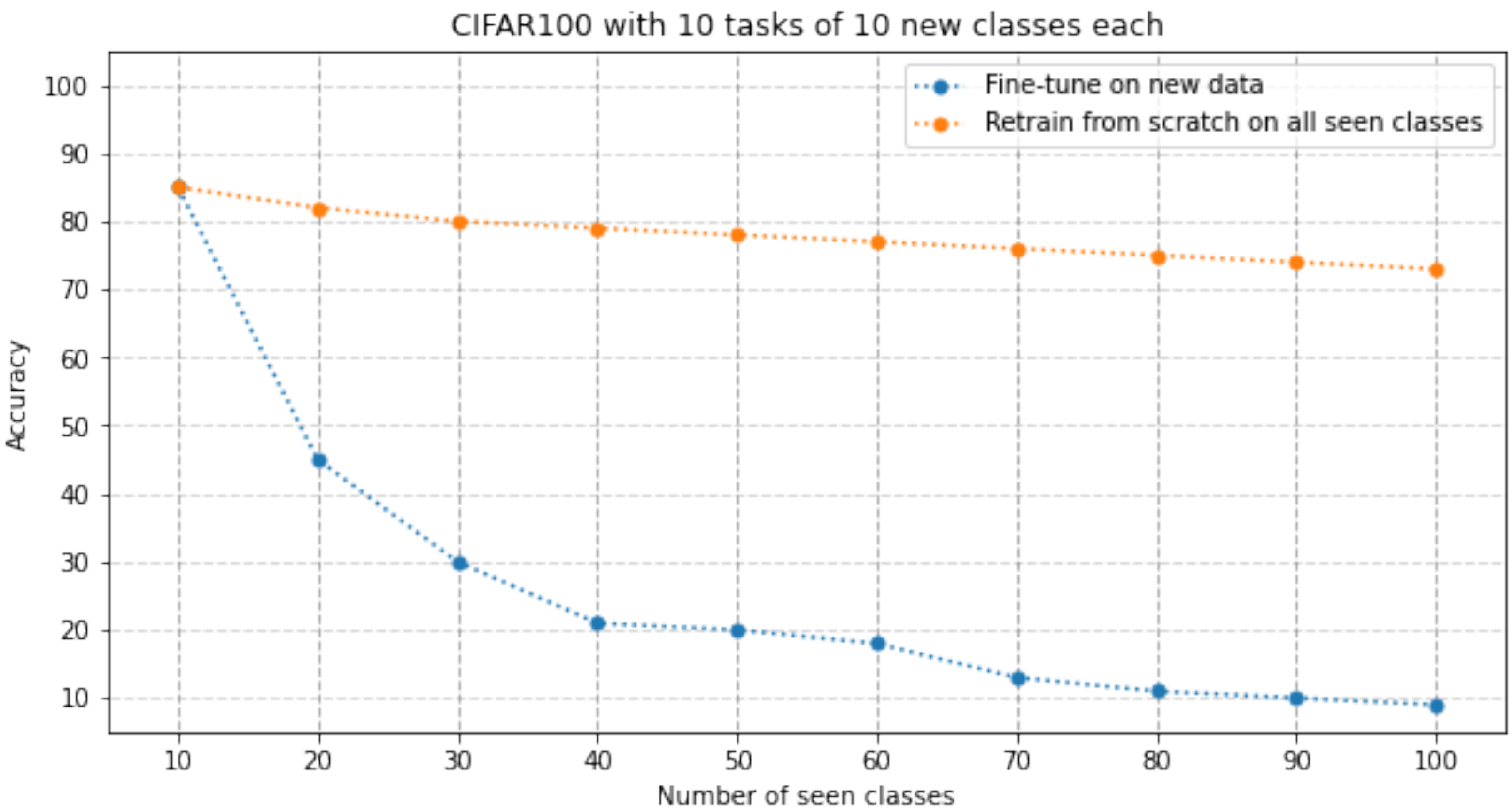
[14]: Chaudhry et al., Riemannian Walk for Incremental Learning: Understanding Forgetting and Intransigence, 2018

Example

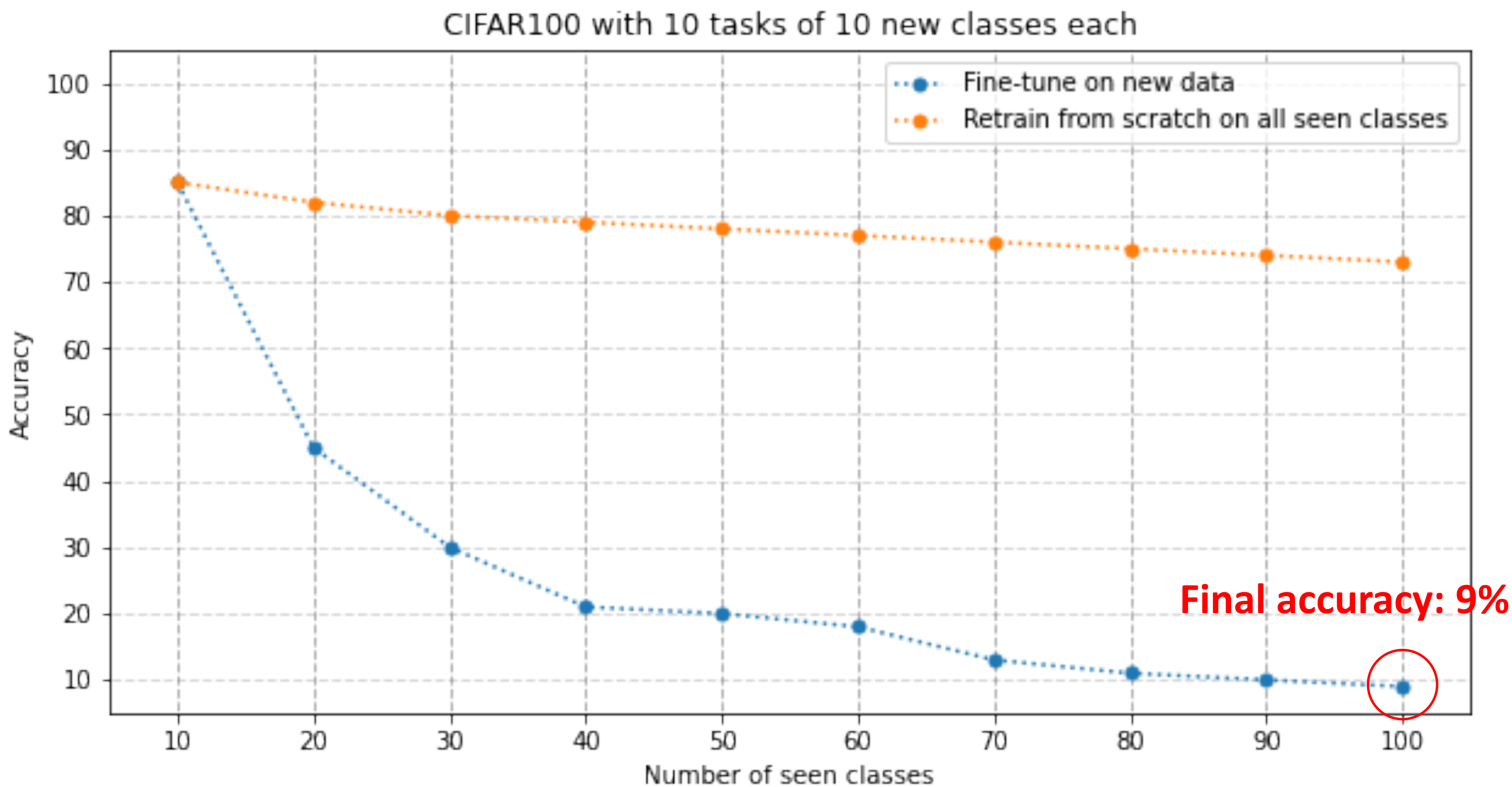

CIFAR100 with 10 tasks of 10 new classes each
Retrain from scratch on all seen classes
Accuracy
100
90
80
70
60
50
40
30
20
10
10
20
30
40
50
60
70
80
90
100
Number of seen classes

Example

heuritech SCIENCES SORBONNE UNIVERSITÉ

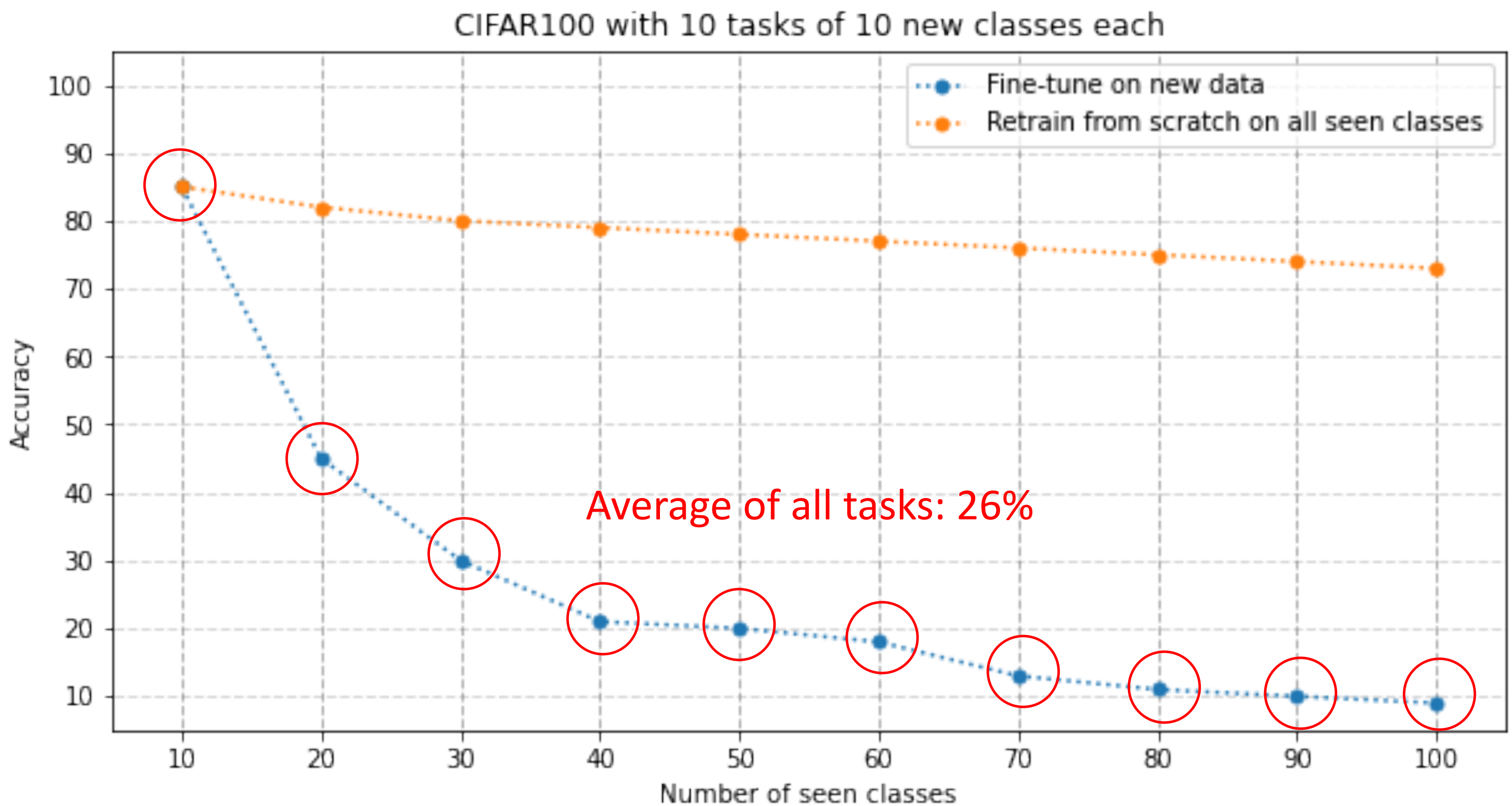


Example

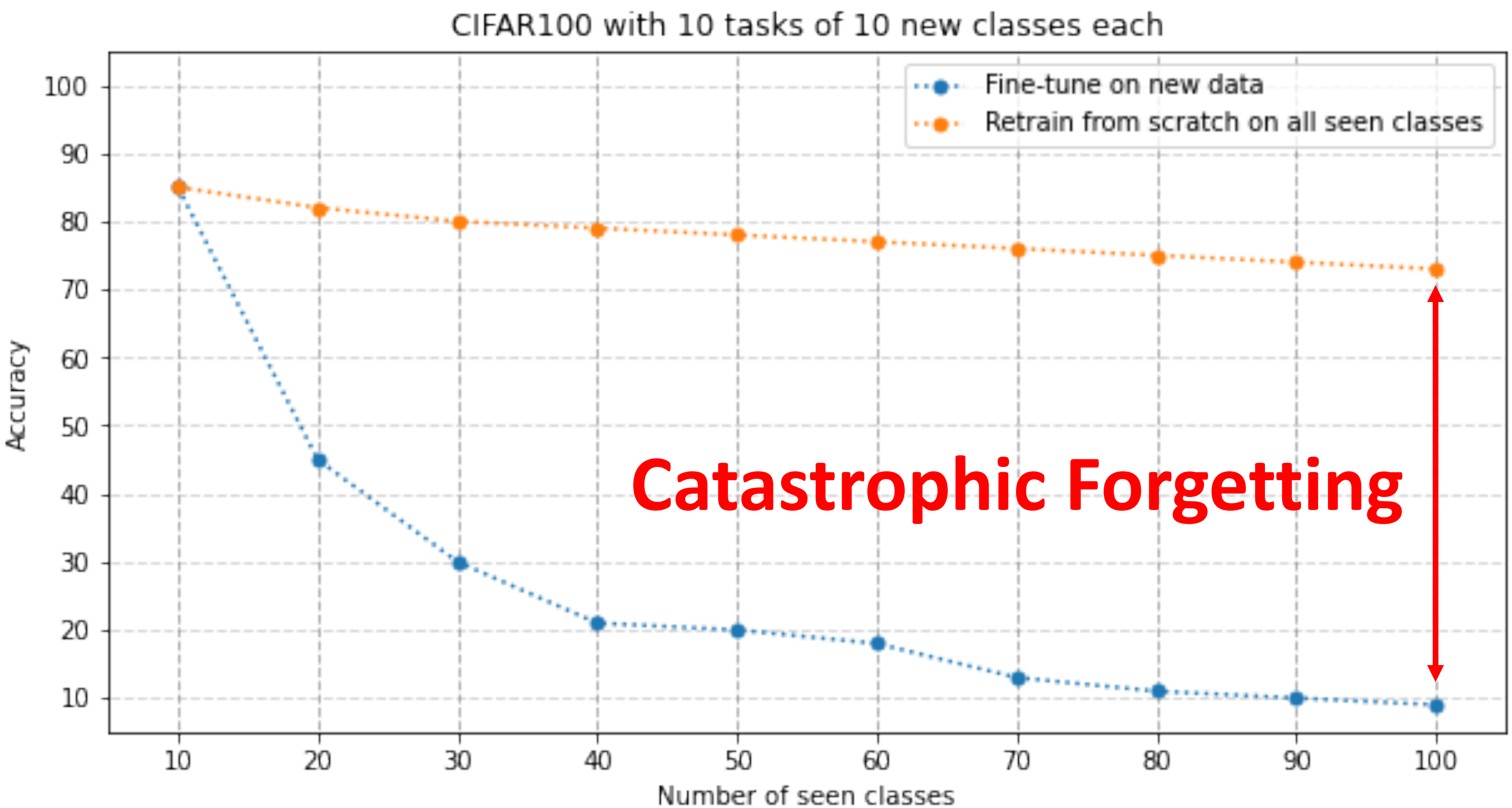

CIFAR100 with 10 tasks of 10 new classes each
Fine-tune on new data
Retrain from scratch on all seen classes
Accuracy
100
90
80
70
60
50
40
30
20
10
10 20 30 40 50 60 70 80 90 100
Number of seen classes
Final accuracy: 9%

Example

heuritech SCIENCES SORBONNE UNIVERSITÉ



Example


CIFAR100 with 10 tasks of 10 new classes each
Fine-tune on new data
Retrain from scratch on all seen classes
Catastrophic Forgetting
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

How to Solve it?

Broad Strategies

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

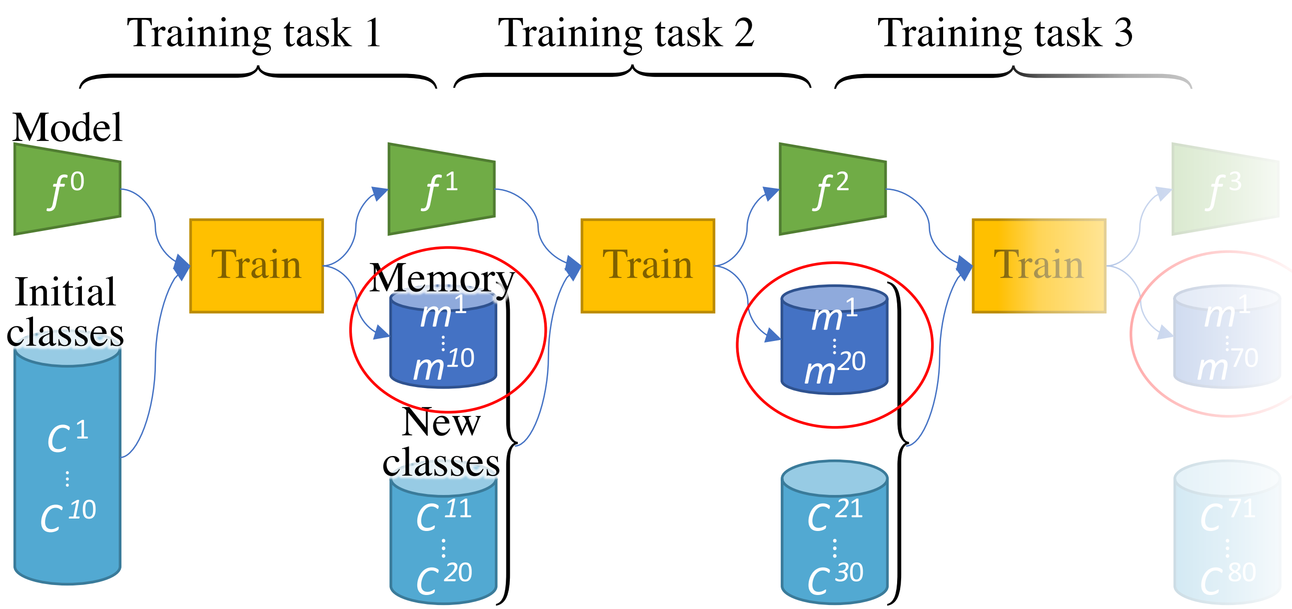
Broad Strategies

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

1. Rehearsal

Replay a limited amount of previous data

e.g. iCaRL [3]



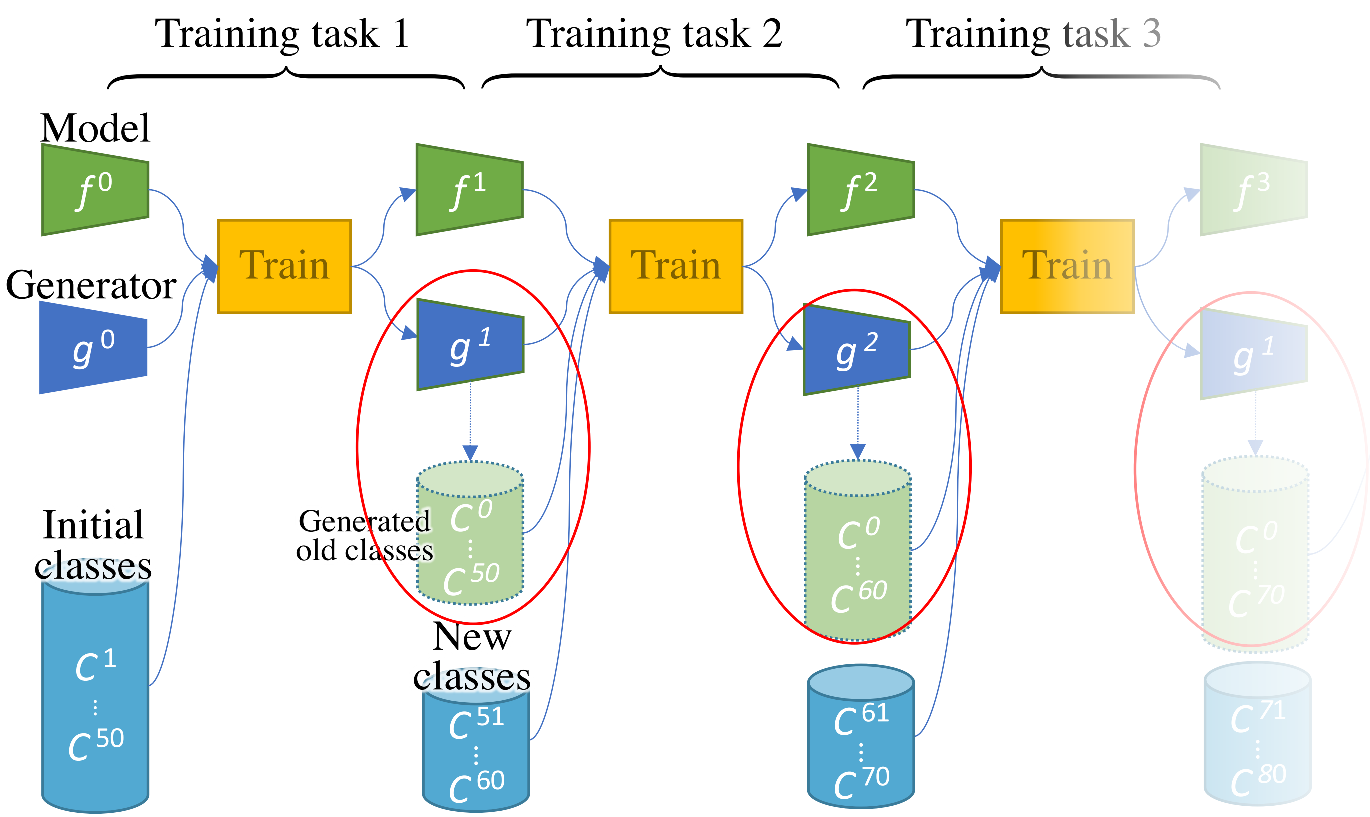
[3]: Rebuffi et al., iCaRL: Incremental Classifier and Representation Learning, 2017

1. Rehearsal

heuritech SCIENCES SORBONNE UNIVERSITÉ

Generate a limited amount of previous data

e.g. DGR [15]



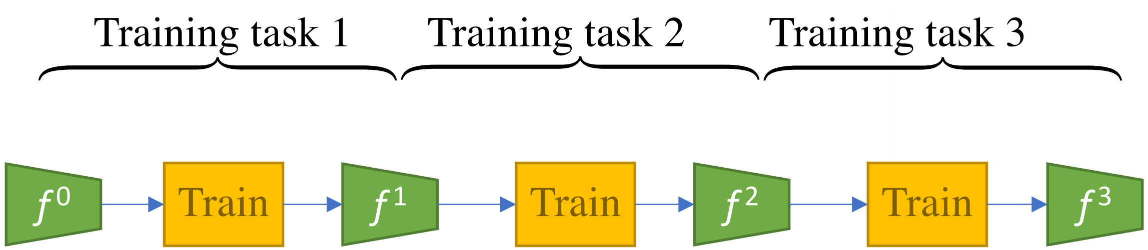
[15]: Shin et al., Continual Learning with Deep Generative Replay, 2017

Broad Strategies

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

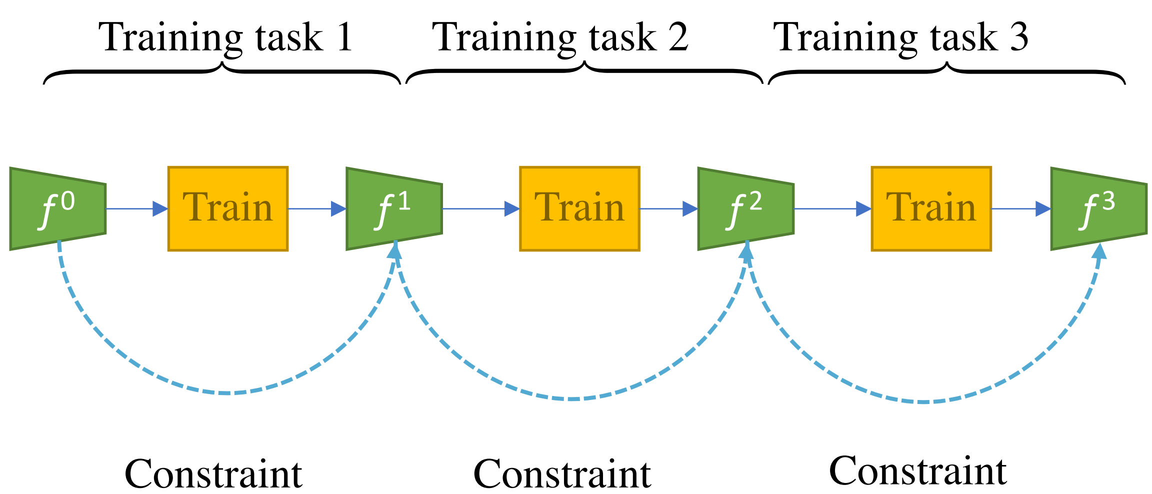
2. Constraints

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



2. Constraints

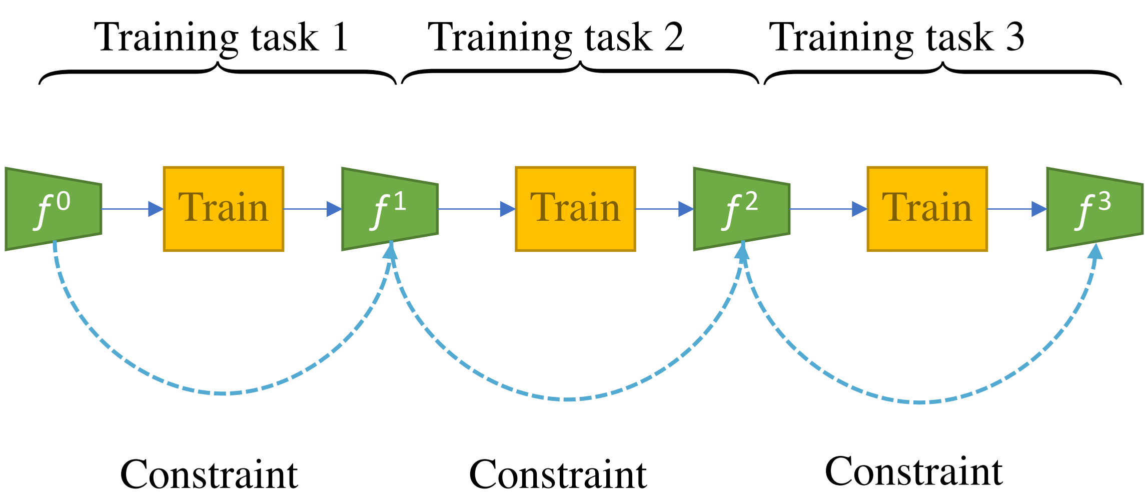
Constraints between f^{t-1} and f^{t}:



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. **Architecture**
4. Classifier Correction

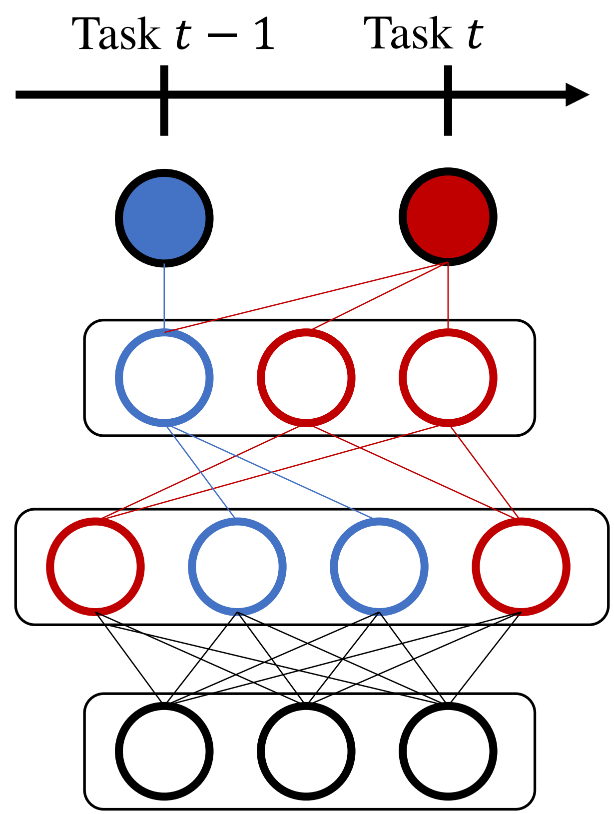
3. Architecture

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. **Classifier Correction**

4. Classifier Correction

Classifier is **biased** towards new classes

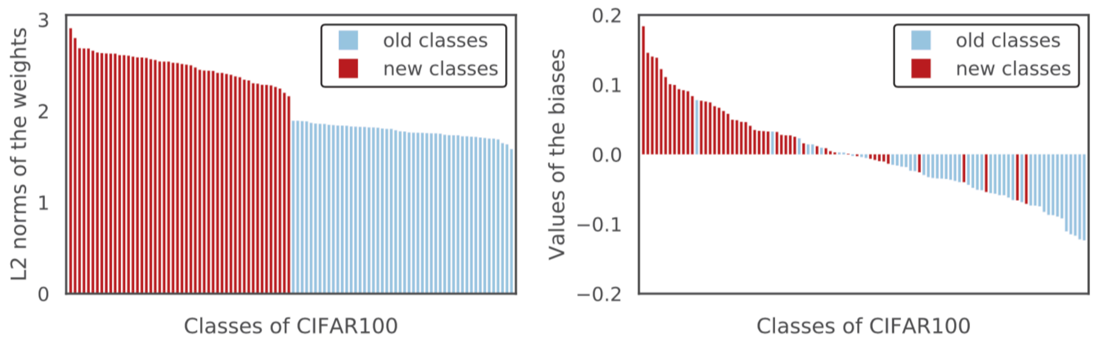
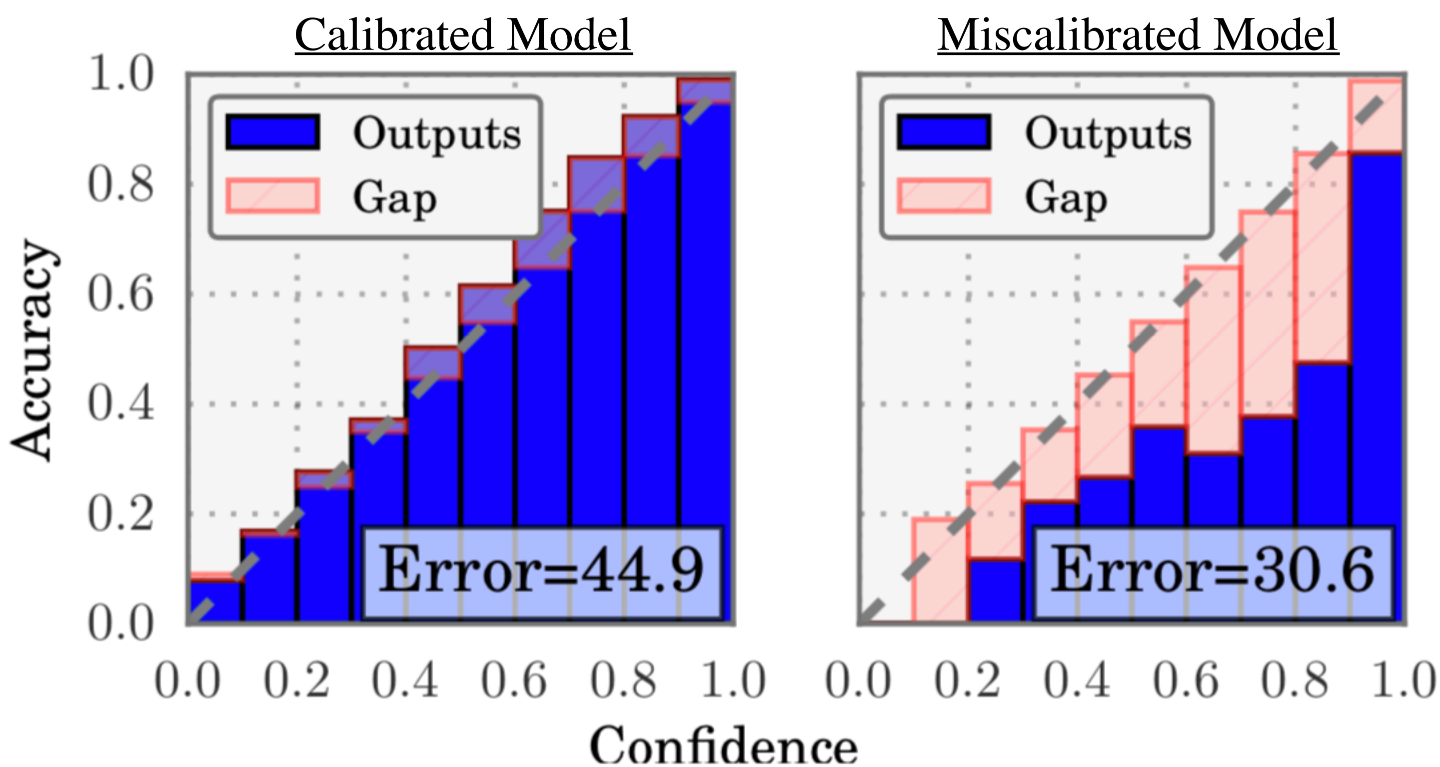


Figure 3. Visualization of the weights and biases in the last layer for old and new classes. The results come from the incremental setting of CIFAR100 (1 phase) by iCaRL [29].

4. Classifier Correction

Classifier is **biased** towards new classes

Can be recalibrated (BiC [11])

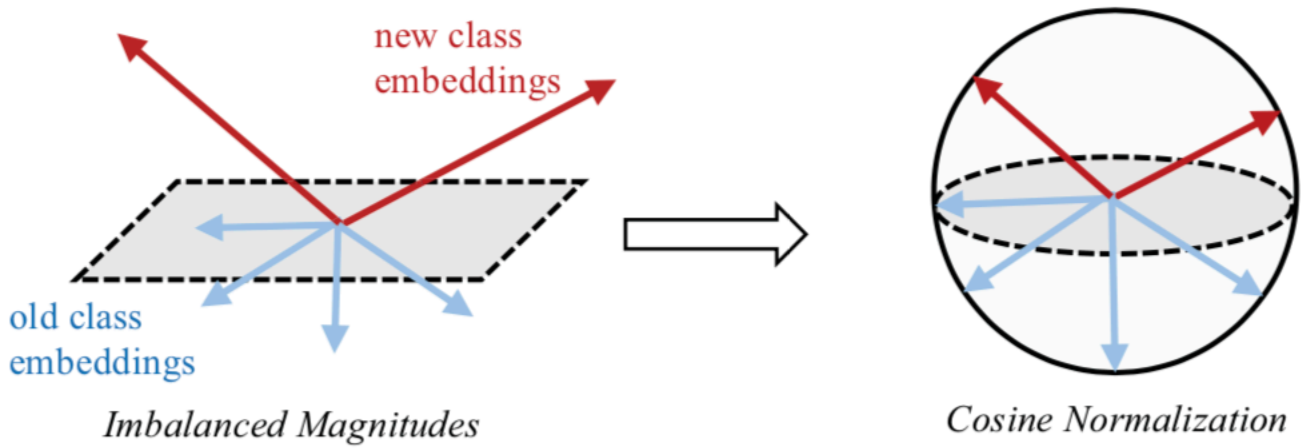


[11]: Wu et al., Large scale incremental learning, 2019

4. Classifier Correction

Classifier is **biased** towards new classes

Or normalized (LUCIR [12])



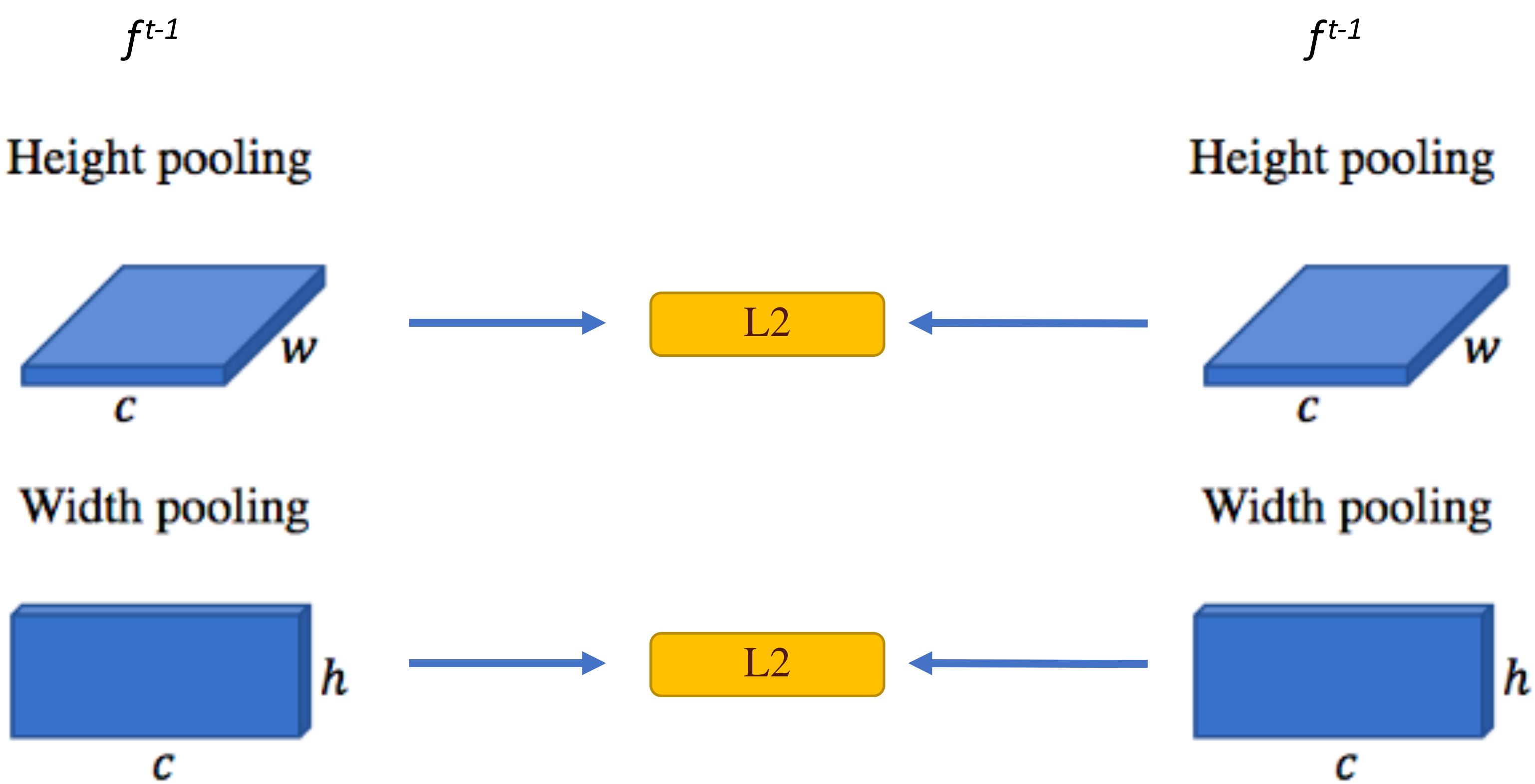
[11]: Wu et al., Large scale incremental learning, 2019

[12]: Hou et al., Learning an unified classifier incrementally via rebalancing, 2019

Previous work:

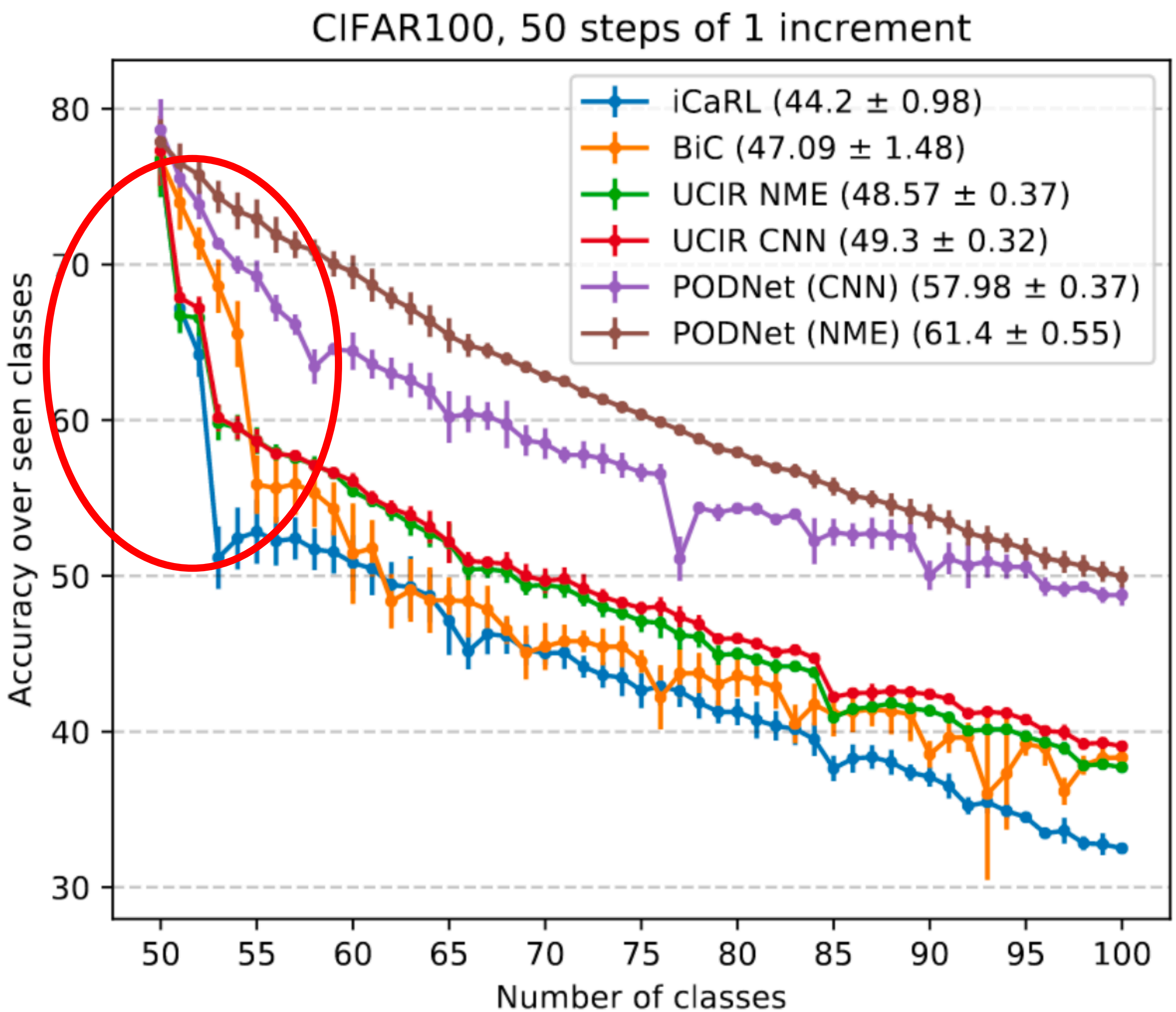
PODNet , ECCV 2020

- Multi-modal metric-based classifier
- **Multi-stage features-based distillation loss** (POD)



PODNet, ECCV 2020

Catastrophic forgetting



Good for long continual training

Learning without Forgetting for Continual Semantic Segmentation

PLOP, CVPR 2021

heuritech SCIENCES SORBONNE UNIVERSITÉ

PLOP: Learning without Forgetting for Continual Semantic Segmentation

Arthur Douillard Yifu Chen Arnaud Dapogny Matthieu Cord

Constraints + Pseudo-labeling

Segmentation

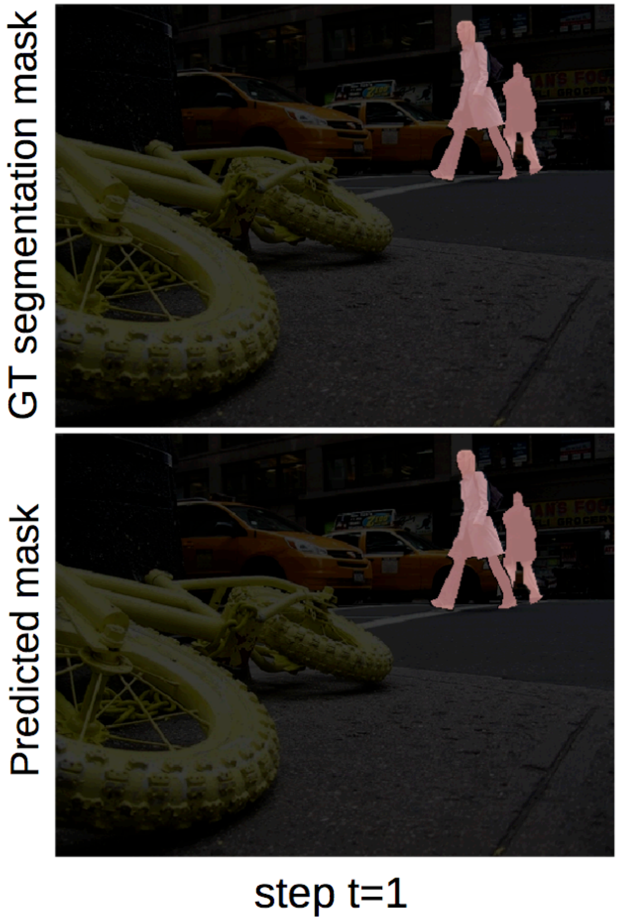
Semantic Segmentation → each pixel is labeled

Continual?

Semantic Segmentation → each pixel is labeled

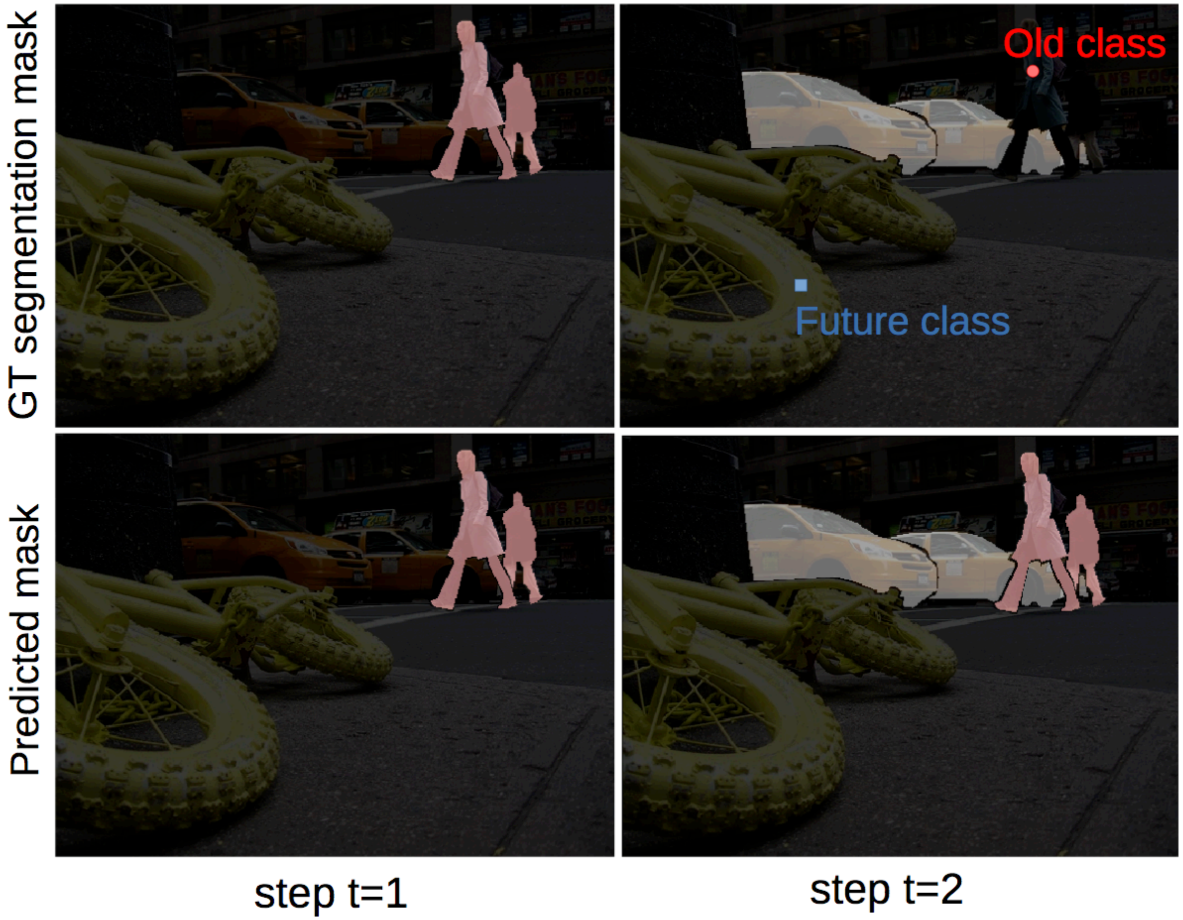
Continual Semantic Segmentation?

Background shift



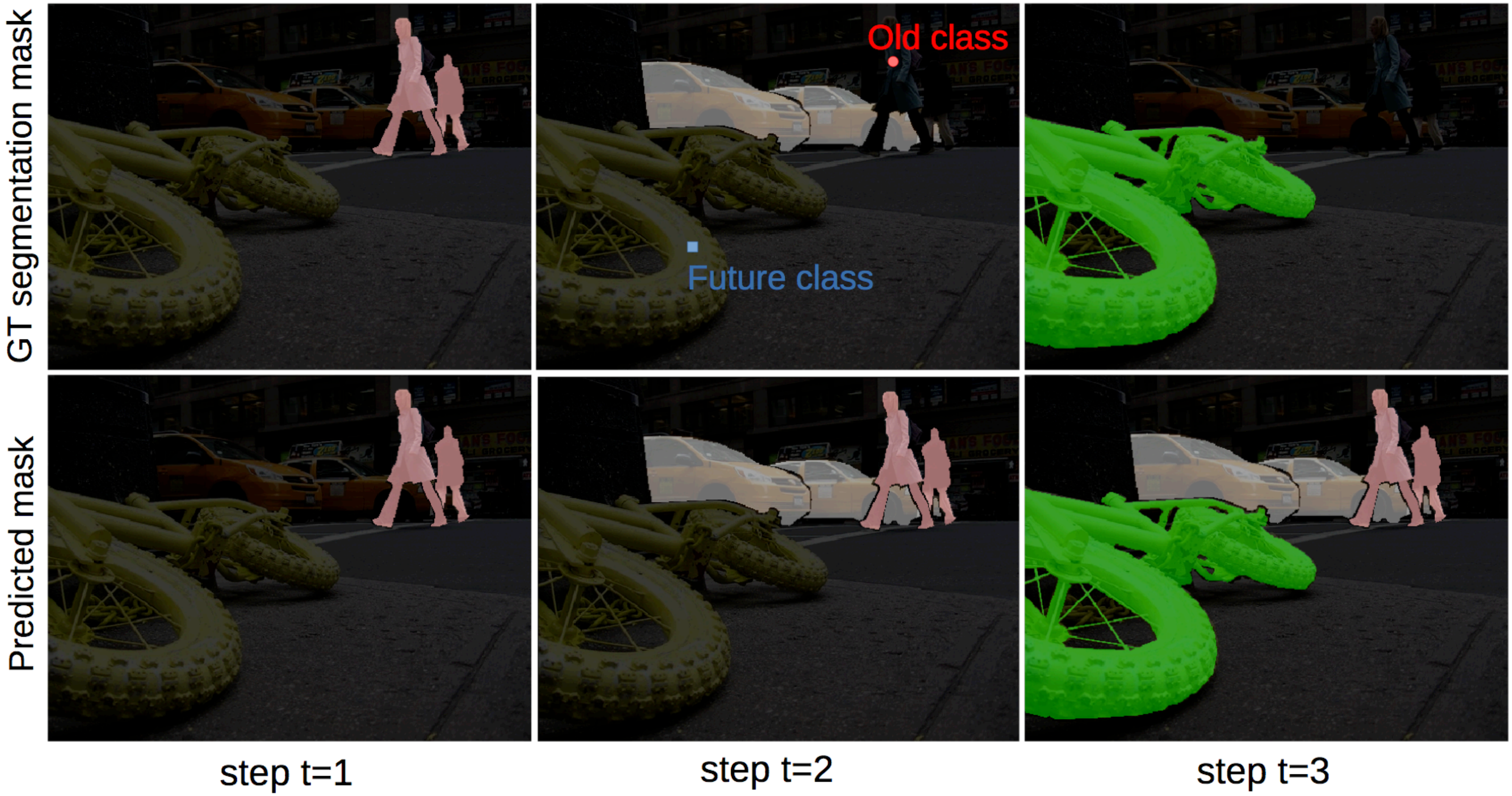
[13]: Cermelli et al., Modeling the Background for Incremental in Semantic Segmentation, 2020

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

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

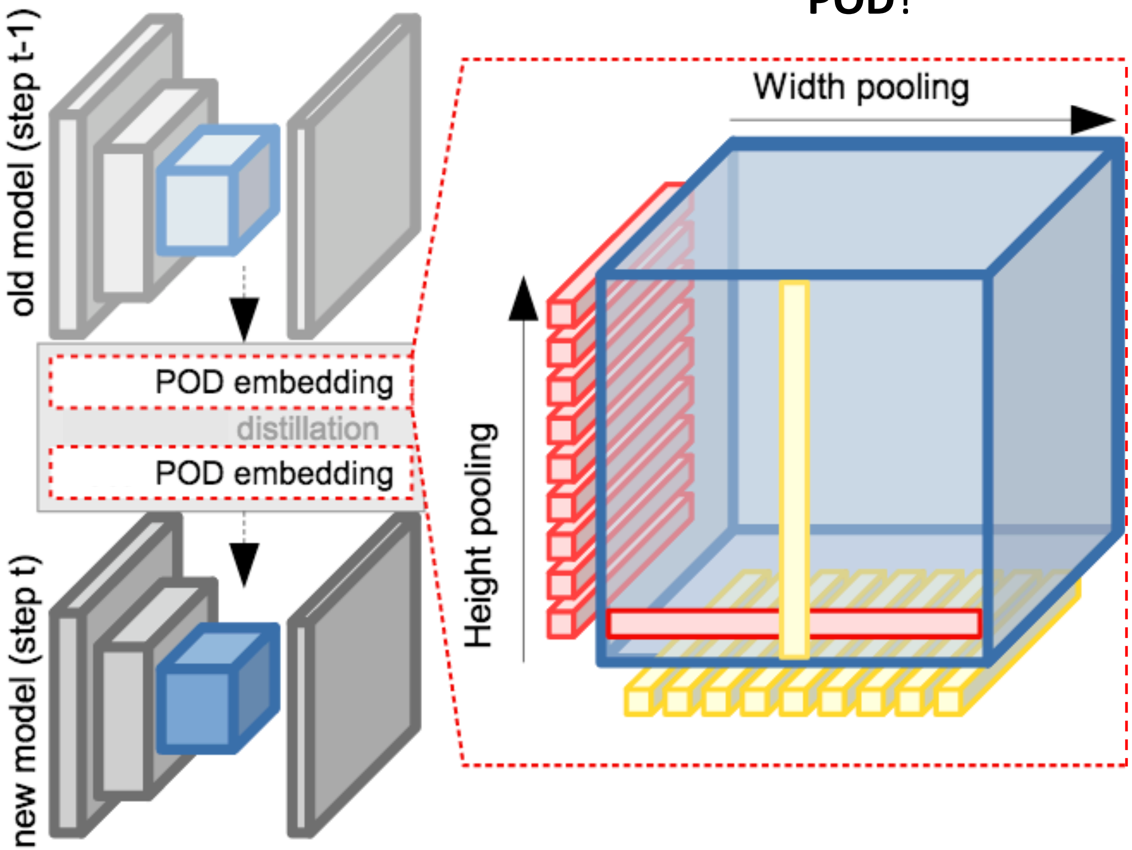
Problem 1: Forgetting

Problems:

- **<u>Forgetting is particularly strong</u>**
- Images at task t are partially labeled

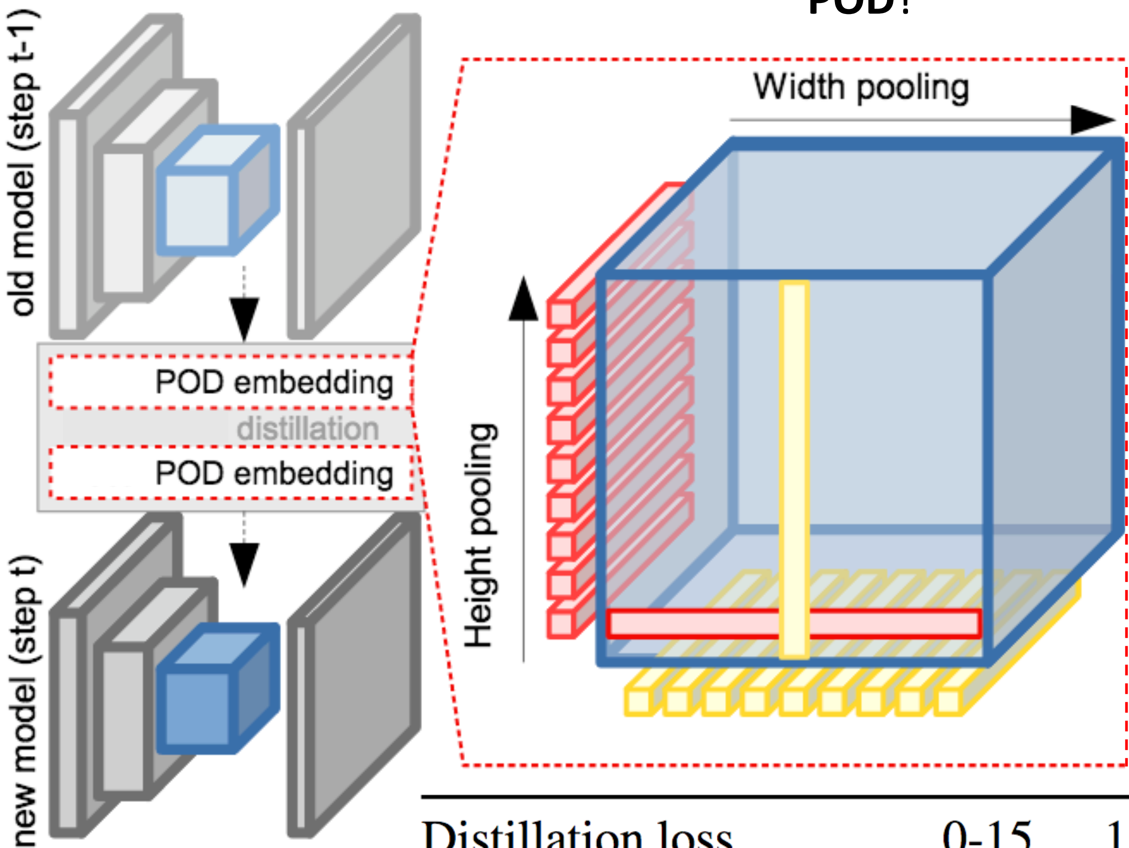
Problem 1: Forgetting

POD?



Problem 1: Forgetting

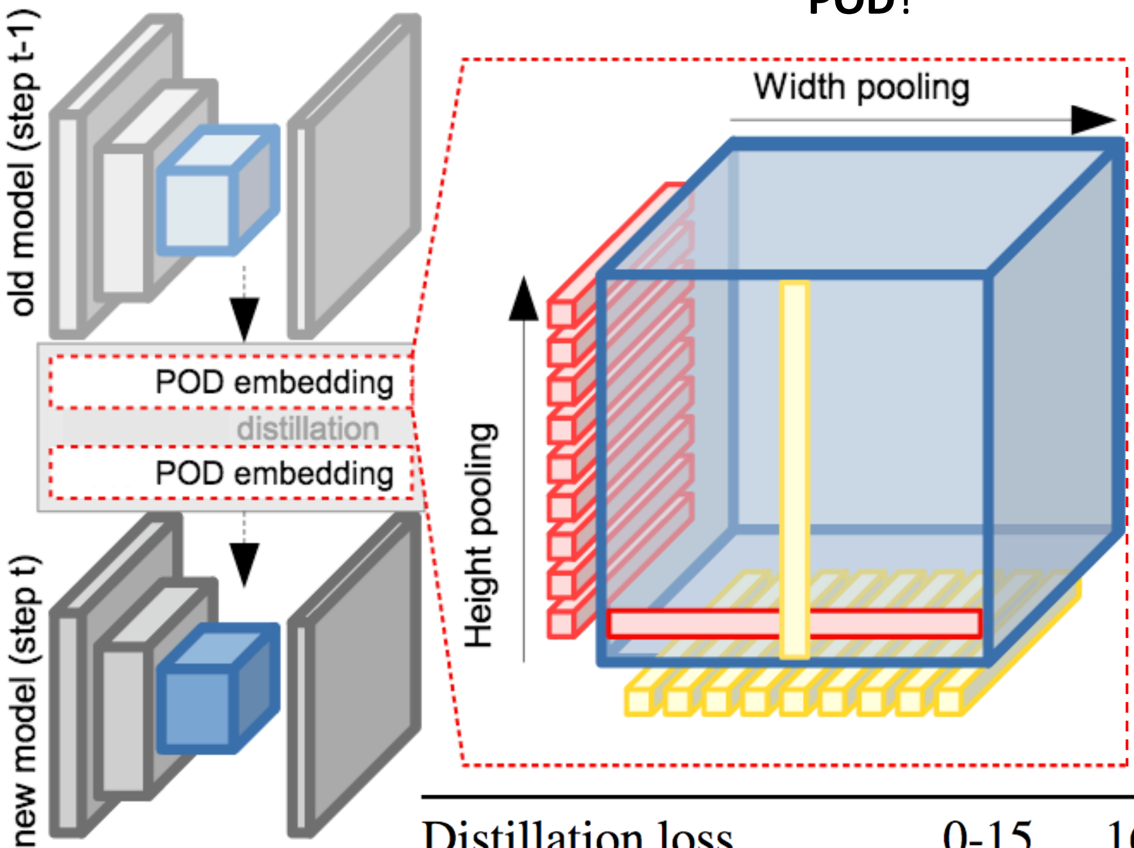
POD?



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

POD?



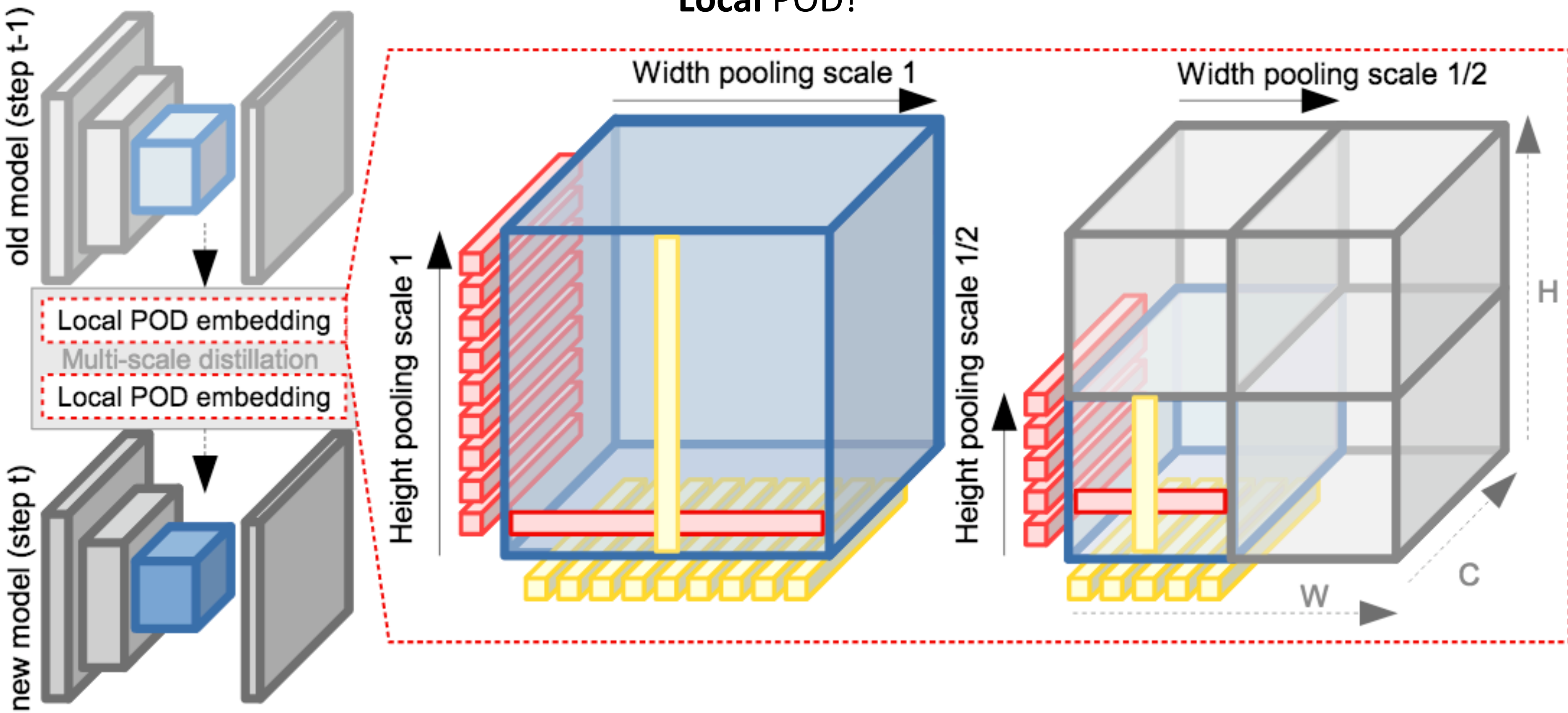
Segmentation
≠
Classification

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

heuritech SCIENCES SORBONNE UNIVERSITÉ

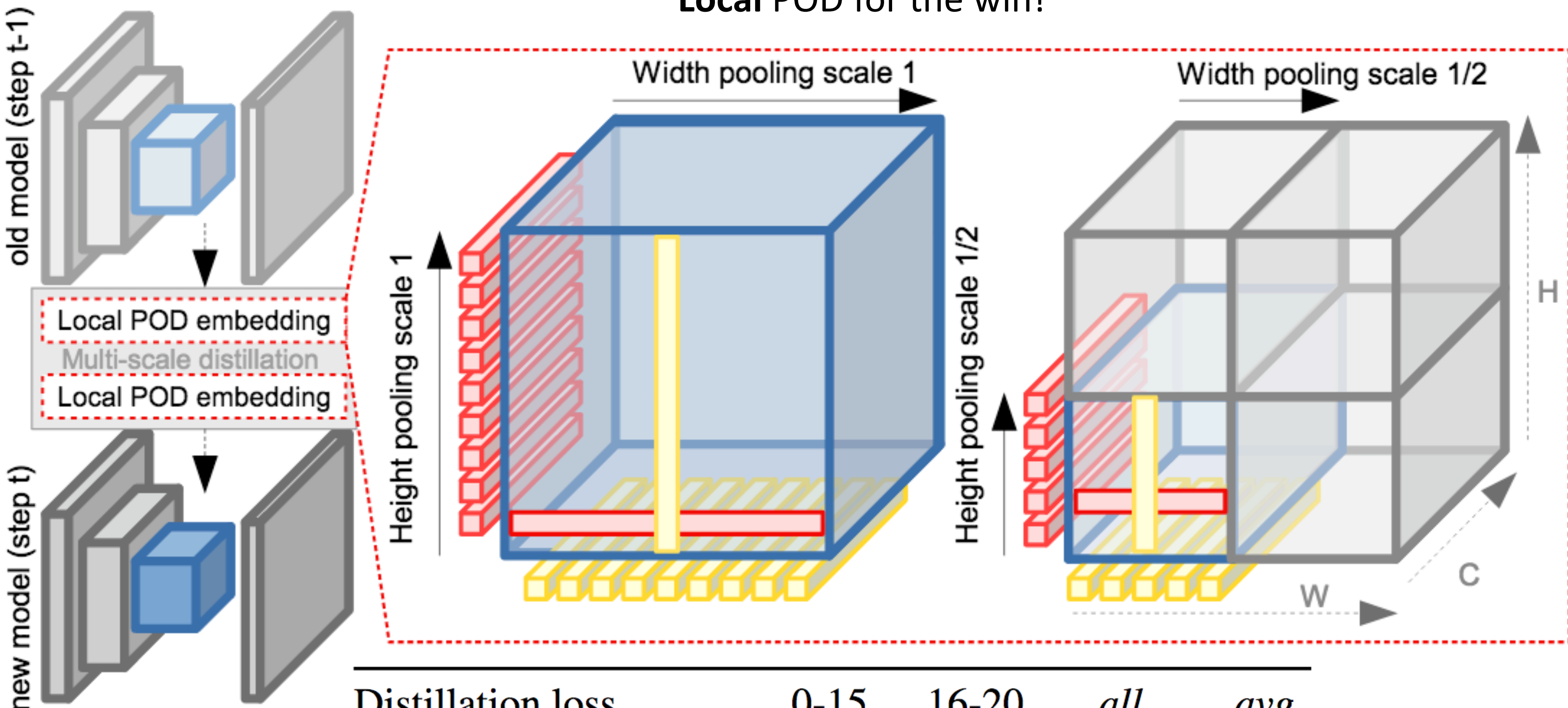
Local POD!



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

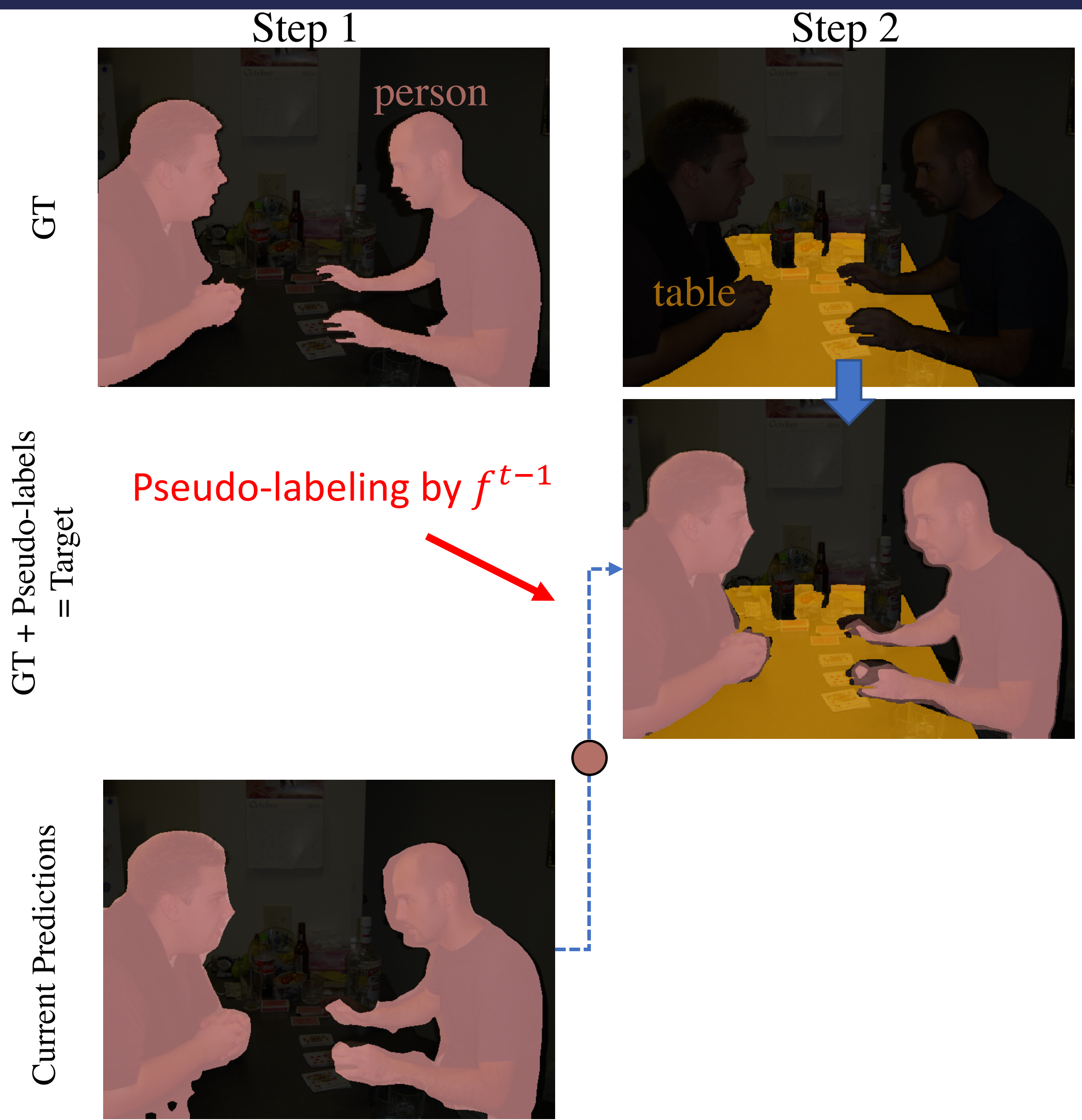
Step 1

Step 2

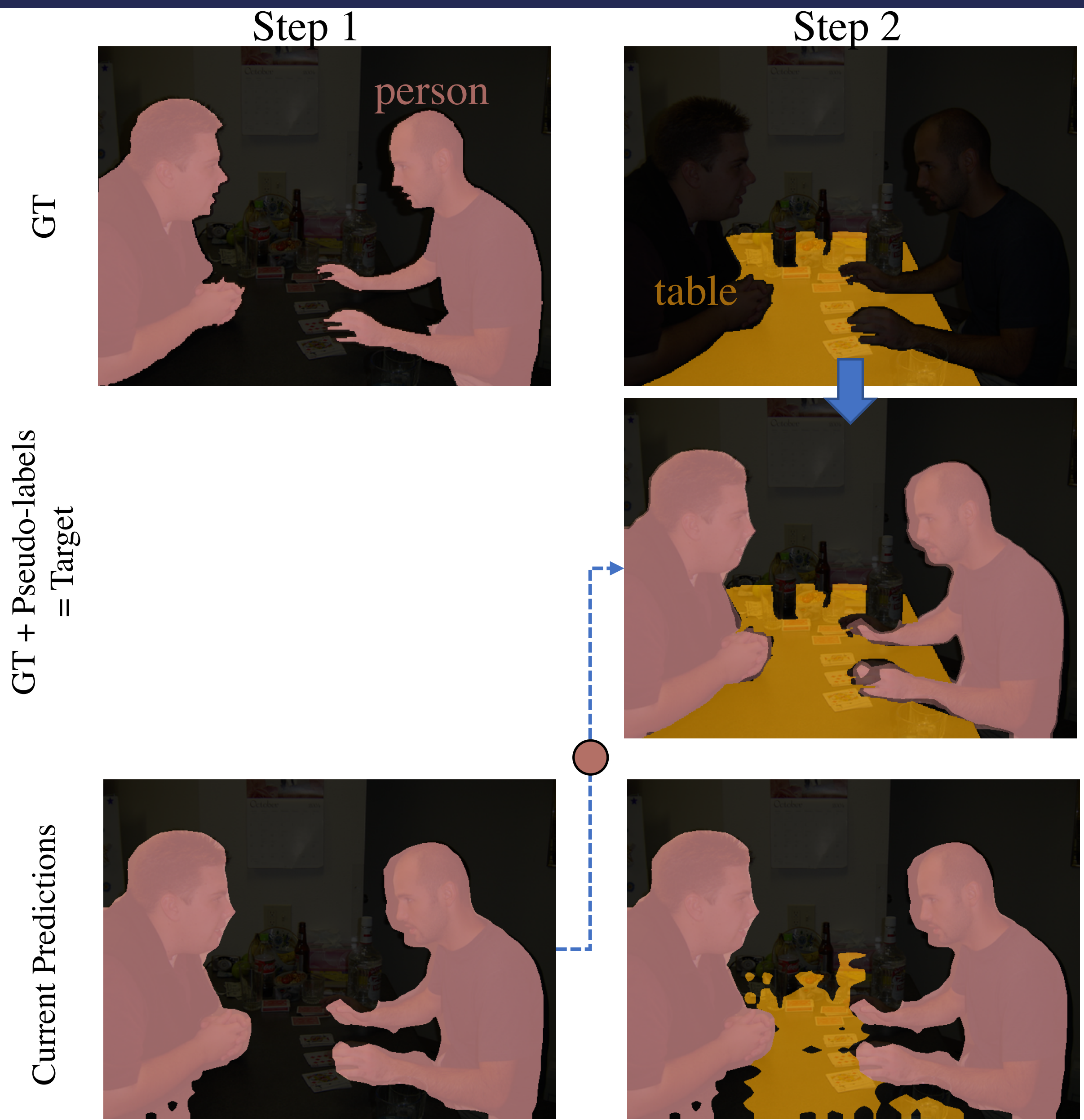
GT

Current Predictions

Problem 1: Background shift

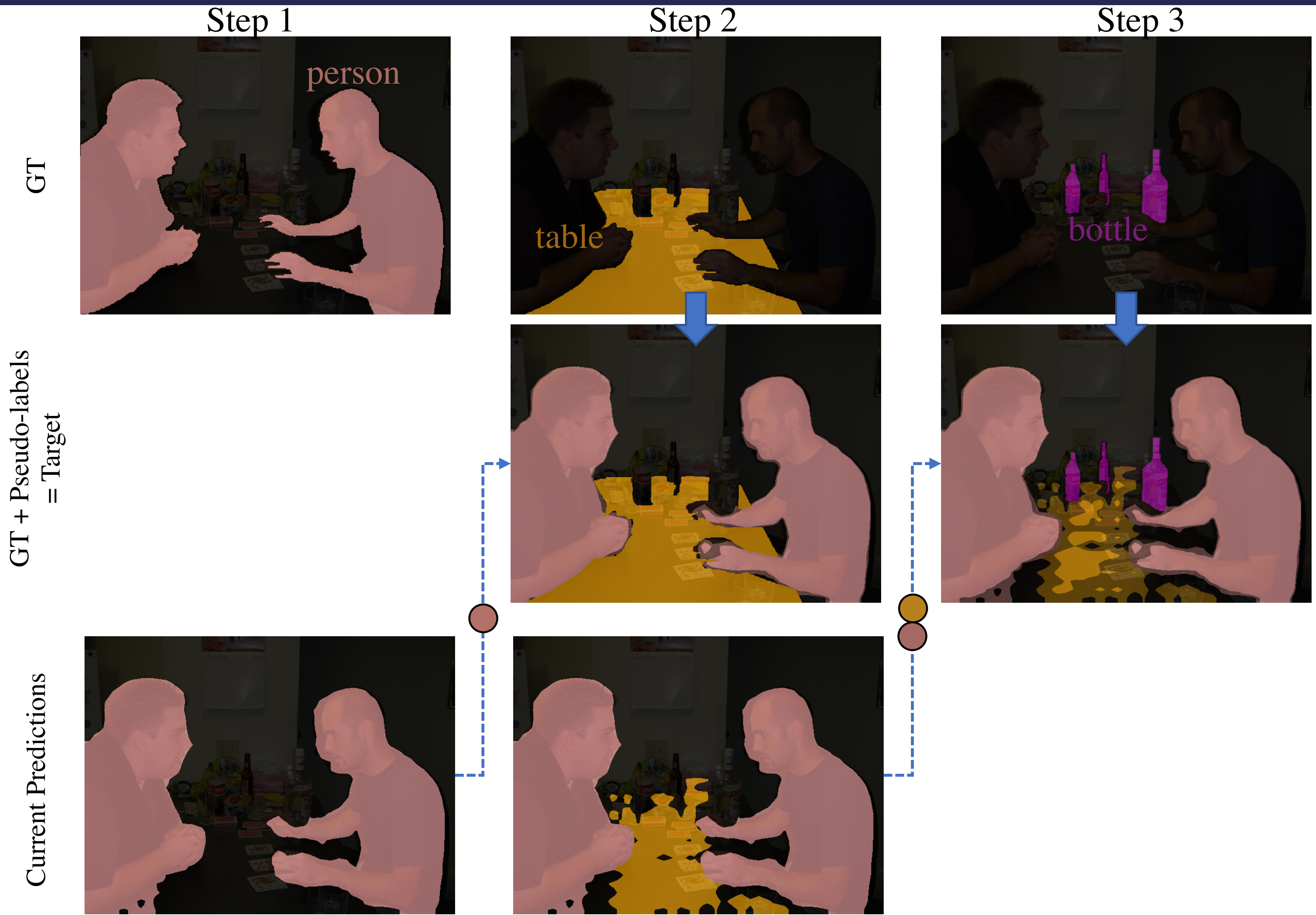


Problem 1: Background shift



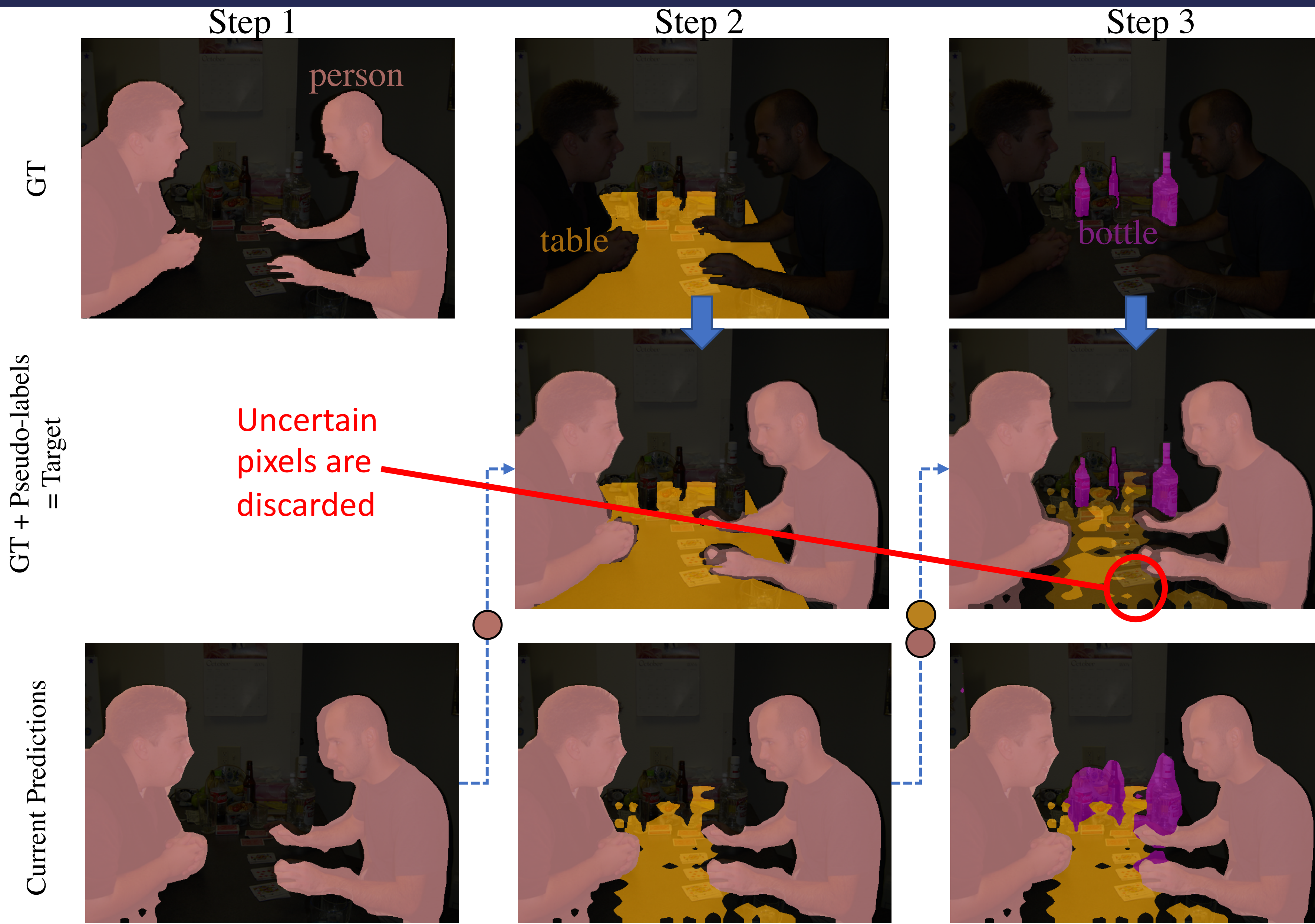
Problem 1: Background shift

Problem 1: Background shift



Problem 1: Background shift

Problem 1: Background shift



Problem 1: Background shift

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

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

Method	**19-1** (2 tasks) 1-19	20	*all*	*avg*	**15-5** (2 tasks) 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)				**15-5** (2 tasks)				**15-1** (6 tasks)			
	1-19	20	*all*	*avg*	1-15	16-20	*all*	*avg*	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

heuritech SCIENCES SORBONNE UNIVERSITÉ

Pascal-VOC (20 classes) experiments

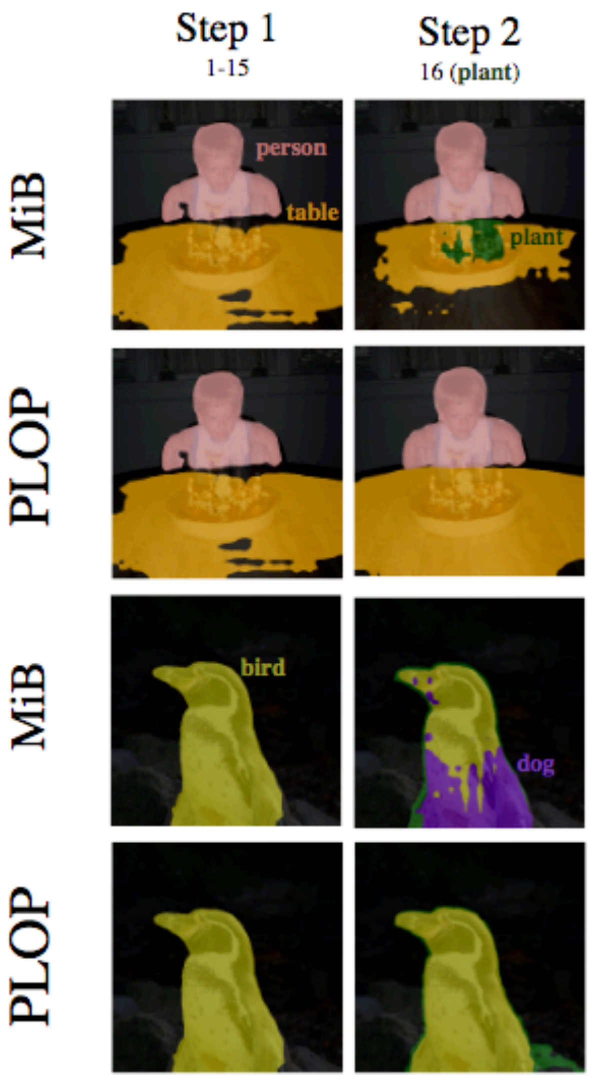
Method	**19-1** (2 tasks)				**15-5** (2 tasks)				**15-1** (6 tasks)			
	1-19	20	*all*	*avg*	1-15	16-20	*all*	*avg*	1-15	16-20	*all*	*avg*
EWC† [36]	26.90	14.00	26.30		24.30	35.50	27.10		0.30	4.30	1.30	
LwF-MC† [54]	64.40	13.30	61.90		58.10	35.00	52.30		6.40	8.40	6.90	
ILT† [49]	67.10	12.30	64.40		66.30	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



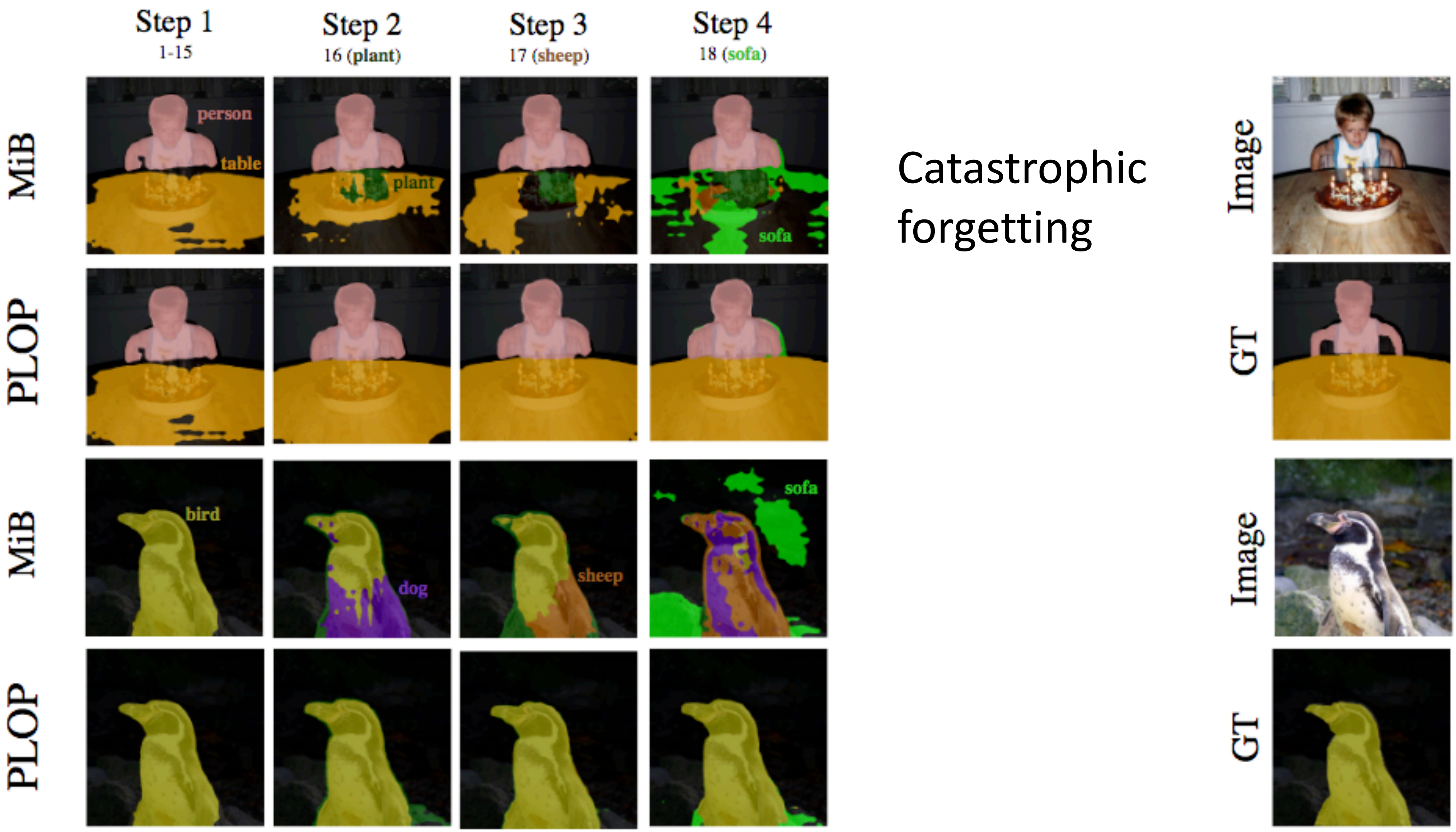
Learn the “plant” class

Visuals

heuritech SCIENCES SORBONNE UNIVERSITÉ

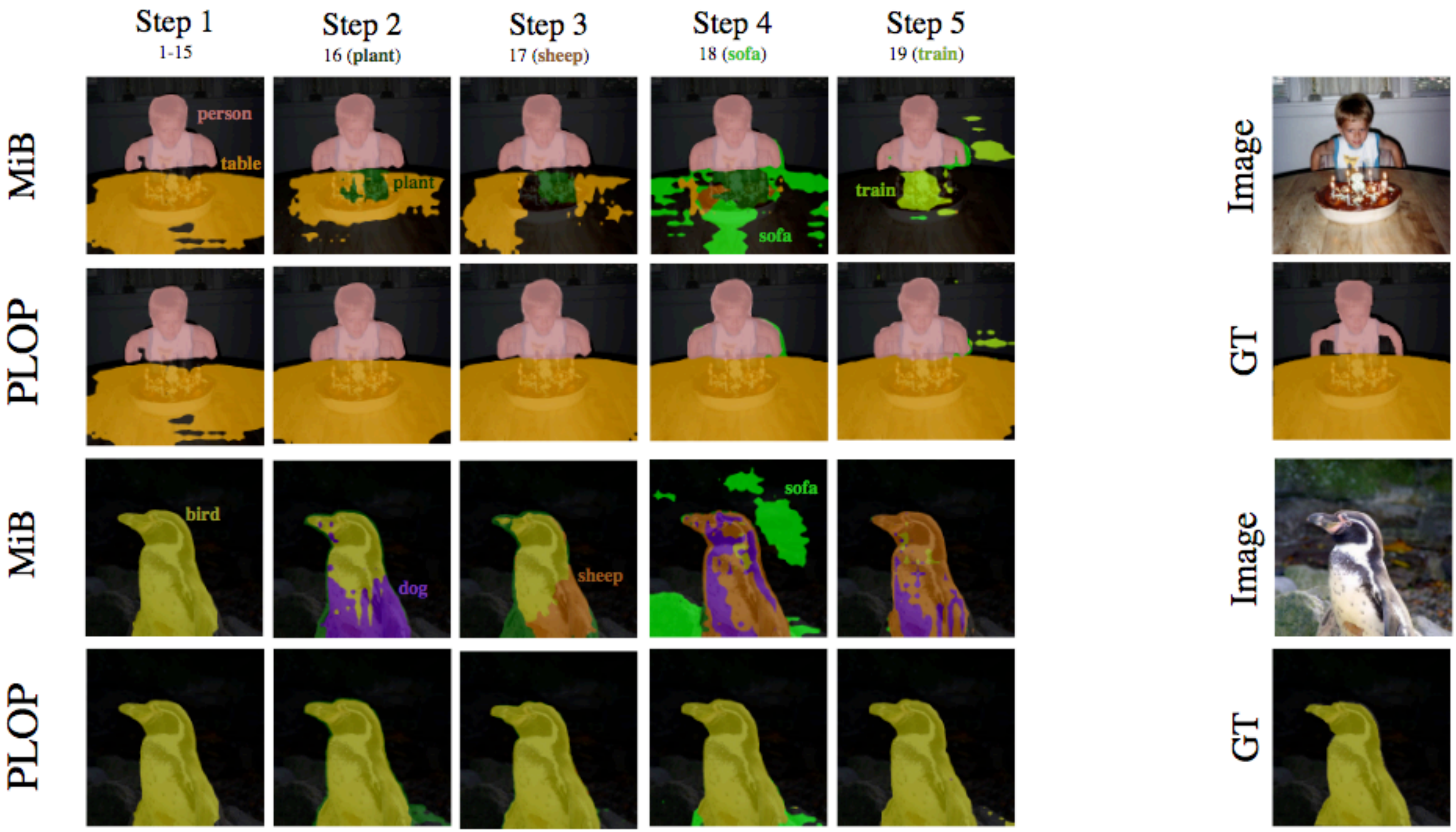
So far, it's still OK

Visuals

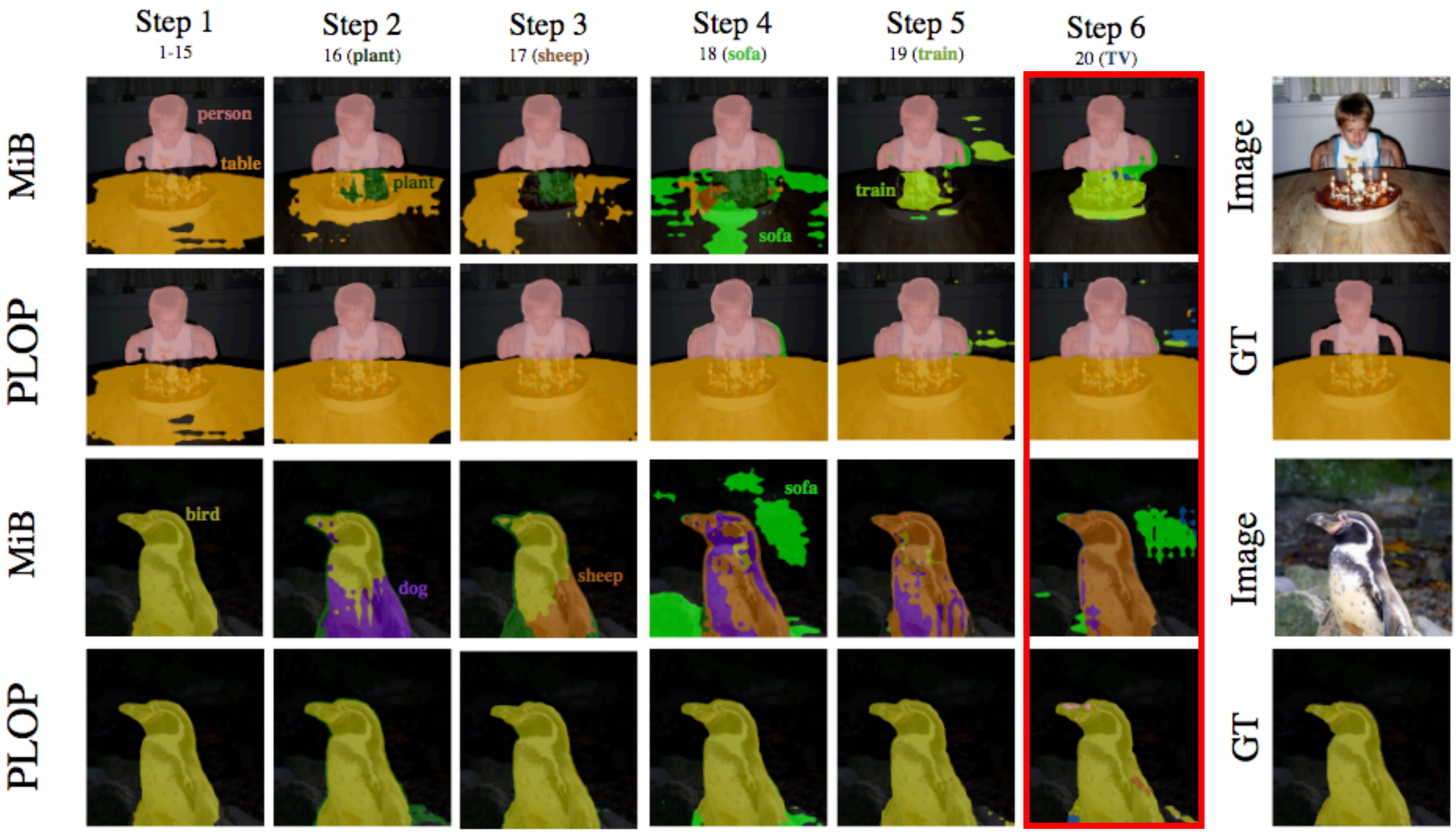


Catastrophic forgetting

Visuals

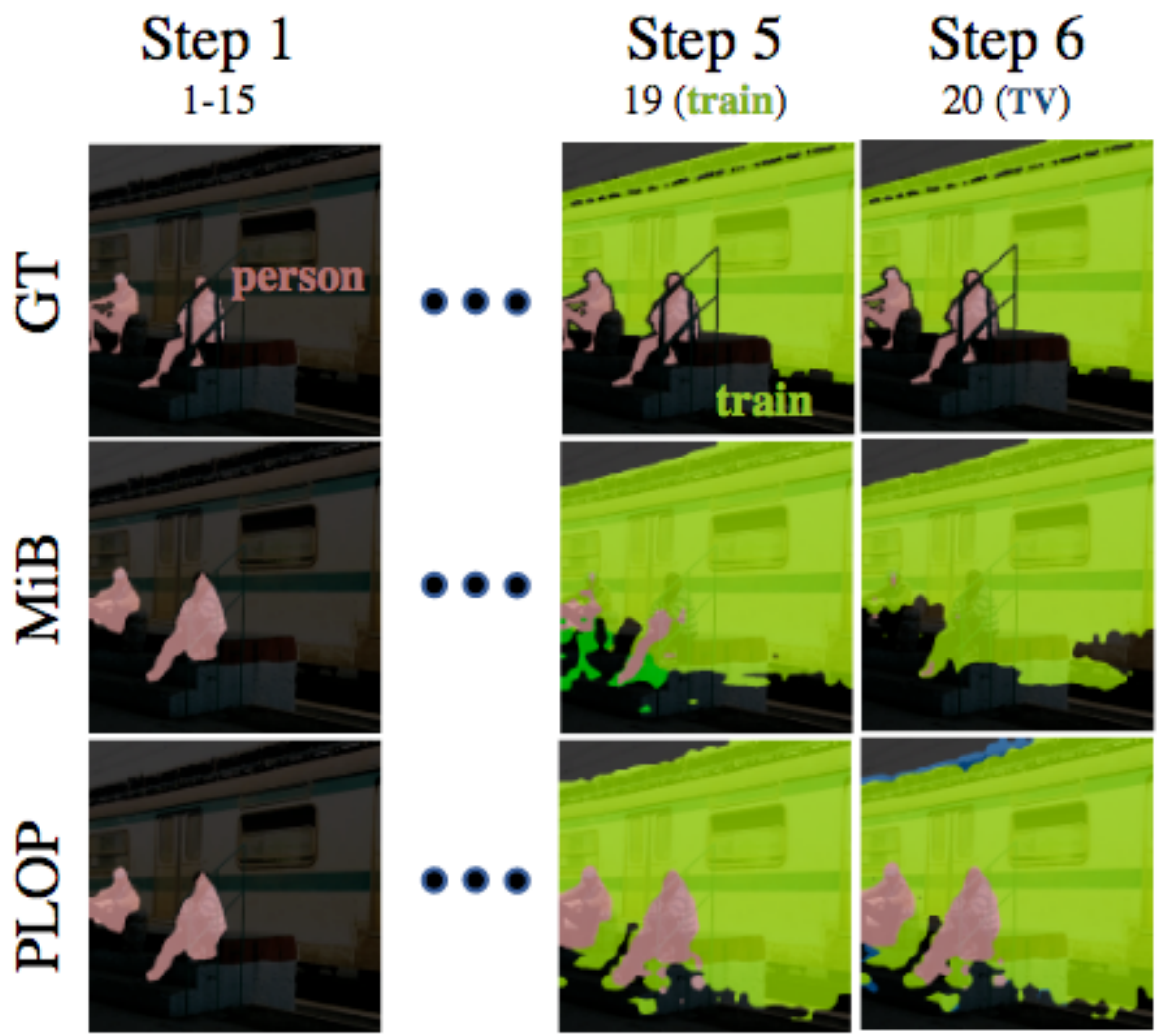

Step 1
1-15
Step 2
16 (plant)
Step 3
17 (sheep)
Step 4
18 (sofa)
Step 5
19 (train)
MiB
PLOP
MiB
PLOP
Image
GT
Image
GT
person
table
plant
sofa
train
bird
dog
sheep
sofa

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



What are your questions?

References

References

[1]: Lomonaco and Maltoni, **CORe50: a New Dataset and Benchmark for Continuous Object Recognition**, 2017
[2]: Robbins, **Catastrophic forgetting, rehearsal and pseudorehearsal**, 1992
[3]: Rebuffi et al., **iCaRL: Incremental Classifier and Representation Learning**, 2017
[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
[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
[11]: Wu et al., **Large scale incremental learning**, 2019
[12]: Hou et al., **Learning an unified classifier incrementally via rebalancing**, 2019
[13]: Cermelli et al., **Modeling the Background for Incremental in Semantic Segmentation**, 2020
[14]: Chaudhry et al., **Riemannian Walk for Incremental Learning: Understanding Forgetting and Intransigence**, 2018
[15]: Shin et al., **Continual Learning with Deep Generative Replay**, 2017
[16] Veniat et al., **Efficient Continual Learning with Modular Networks and Task-Drive Priors**, 2021