heuritech

SCIENCES SORBONNE UNIVERSITÉ

Continual Learning

Learning continuously without forgetting

Arthur Douillard

https://arthurdouillard.com
@Ar_Douillard

LIP6

MLIA Machine Learning & Deep Learning for Information Access

Who am I?

Brief Bio

PhD student at **Sorbonne** with Prof. Matthieu Cord since July 2019

Research Scientist at **Heuritech**

Teacher at **EPITA**

… and an ex-intern at Dataiku

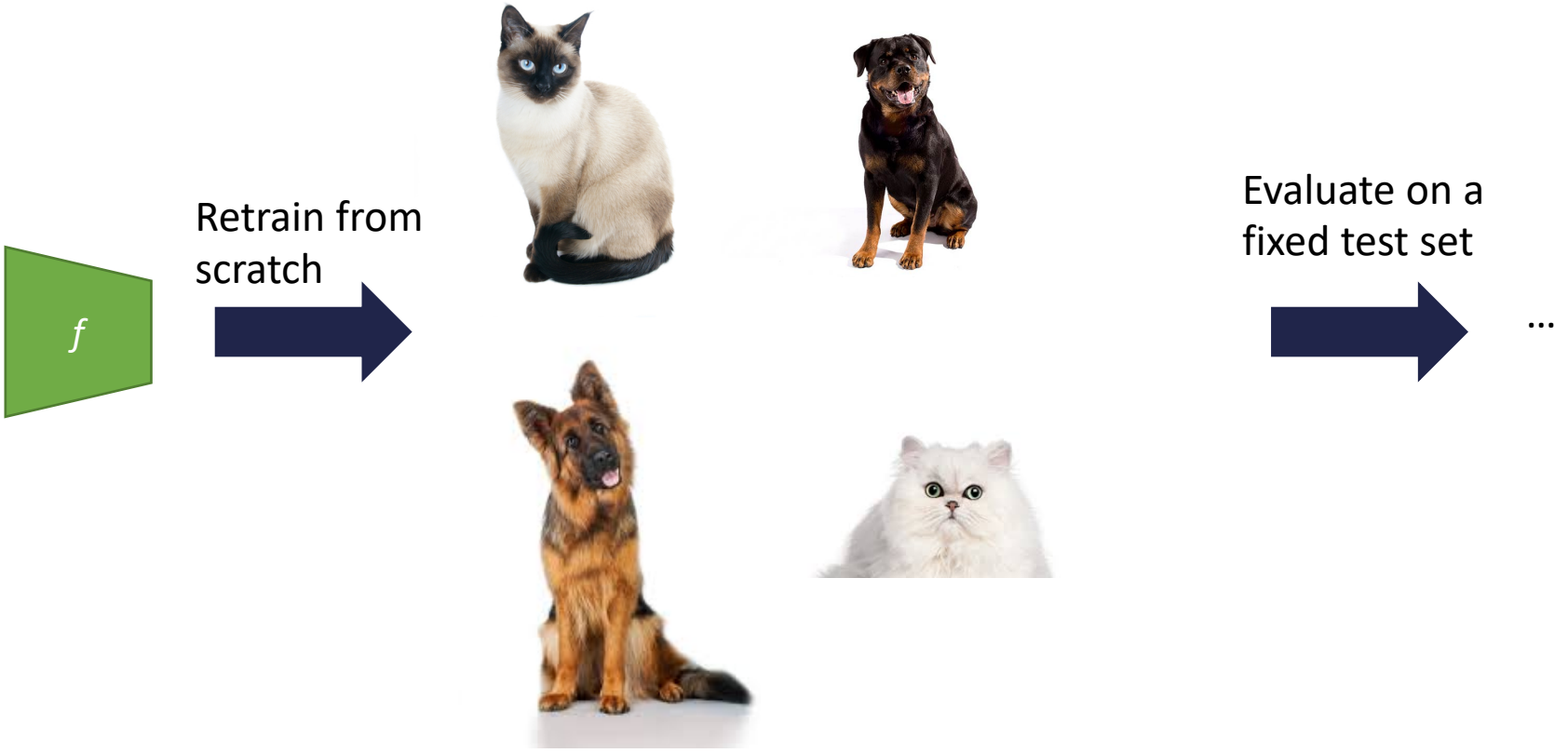
What is Continual Learning?

What

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

What

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



What

heuritech

SCIENCES SORBONNE UNIVERSITÉ

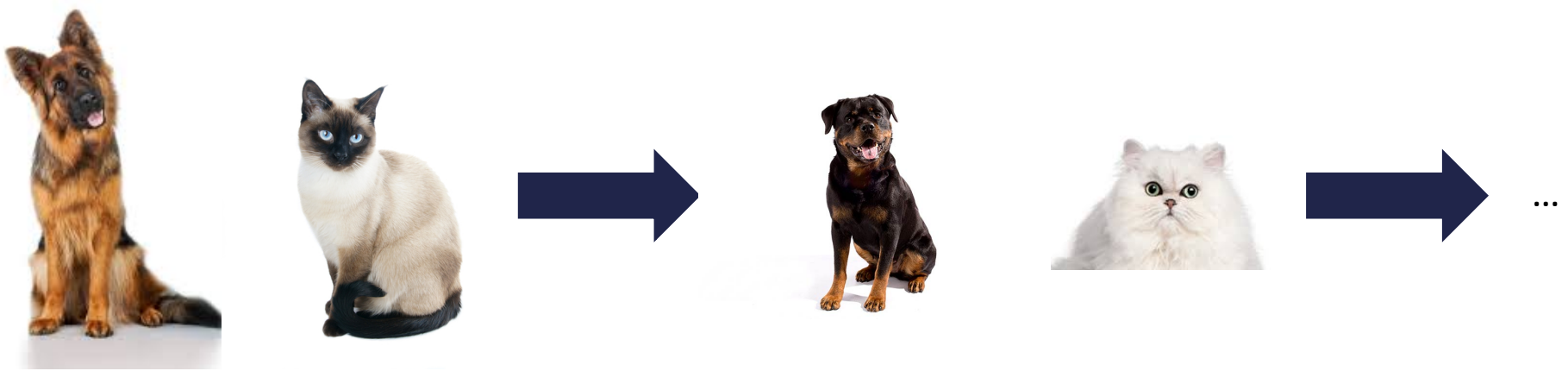
Retraining everytime is not always possible:

- **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 never **independent and identically distributed** (i.i.d.)

New samples [1] may appear:



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

What

Real world data is never **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

What

Real world data is never **independent and identically distributed** (i.i.d.)

New samples and classes [1] may appear:

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

Protocol

heuritech SCIENCES SORBONNE UNIVERSITÉ

Protocol

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

Protocol

heuritech SCIENCES SORBONNE UNIVERSITÉ

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

heuritech

SCIENCES SORBONNE UNIVERSITÉ

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

heuritech

SCIENCES SORBONNE UNIVERSITÉ

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

heuritech

SCIENCES SORBONNE UNIVERSITÉ

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 $\{1, \dots, t\}$

Evaluation

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

[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]?

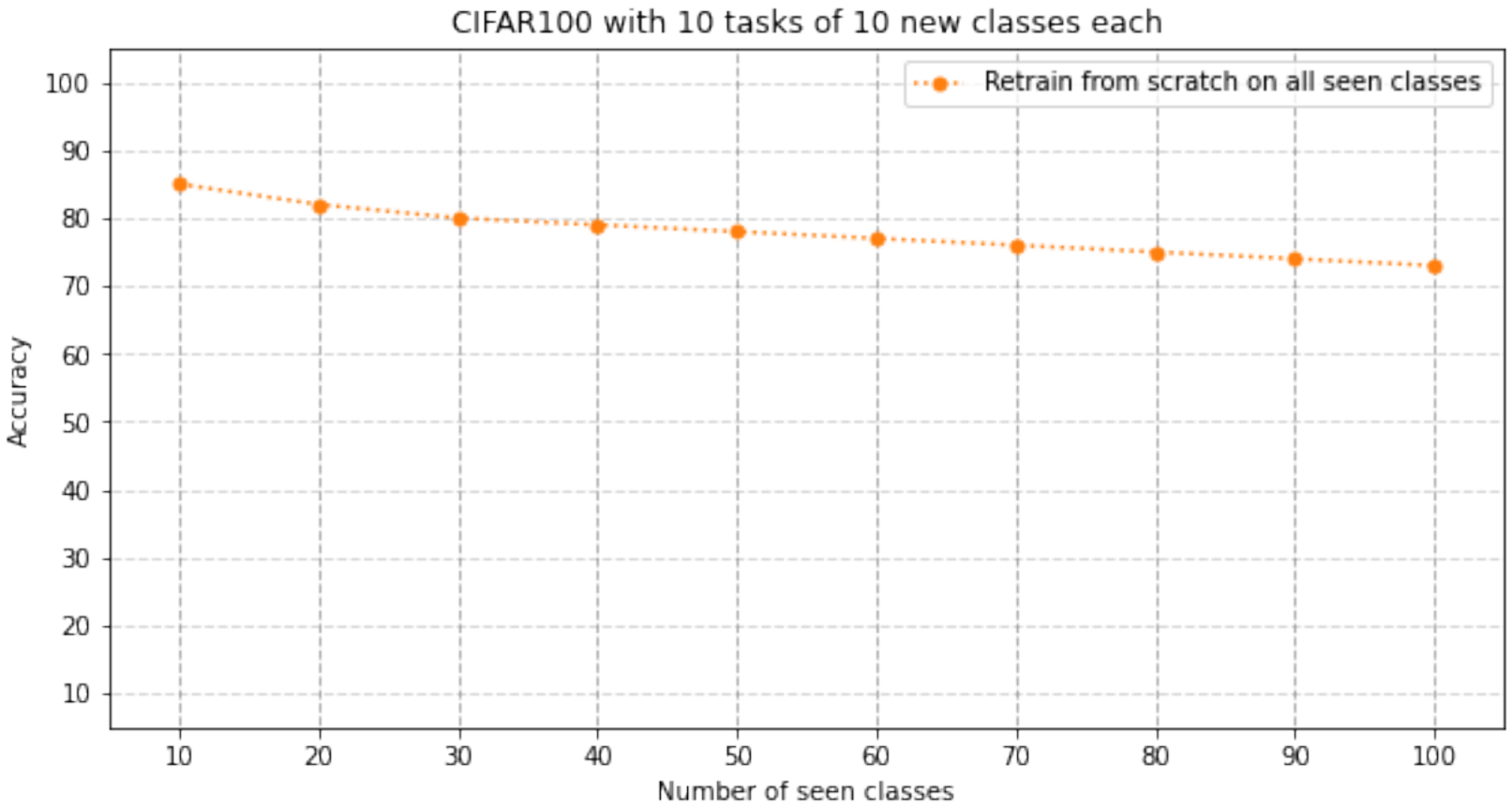
Final Evaluation:

Single → {dog, cat, boat, plane} ?

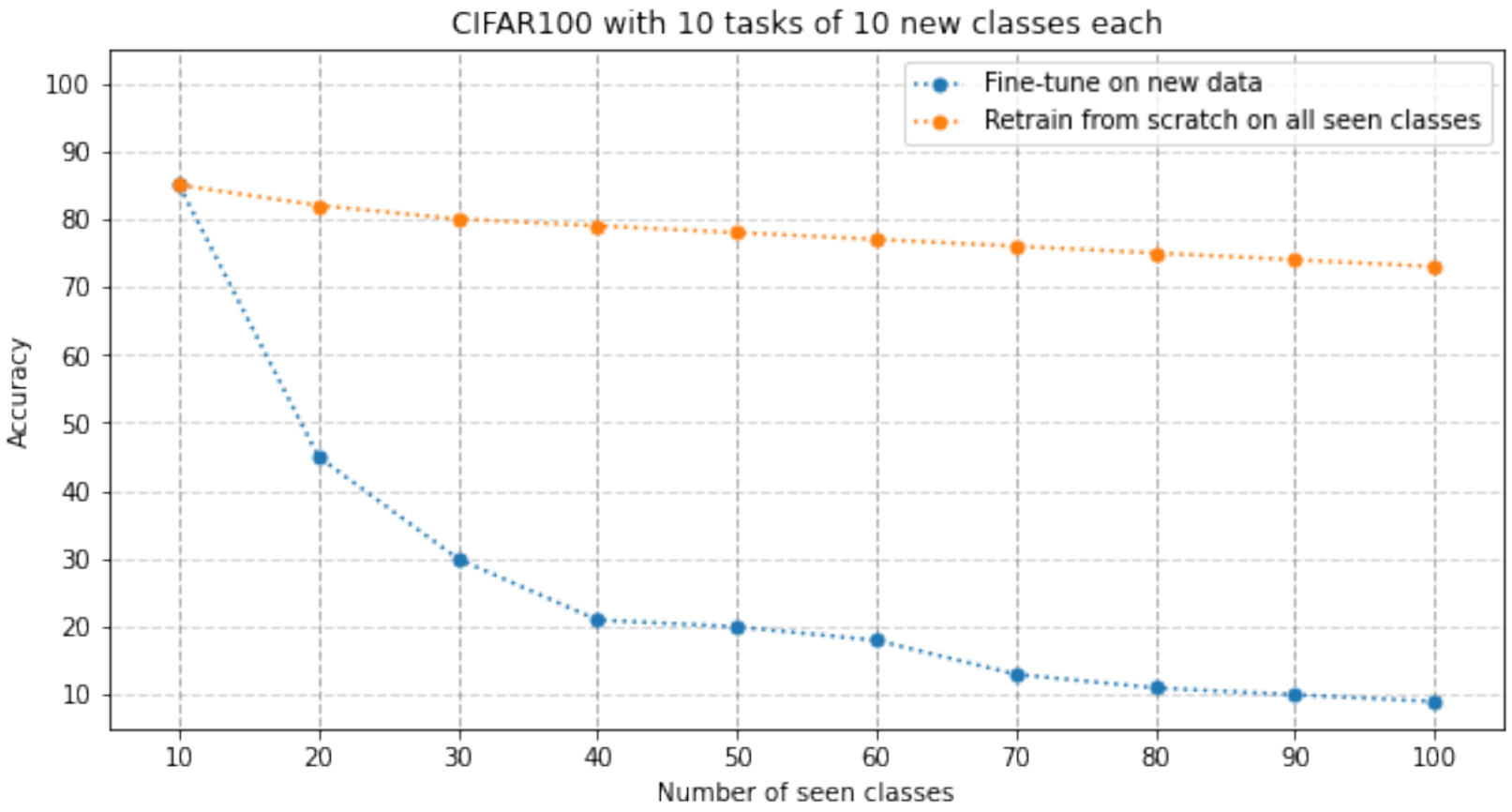
Multi → {dog, cat} ?

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

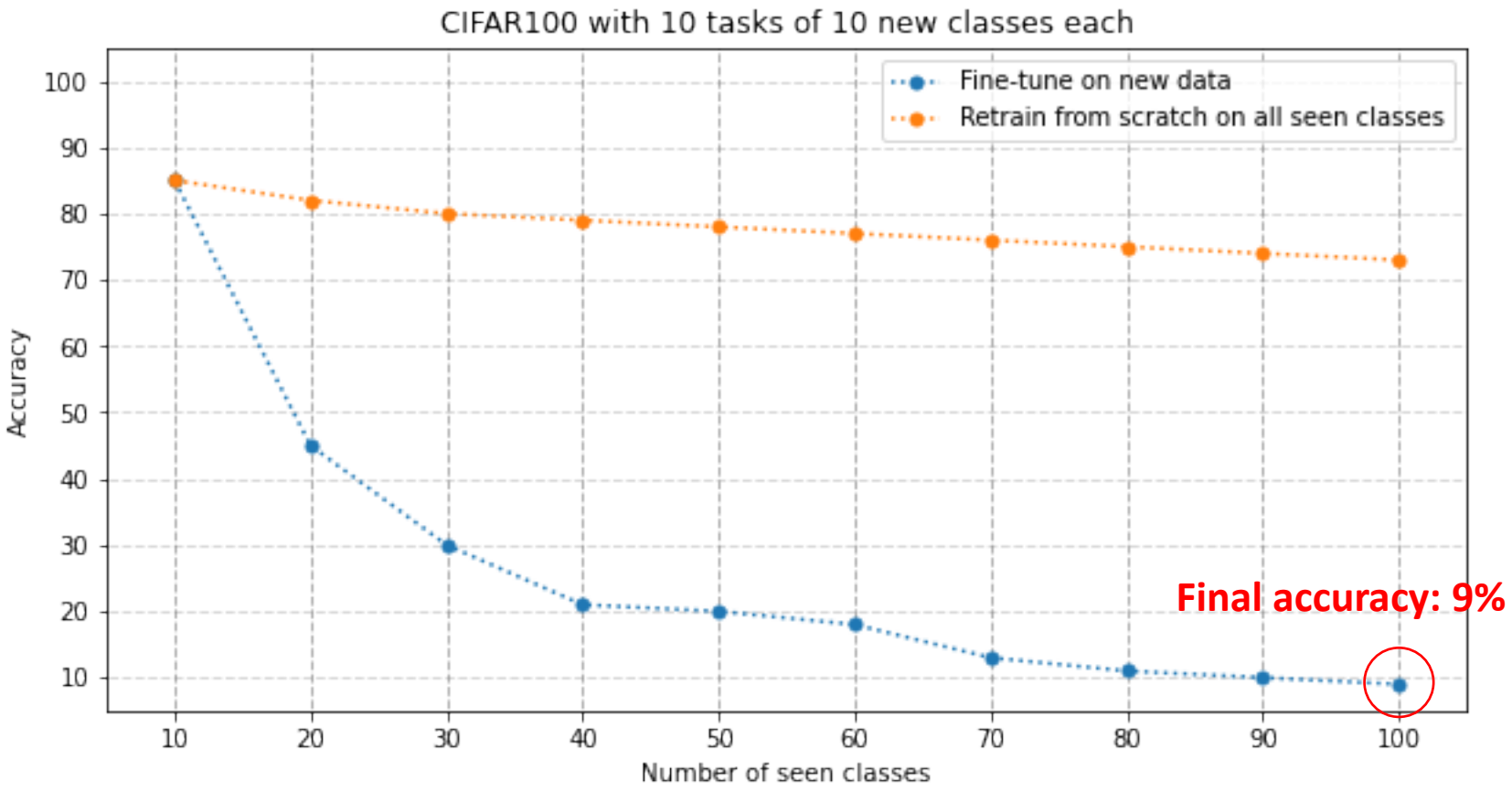
Example


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

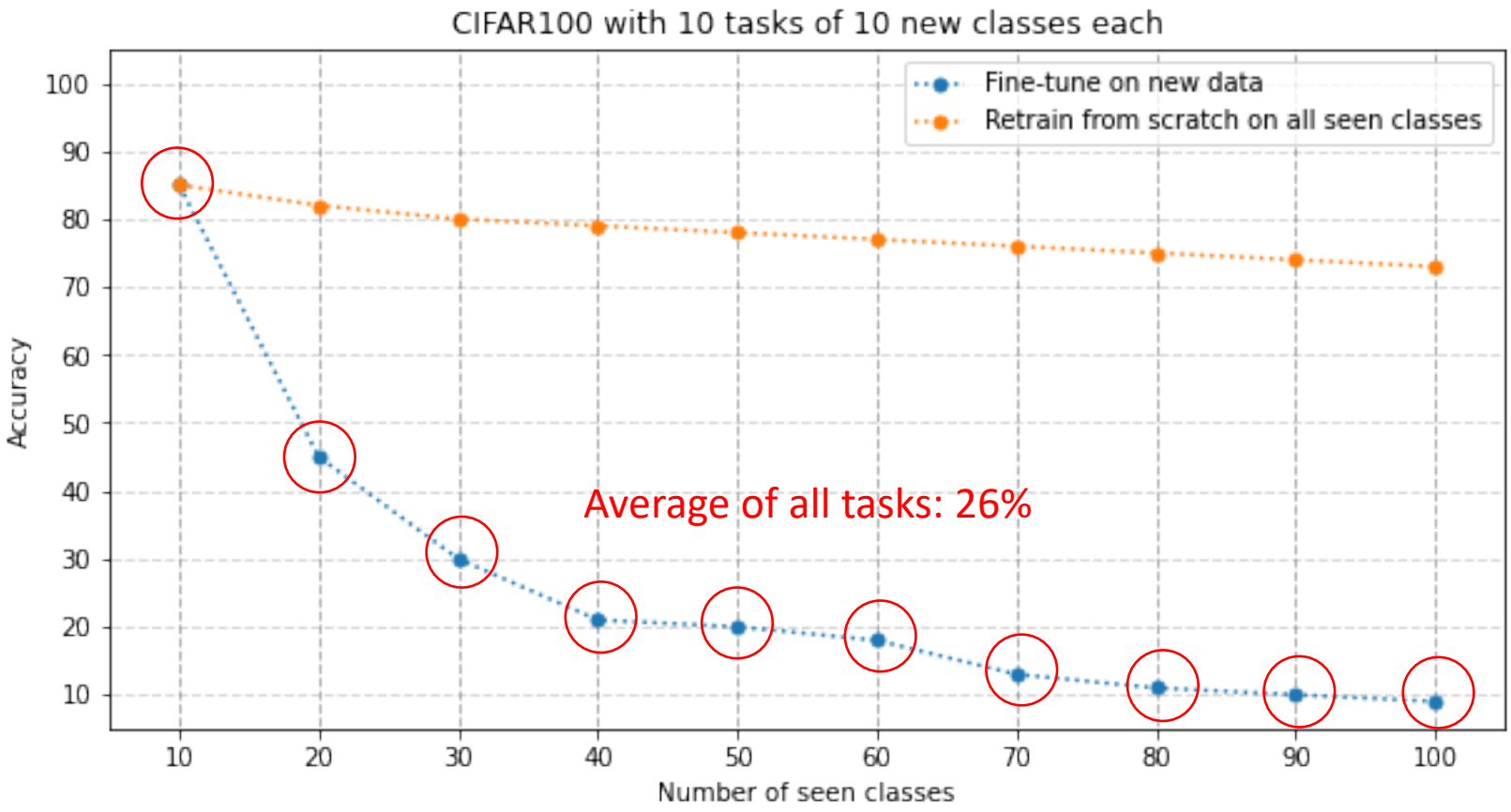
Example


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

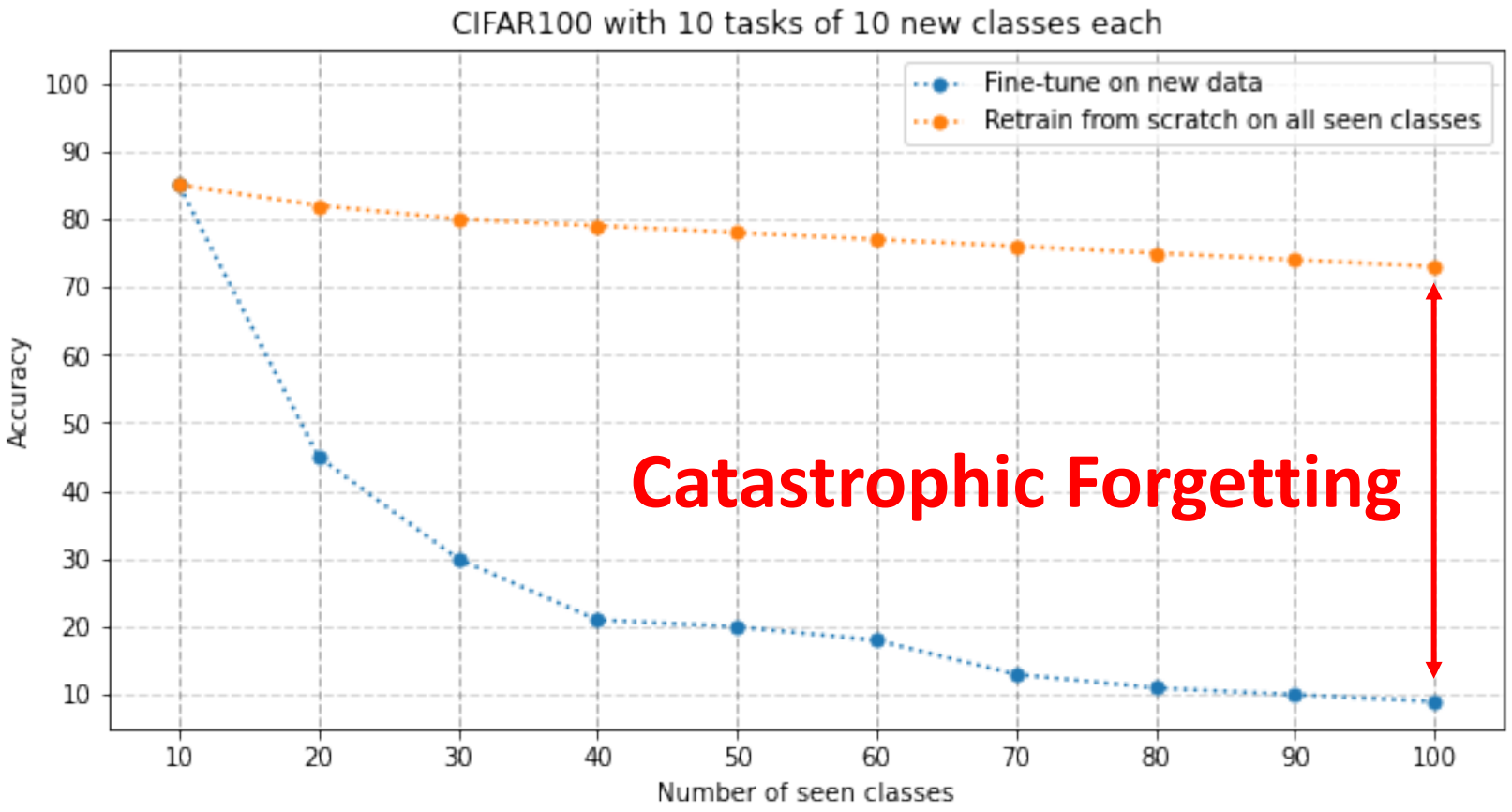
Example


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

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
Number of seen classes
100
90
80
70
60
50
40
30
20
10
10
20
30
40
50
60
70
80
90
100

How to Solve it?

Broad Strategies

heuritech

SCIENCES SORBONNE UNIVERSITÉ

1. Rehearsal
2. Constraints
3. Sub-networks
4. Classifier Correction

Broad Strategies

heuritech

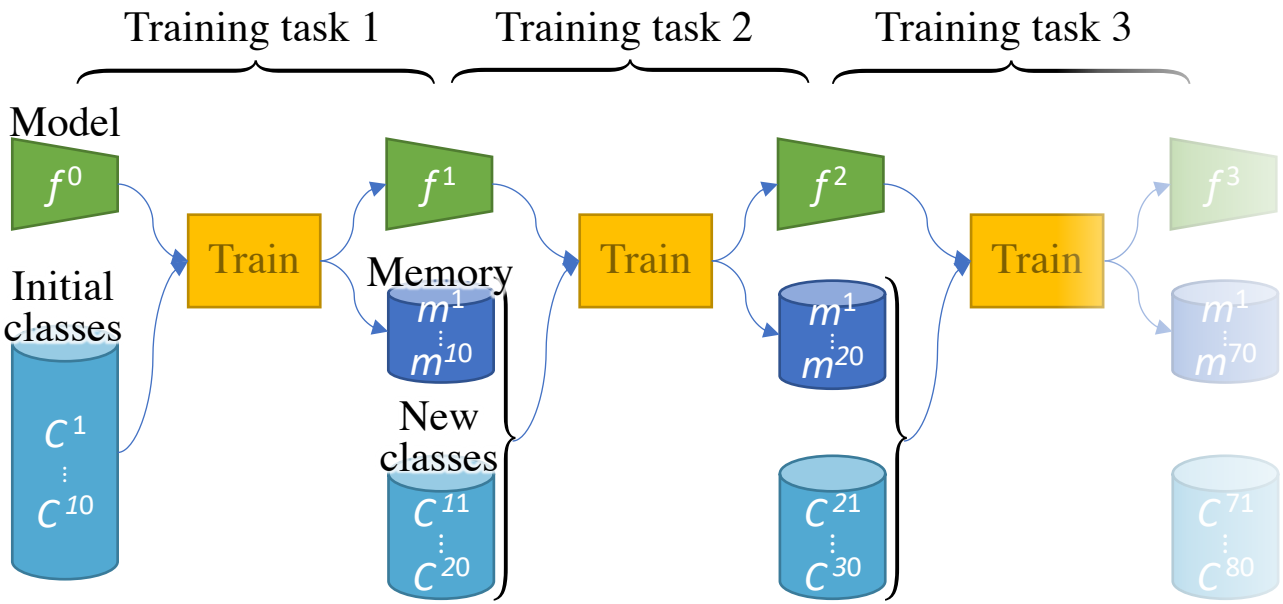
SCIENCES SORBONNE UNIVERSITÉ

1. **Rehearsal**
2. Constraints
3. Sub-networks
4. Classifier Correction

1. Rehearsal

Replay a limited amount of previous data

e.g. iCaRL [3]

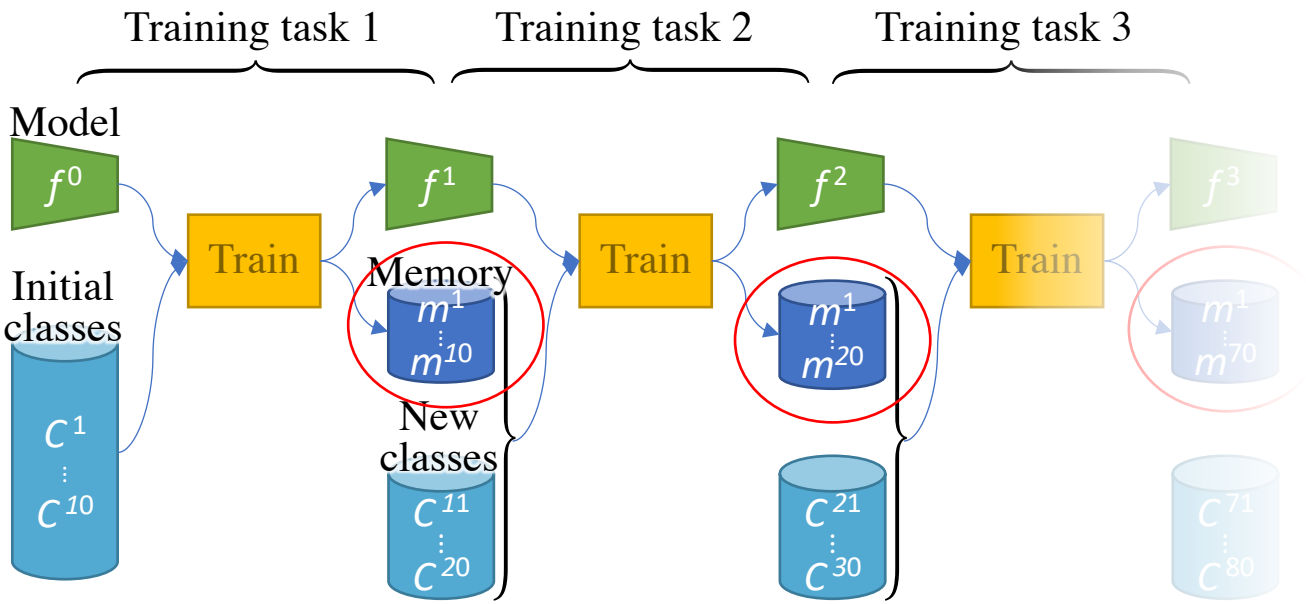


[3]: Rebuffi et al., iCaRL: Incremental Classifier and Representation Learning, 2017

1. Rehearsal

Replay a limited amount of previous data

e.g. iCaRL [3]

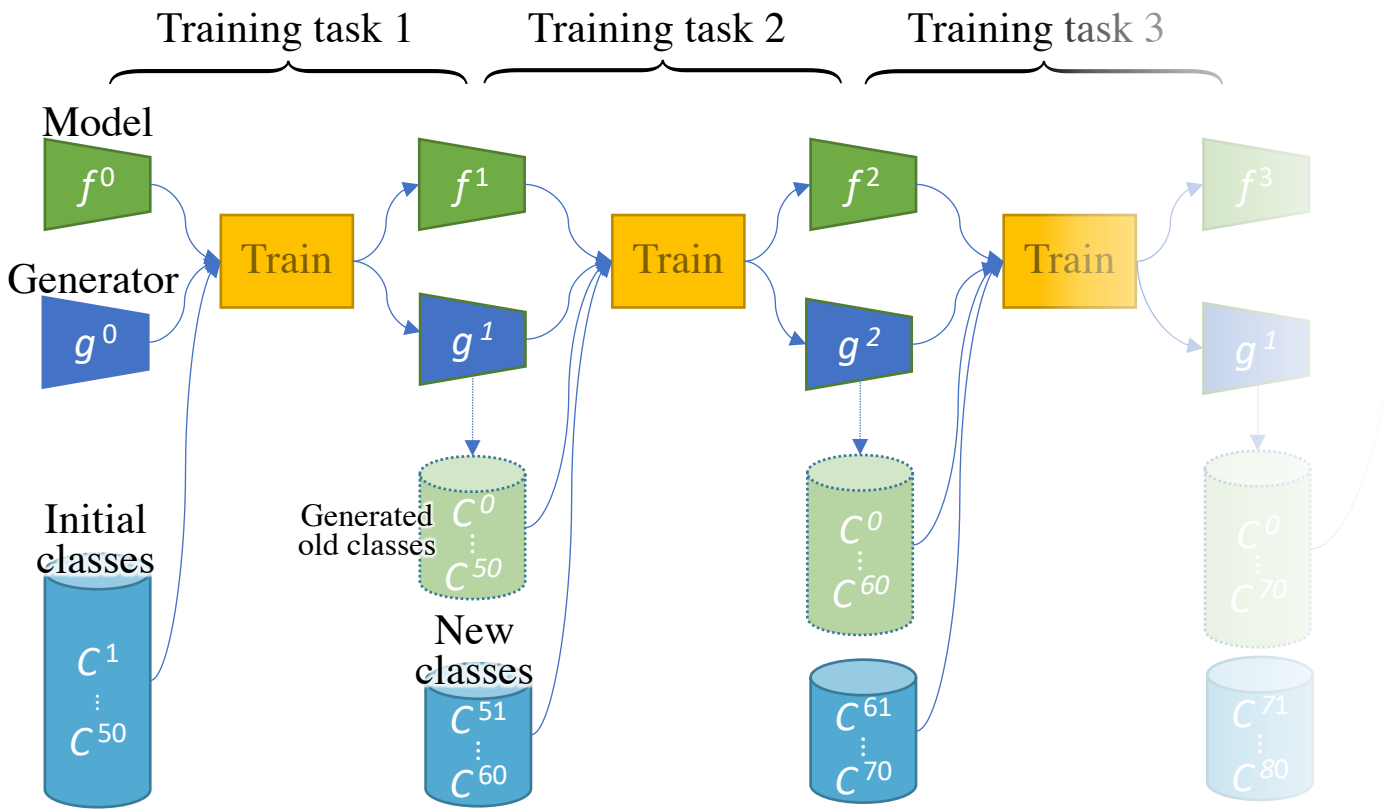


[3]: Rebuffi et al., iCaRL: Incremental Classifier and Representation Learning, 2017

1. Rehearsal

Replay a limited amount of previous data

e.g. DGR [15]

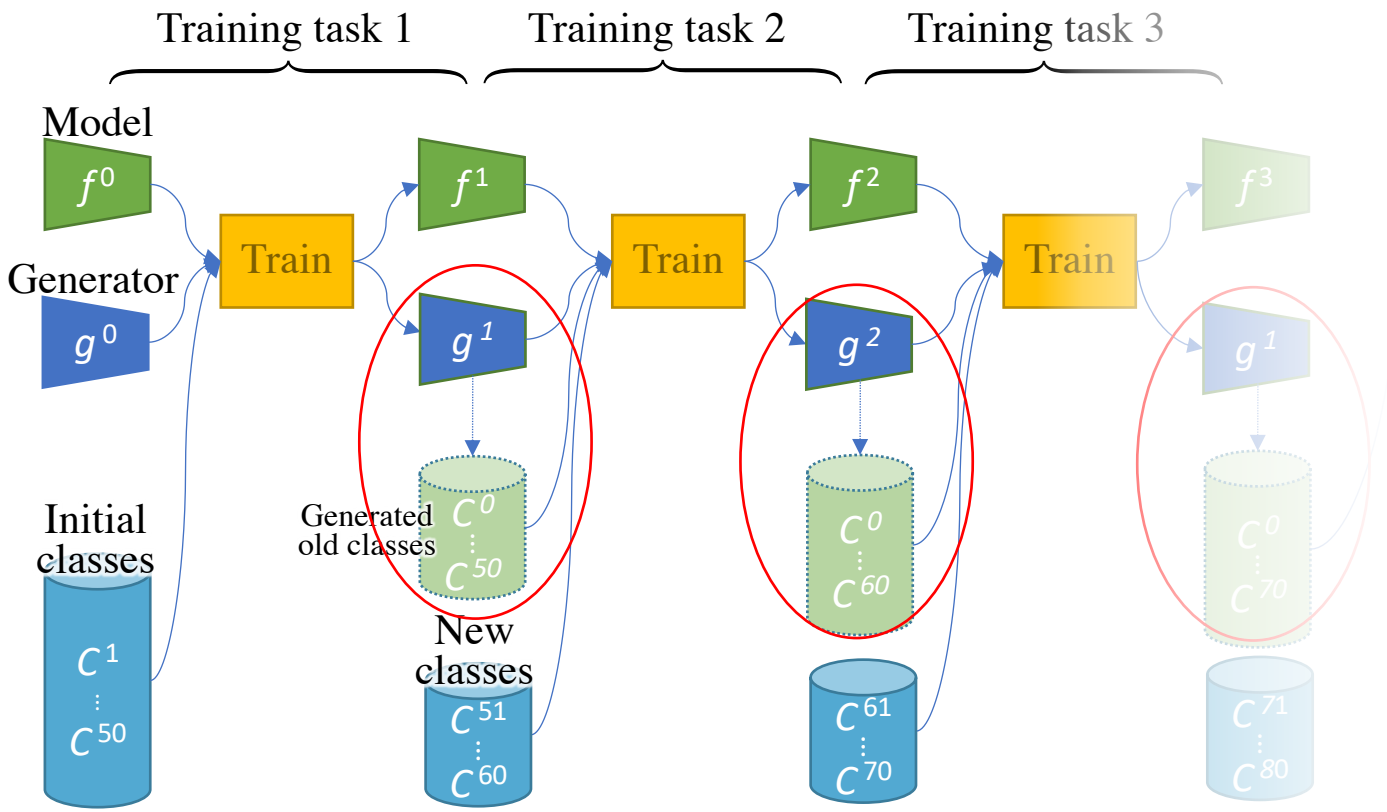


[15]: Shin et al., Continual Learning with Deep Generative Replay, 2017

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

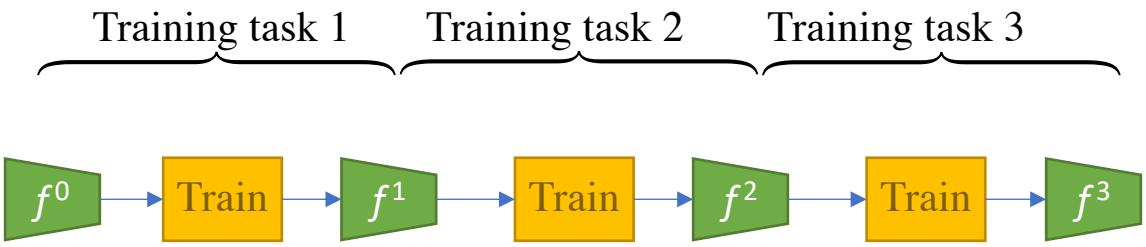
heuritech

SCIENCES SORBONNE UNIVERSITÉ

1. Rehearsal
2. **Constraints**
3. Sub-networks
4. Classifier Correction

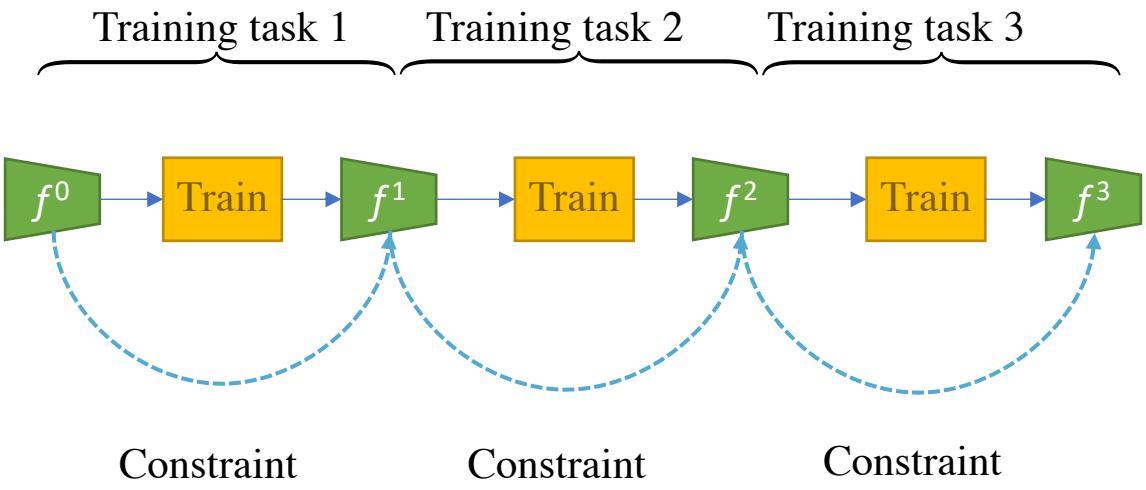
2. Constraints

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



2. Constraints

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

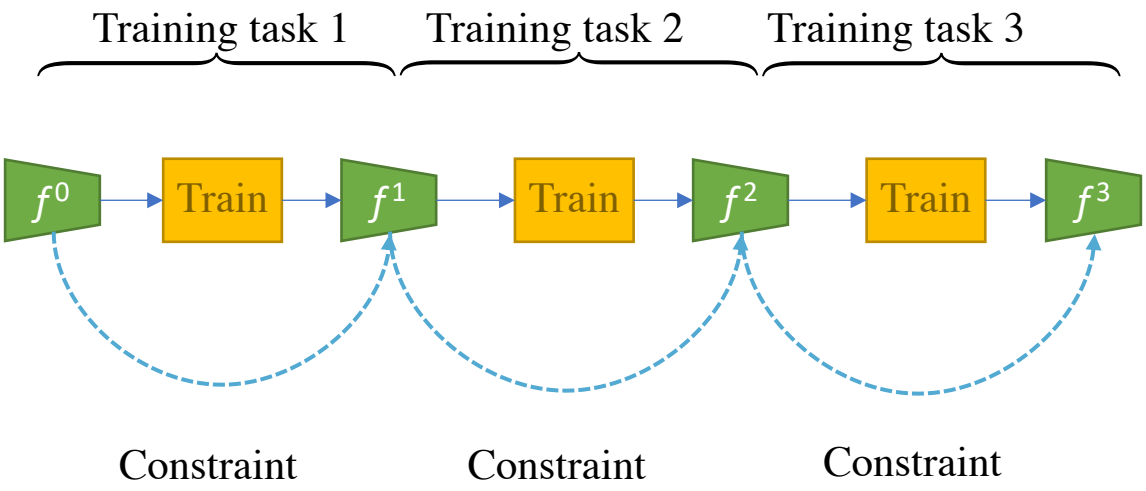


2. Constraints

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

On the weights (EWC [4])

$$\mathcal{L}(\theta) = \mathcal{L}_B(\theta) + \sum_i \frac{\lambda}{2} F_i(\theta_i - \theta^*_{A,i})^2$$



[4]: Kirkpatrick et al., Overcoming catastrophic forgetting in neural networks, 2017

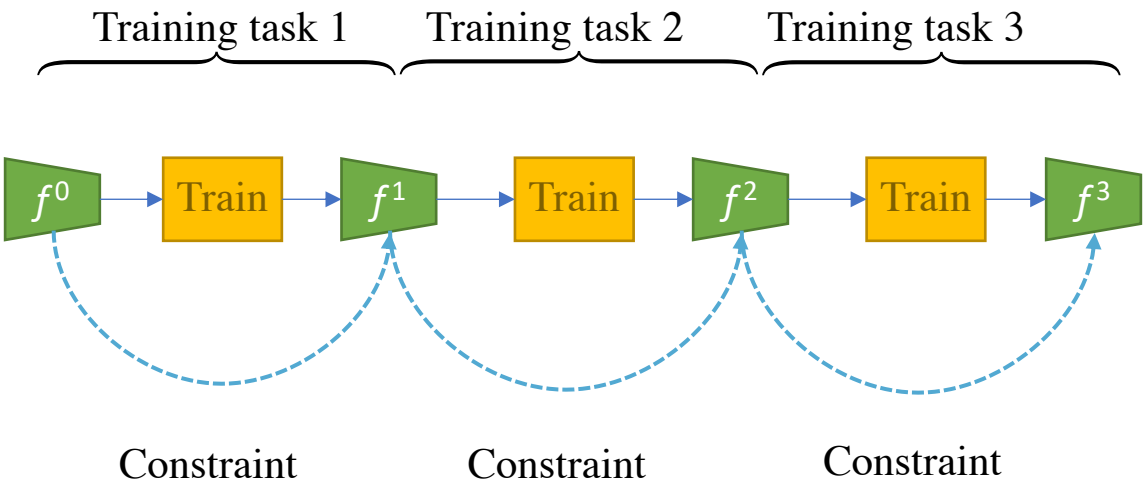
2. Constraints

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

On the probabilities (LwF [5])

$$\mathcal{L}_{old}(\mathbf{y}_o, \hat{\mathbf{y}}_o) = -H(\mathbf{y}'_o, \hat{\mathbf{y}}'_o) = -\sum_{i=1}^{l} y_o'^{(i)} \log \hat{y}_o'^{(i)}$$

$$y_o'^{(i)} = \frac{(y_o^{(i)})^{1/T}}{\sum_j (y_o^{(j)})^{1/T}}, \quad \hat{y}_o'^{(i)} = \frac{(\hat{y}_o^{(i)})^{1/T}}{\sum_j (\hat{y}_o^{(j)})^{1/T}}.$$



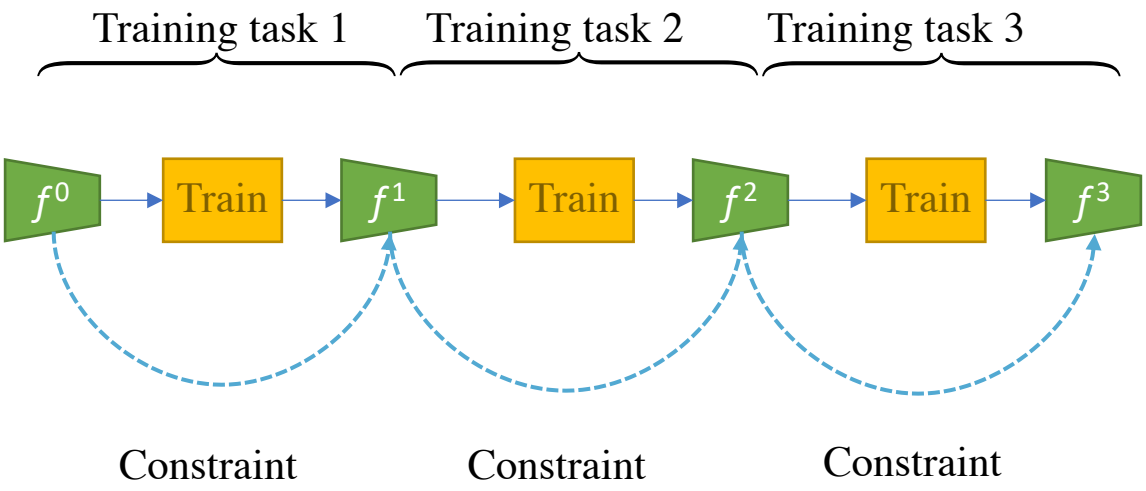
[4]: Kirkpatrick et al., Overcoming catastrophic forgetting in neural networks, 2017
[5]: Li and Hoiem, Learning without forgetting, 2016

2. Constraints

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

On the gradients (GEM [6])

$$\langle g, g_k \rangle := \left\langle \frac{\partial \ell(f_\theta(x,t),y)}{\partial \theta}, \frac{\partial \ell(f_\theta, \mathcal{M}_k)}{\partial \theta} \right\rangle \geq 0, \text{ for all } k < t.$$



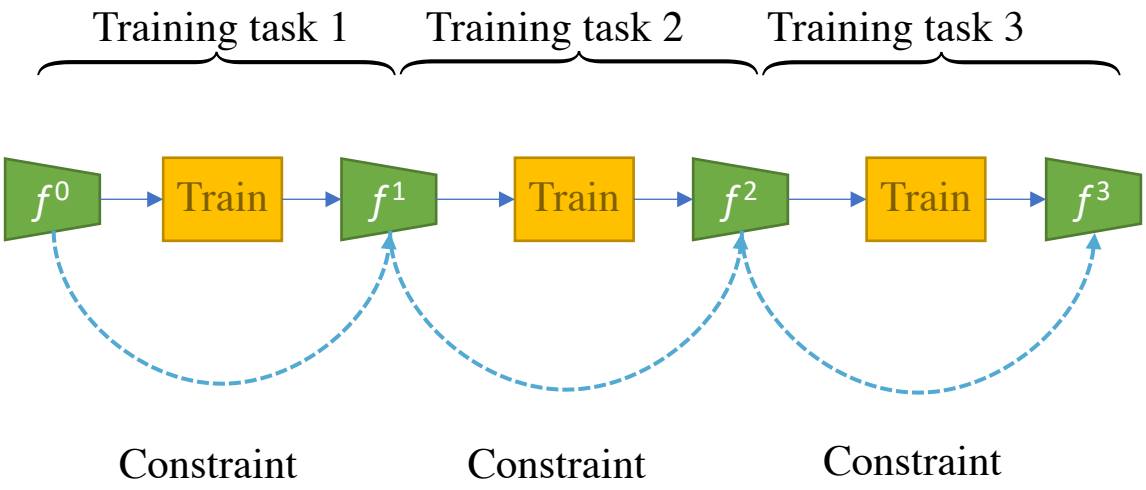
[4]: Kirkpatrick et al., Overcoming catastrophic forgetting in neural networks, 2017
[5]: Li and Hoiem, Learning without forgetting, 2016
[6]: Lopez-Paz and Ranzato, Gradient episodic memory for continual learning, 2017

2. Constraints

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

On the features (PODNet [7])

$$\mathcal{L}_{\text{POD-width}}(\mathbf{h}_\ell^{t-1}, \mathbf{h}_\ell^t) = \sum_{c=1}^{C} \sum_{h=1}^{H} \left\| \sum_{w=1}^{W} \mathbf{h}_{\ell,c,w,h}^{t-1} - \sum_{w=1}^{W} \mathbf{h}_{\ell,c,w,h}^{t} \right\|^2$$



[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

heuritech

SCIENCES SORBONNE UNIVERSITÉ

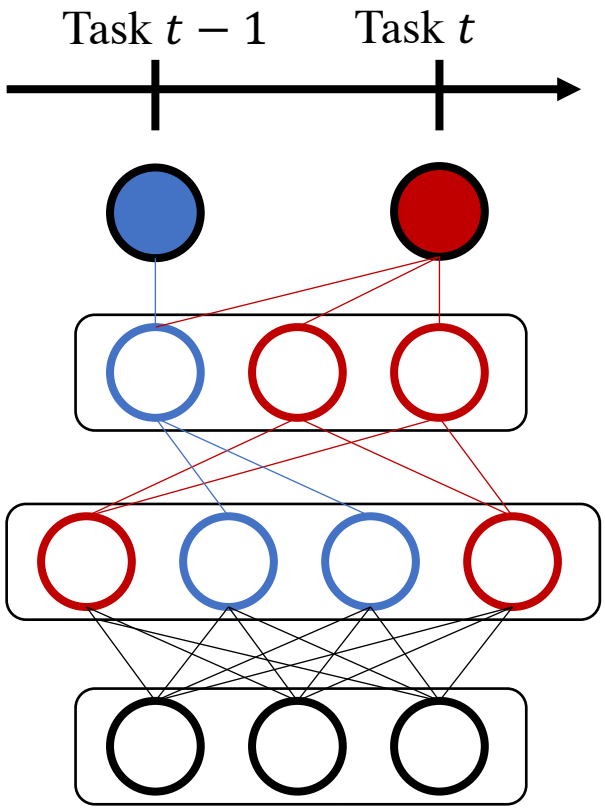
1. Rehearsal
2. Constraints
3. **Sub-networks**
4. Classifier Correction

3. Sub-networks

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]).



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

Broad Strategies

heuritech

SCIENCES SORBONNE UNIVERSITÉ

1. Rehearsal
2. Constraints
3. Sub-networks
4. **Classifier Correction**

4. Classifier Correction

Classifier is **biased** towards new classes

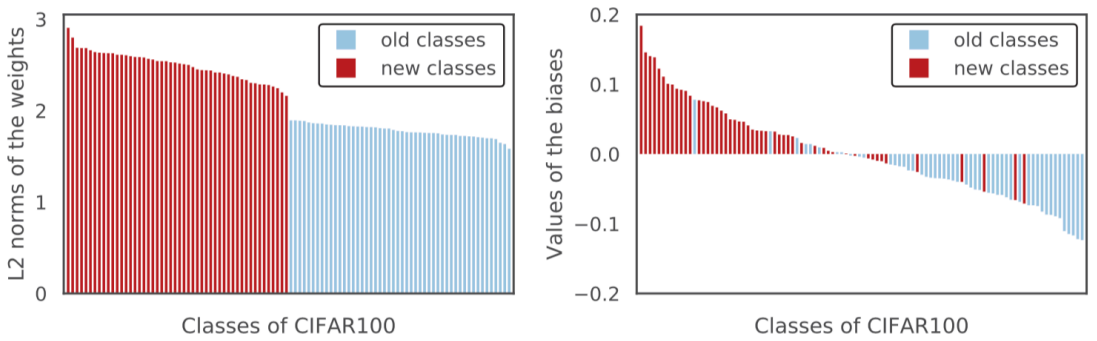
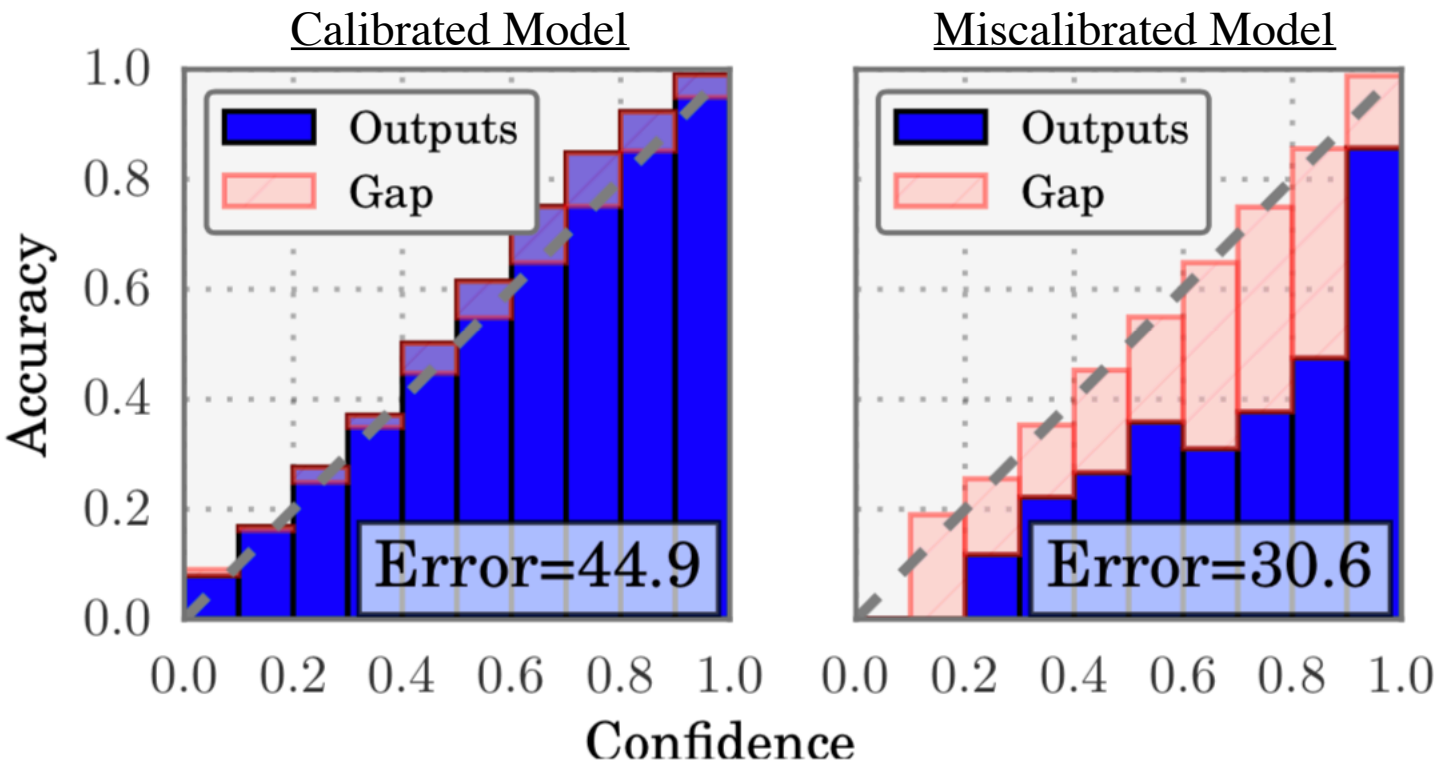


Figure 3. Visualization of the weights and biases in the last layer for old and new classes. The results come from the incremental setting of CIFAR100 (1 phase) by iCaRL [29].

4. Classifier Correction

Classifier is **biased** towards new classes

Can be recalibrated (BiC [11])

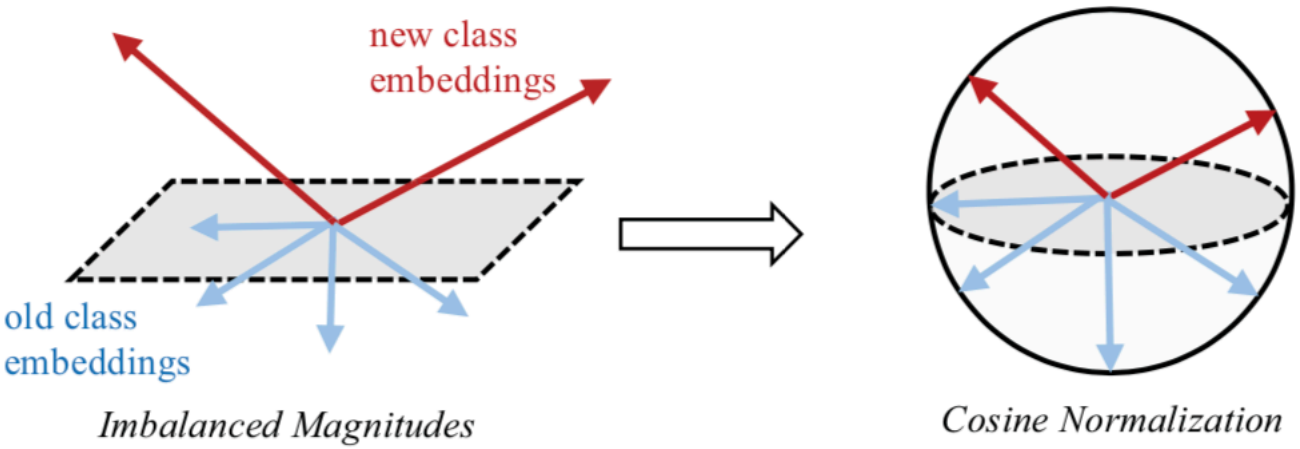


[11]: Wu et al., Large scale incremental learning, 2019

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

Two of our publications

1. PODNet, ECCV 2020

heuritech SCIENCES SORBONNE UNIVERSITÉ

PODNet: Pooled Outputs Distillation for Small-Tasks Incremental Learning

Arthur Douillard[1,2], Matthieu Cord[2,3], Charles Ollion[1], Thomas Robert[1], and Eduardo Valle[4]

Rehearsal + Constraints

1. PODNet, ECCV 2020

heuritech

SCIENCES SORBONNE UNIVERSITÉ

Problems of previous constraints:

- Probabilities → weak

1. PODNet, ECCV 2020

Problems of previous constraints:

- Probabilities → weak
- Weights → Slow and heavy

1. PODNet, ECCV 2020

heuritech

SCIENCES SORBONNE UNIVERSITÉ

Problems of previous constraints:

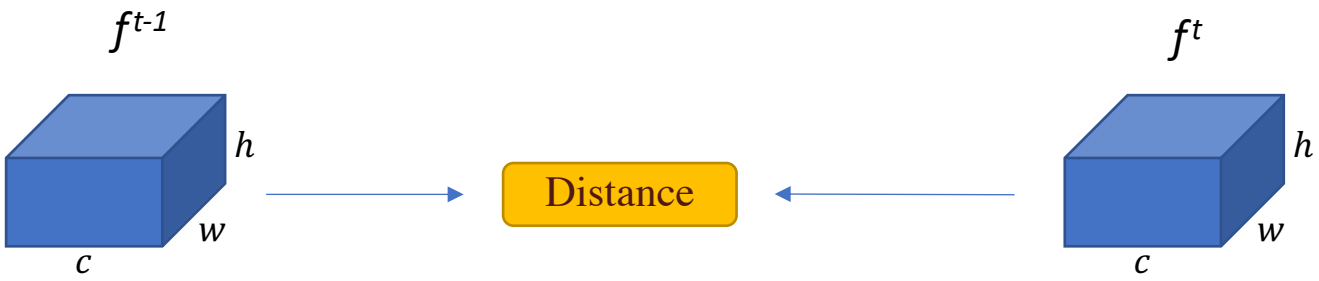
- Probabilities → weak
- Weights → Slow and heavy
- Gradients → Very slow

1. PODNet, ECCV 2020

Problems of previous constraints:

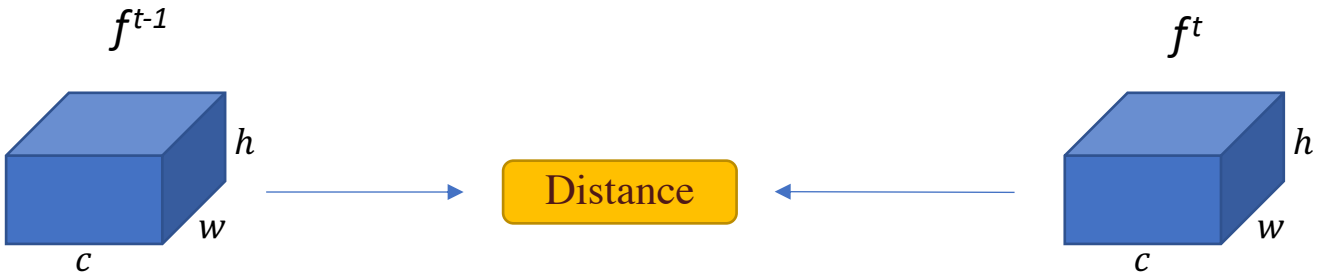
- Probabilities → weak
- Weights → Slow and heavy
- Gradients → Very slow

What if we constrain **intermediary features?**



1. PODNet, ECCV 2020

What if we constrain **intermediary features?**

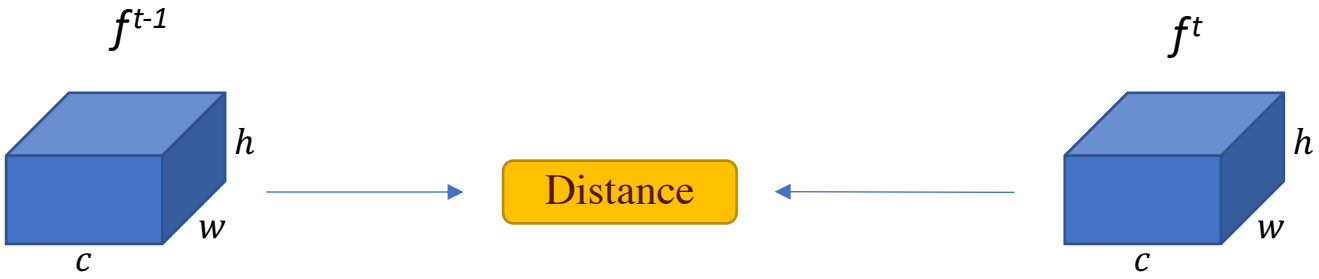


Not working!

Loss	NME	CNN
None	53.29	52.98
POD-pixels	49.74	52.34

1. PODNet, ECCV 2020

What if we constrain **intermediary features?**

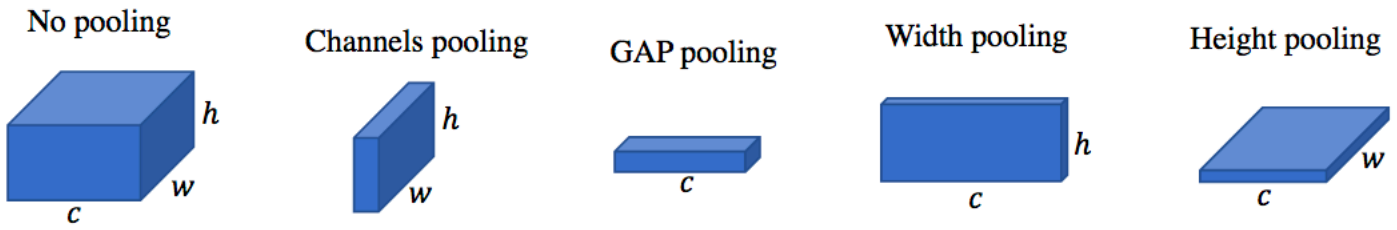


- Too much constraints ($C \times W \times H$)
- Too **sensitive** to outliers

1. PODNet, ECCV 2020

heuritech SCIENCES SORBONNE UNIVERSITÉ

Solution: matching statistics instead exact pixels

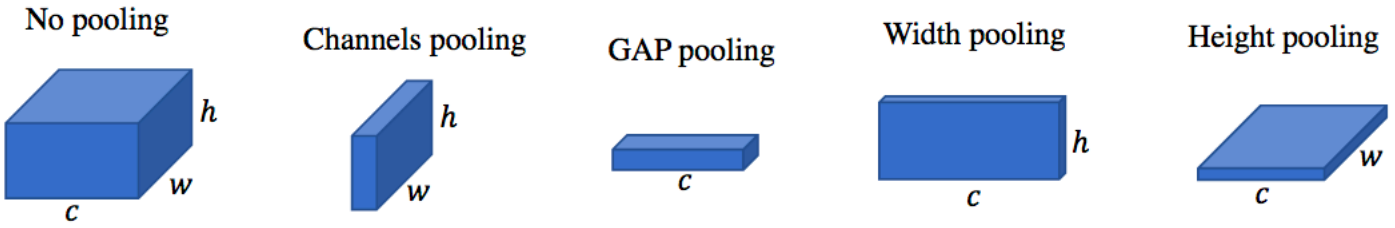


1. PODNet, ECCV 2020

heuritech

SCIENCES SORBONNE UNIVERSITÉ

Solution: matching statistics instead exact pixels



Loss	NME	CNN
None	53.29	52.98
POD-pixels	49.74	52.34
POD-channels	57.21	54.64
POD-gap	58.80	55.95
POD-width	60.92	57.51
POD-height	60.64	57.50
POD-spatial	**61.40**	**57.98**
GradCam [5]	54.13	52.48
Perceptual Style [16]	51.01	52.25

1. PODNet, ECCV 2020

heuritech

SCIENCES SORBONNE UNIVERSITÉ

50 classes + 5 x 10 classes

	CIFAR100
New classes per step	5 steps 10
*iCaRL** [33]	57.17
iCaRL	58.08 ± 0.59
BiC [38]	56.86 ± 0.46
*UCIR (NME)** [14]	63.12
UCIR (NME) [14]	63.63 ± 0.87
*UCIR (CNN)** [14]	63.42
UCIR (CNN) [14]	64.01 ± 0.91
PODNet (NME)	$\mathbf{64.48 \pm 1.32}$
PODNet (CNN)	$\mathbf{64.83 \pm 0.98}$

1. PODNet, ECCV 2020

heuritech

SCIENCES SORBONNE UNIVERSITÉ

50 classes + 10 x 5 classes

	CIFAR100	
	10 steps	5 steps
New classes per step	5	10
*iCaRL** [33]	52.57	57.17
iCaRL	53.78 ± 1.16	58.08 ± 0.59
BiC [38]	53.21 ± 1.01	56.86 ± 0.46
*UCIR (NME)** [14]	60.12	63.12
UCIR (NME) [14]	60.83 ± 0.70	63.63 ± 0.87
*UCIR (CNN)** [14]	60.18	63.42
UCIR (CNN) [14]	61.22 ± 0.69	64.01 ± 0.91
PODNet (NME)	**64.03 ± 1.30**	**64.48 ± 1.32**
PODNet (CNN)	**63.19 ± 1.16**	**64.83 ± 0.98**

1. PODNet, ECCV 2020

50 classes + 25 x 2 classes

	CIFAR100		
	25 steps	10 steps	5 steps
New classes per step	2	5	10
*iCaRL** [33]	—	52.57	57.17
iCaRL	50.60 ± 1.06	53.78 ± 1.16	58.08 ± 0.59
BiC [38]	48.96 ± 1.03	53.21 ± 1.01	56.86 ± 0.46
*UCIR (NME)** [14]	—	60.12	63.12
UCIR (NME) [14]	56.82 ± 0.19	60.83 ± 0.70	63.63 ± 0.87
*UCIR (CNN)** [14]	—	60.18	63.42
UCIR (CNN) [14]	57.57 ± 0.23	61.22 ± 0.69	64.01 ± 0.91
PODNet (NME)	**62.71 ± 1.26**	**64.03 ± 1.30**	**64.48 ± 1.32**
PODNet (CNN)	**60.72 ± 1.36**	**63.19 ± 1.16**	**64.83 ± 0.98**

1. PODNet, ECCV 2020

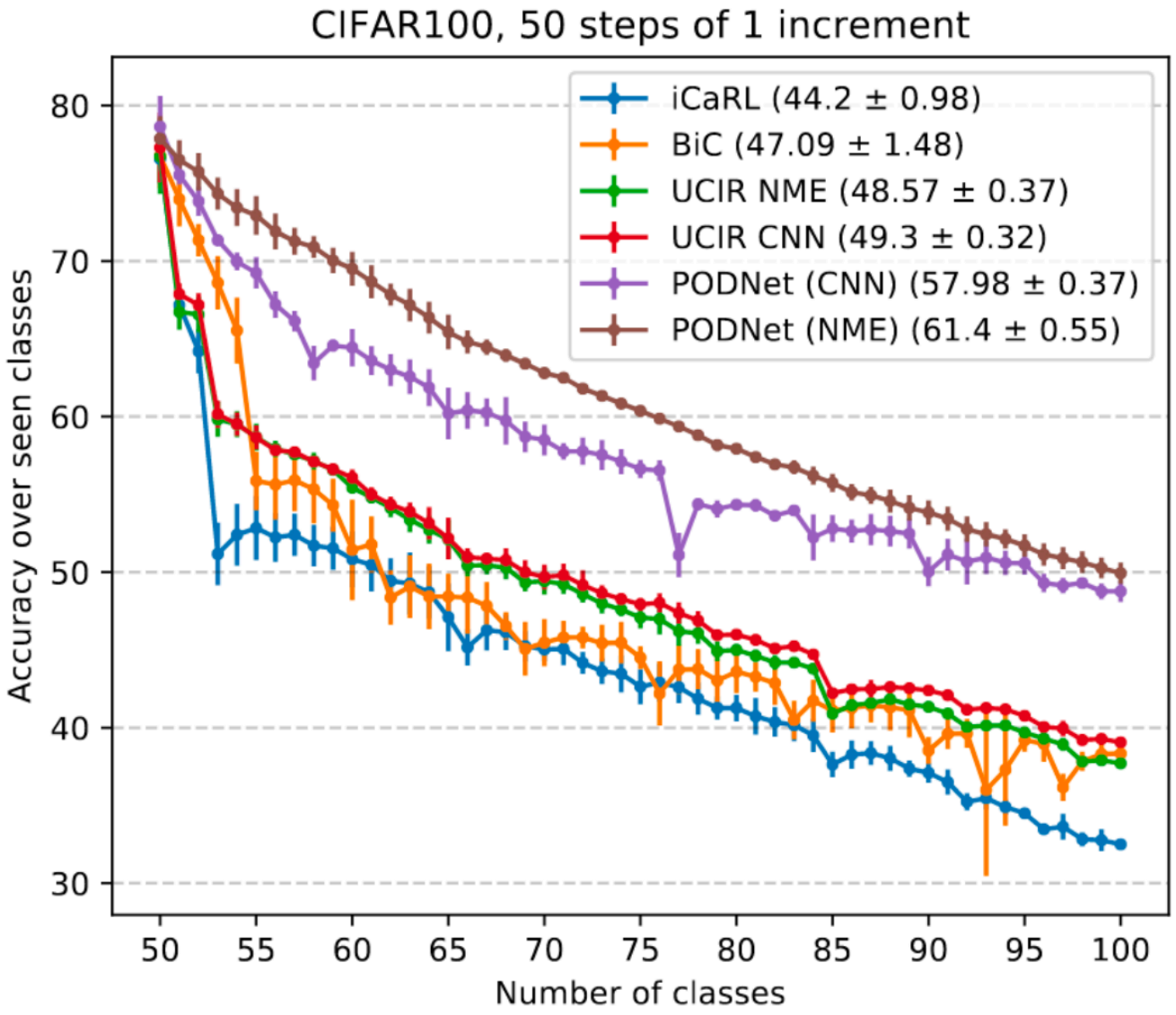
heuritech

SCIENCES SORBONNE UNIVERSITÉ

50 classes + 50 x 1 classes

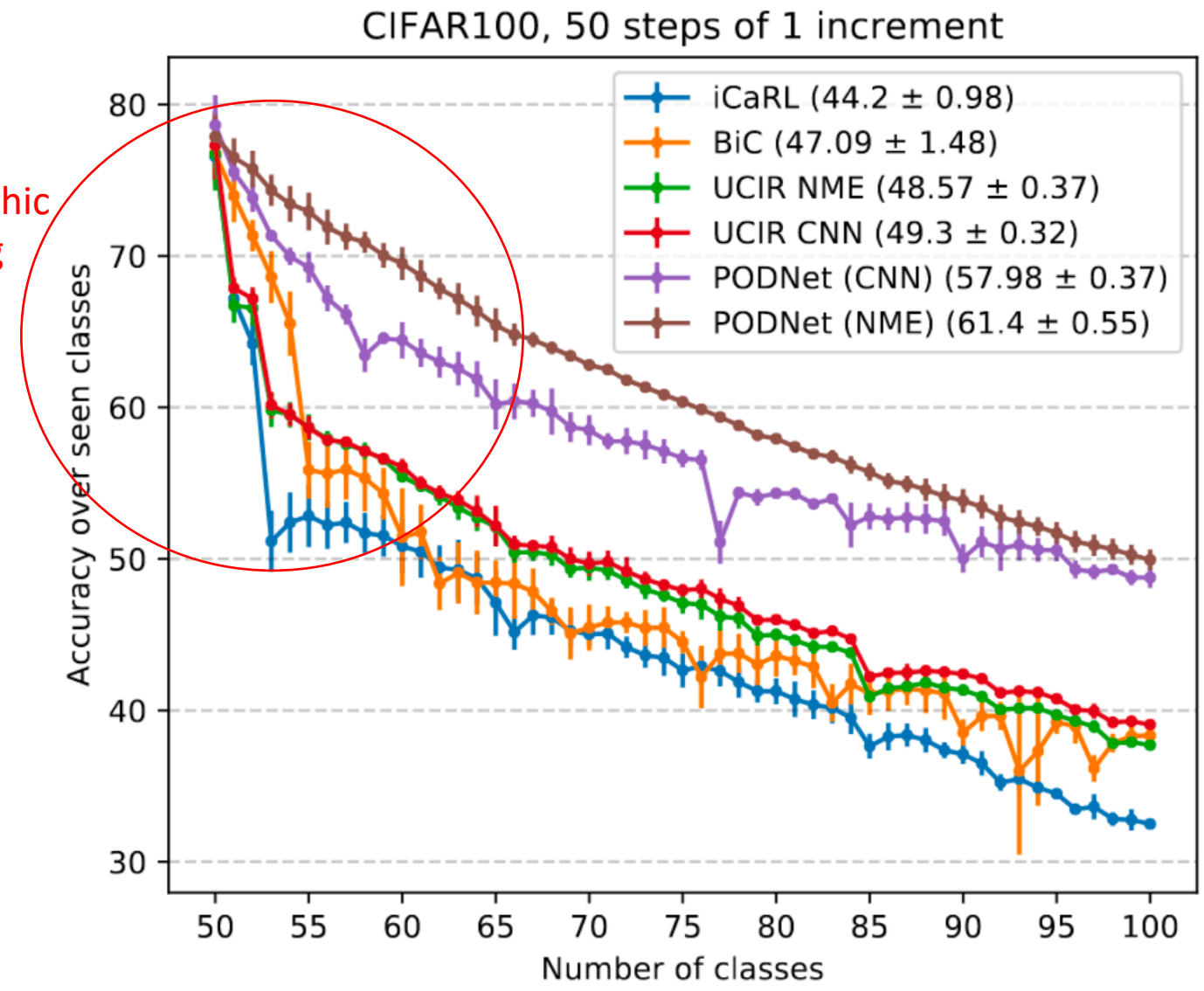
	CIFAR100			
	50 steps	25 steps	10 steps	5 steps
New classes per step	1	2	5	10
*iCaRL** [33]	—	—	52.57	57.17
iCaRL	44.20 ± 0.98	50.60 ± 1.06	53.78 ± 1.16	58.08 ± 0.59
BiC [38]	47.09 ± 1.48	48.96 ± 1.03	53.21 ± 1.01	56.86 ± 0.46
*UCIR (NME)** [14]	—	—	60.12	63.12
UCIR (NME) [14]	48.57 ± 0.37	56.82 ± 0.19	60.83 ± 0.70	63.63 ± 0.87
*UCIR (CNN)** [14]	—	—	60.18	63.42
UCIR (CNN) [14]	49.30 ± 0.32	57.57 ± 0.23	61.22 ± 0.69	64.01 ± 0.91
PODNet (NME)	**61.40 ± 0.68**	**62.71 ± 1.26**	**64.03 ± 1.30**	**64.48 ± 1.32**
PODNet (CNN)	**57.98 ± 0.46**	**60.72 ± 1.36**	**63.19 ± 1.16**	**64.83 ± 0.98**

1. PODNet, ECCV 2020



1. PODNet, ECCV 2020

Catastrophic forgetting



2. PLOP

heuritech SCIENCES SORBONNE UNIVERSITÉ

PLOP: Learning without Forgetting for Continual Semantic Segmentation

Arthur Douillard Yifu Chen Arnaud Dapogny Matthieu Cord

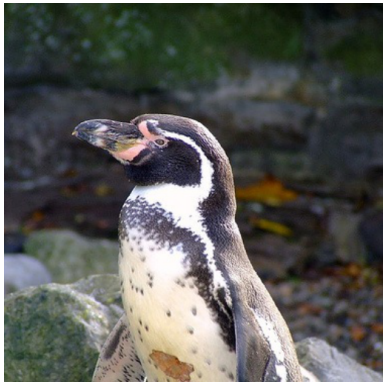
Constraints + Pseudo-labeling

2. PLOP

Semantic Segmentation → each pixel is labeled








2. PLOP

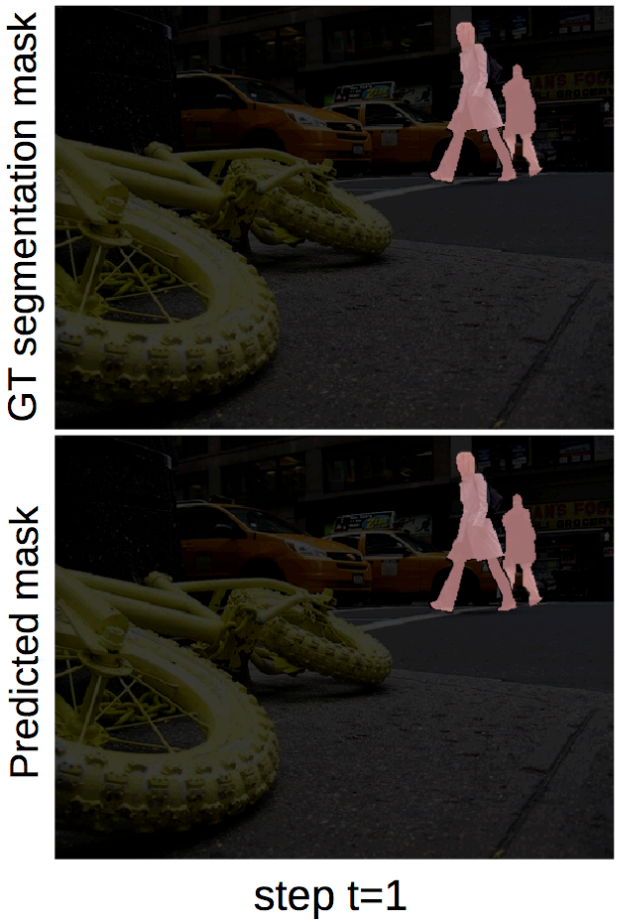
heuritech

SCIENCES SORBONNE UNIVERSITÉ

Semantic Segmentation → each pixel is labeled

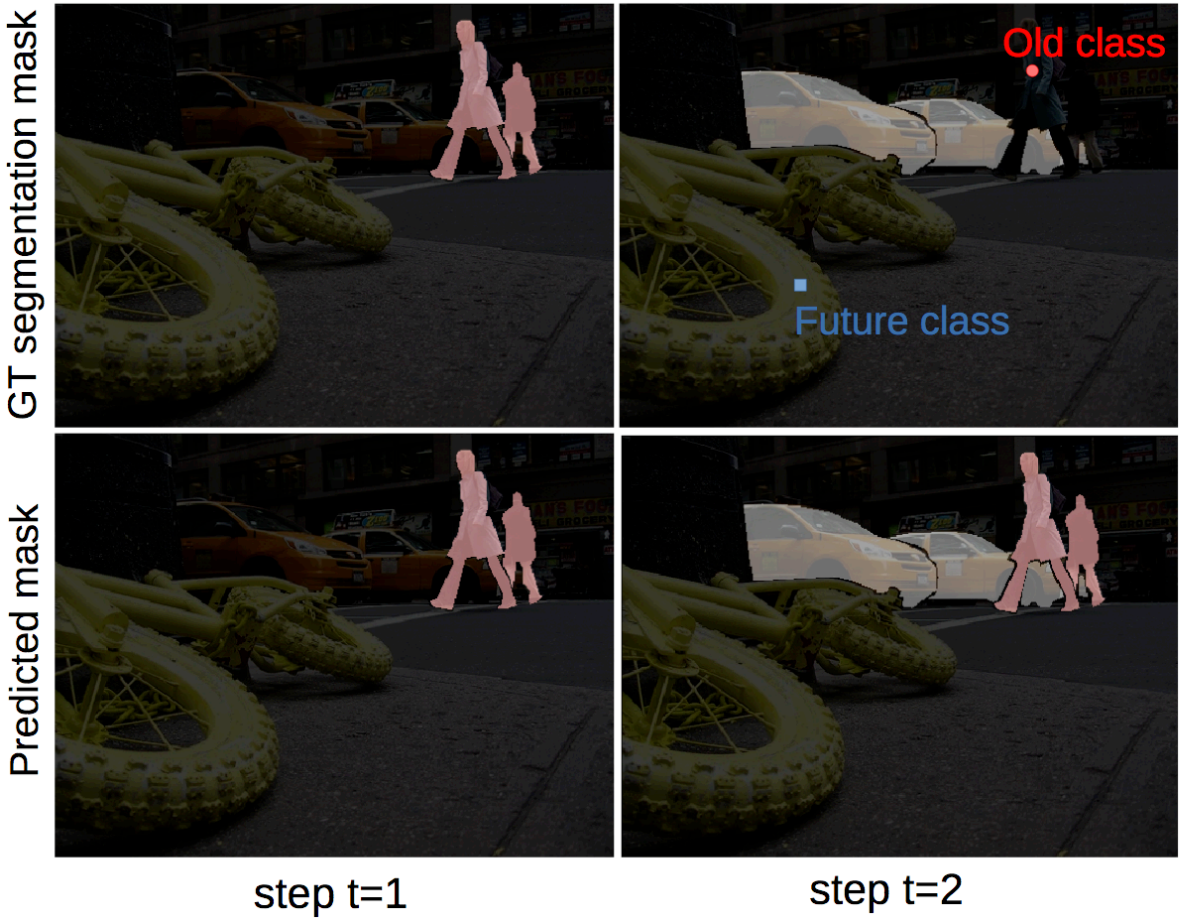
Continual Semantic Segmentation?

2. PLOP



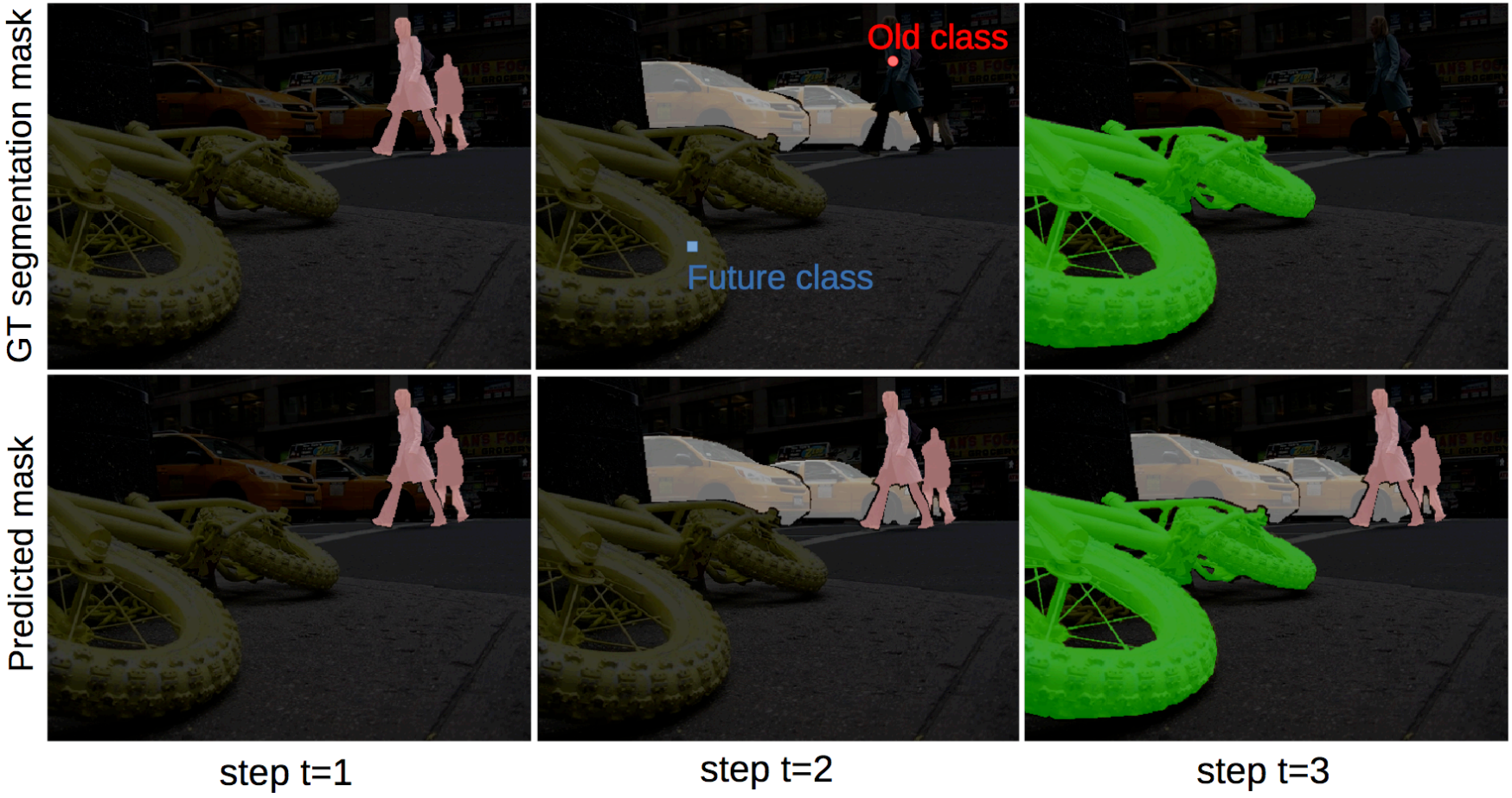
[13]: Cermelli et al., Modeling the Background for Incremental in Semantic Segmentation, 2020

2. PLOP



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

2. PLOP



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

2. PLOP

Problems:

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

2. PLOP

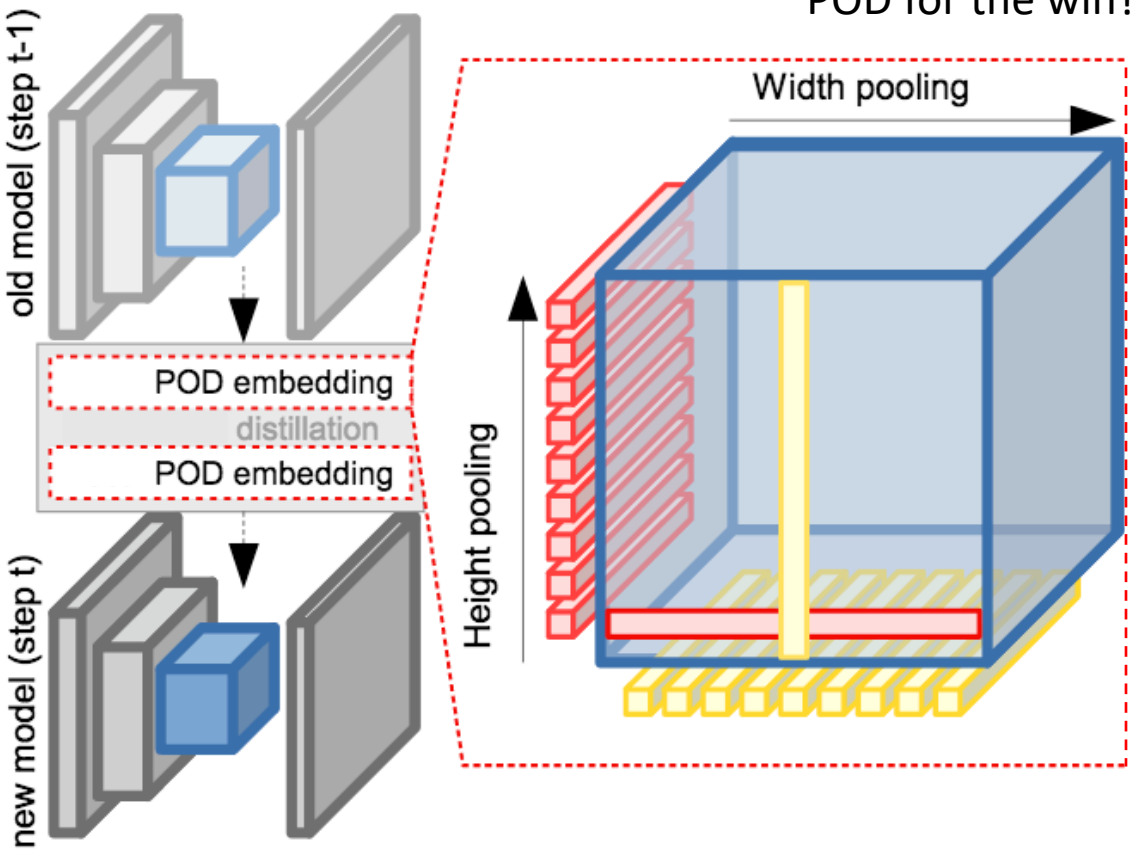
Problems:

- **<u>Forgetting is particularly strong</u>**

- Images at task t are partially labeled

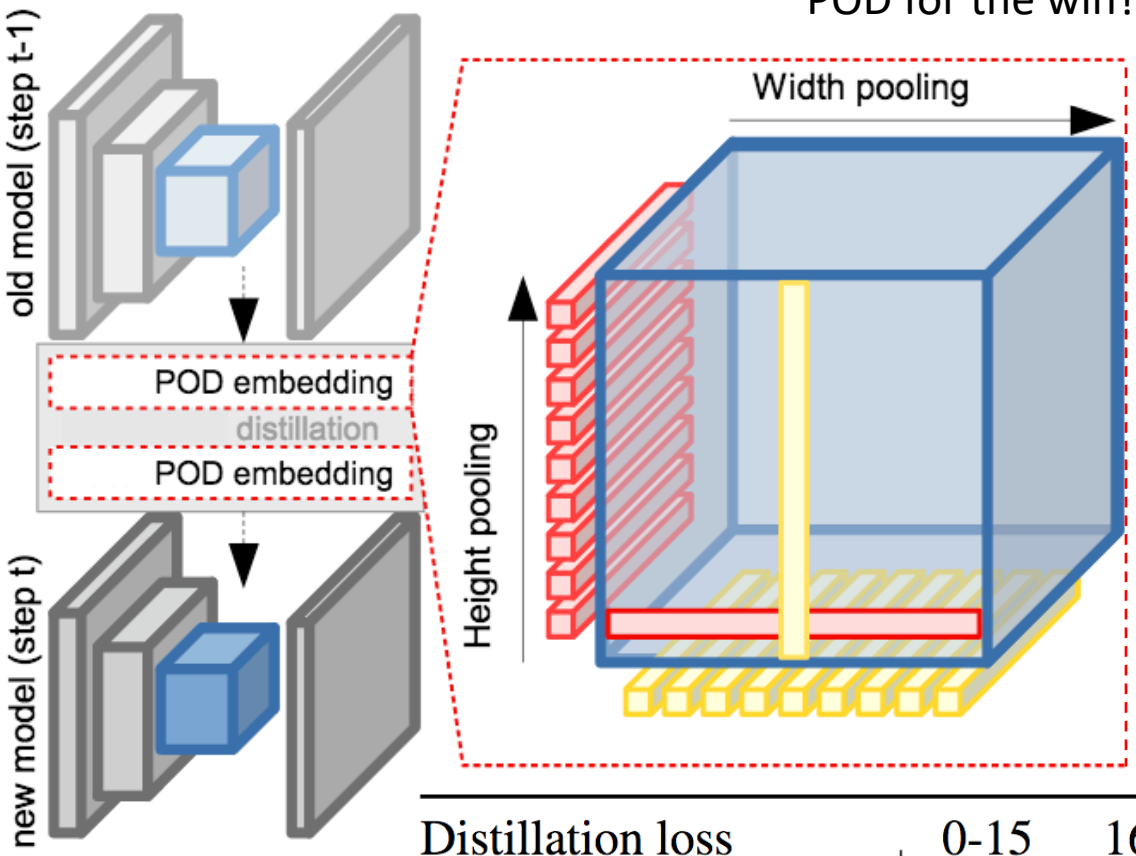
2. PLOP

POD for the win?



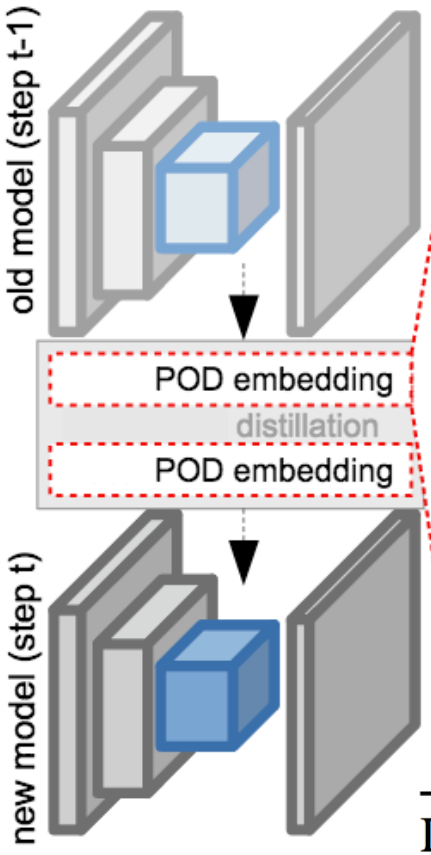
2. PLOP

POD for the win?

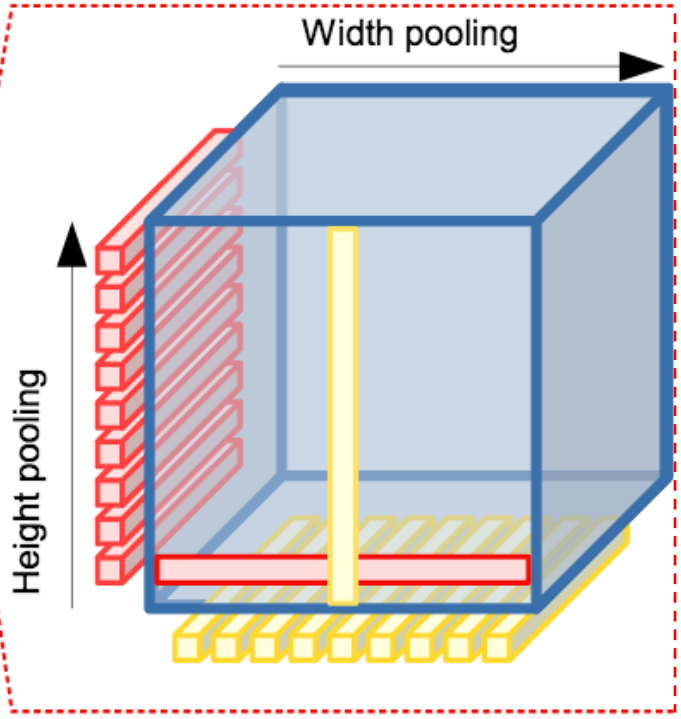


Distillation loss	0-15	16-20	*all*	*avg*
Knowledge Distllation	29.72	4.42	23.69	49.18
UNKD	34.85	5.26	27.80	46.39
POD	43.94	4.82	34.62	53.35

2. PLOP



POD for the win?

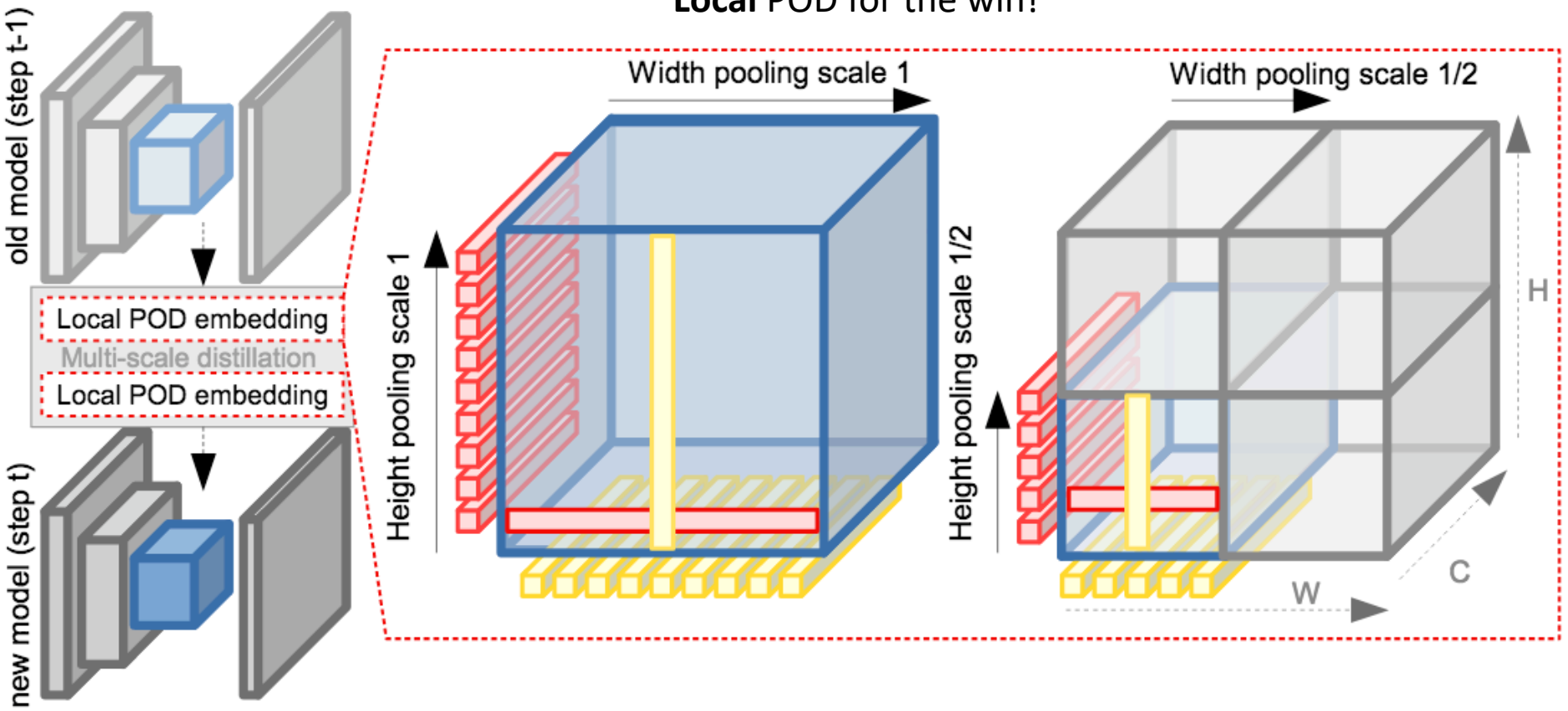


Segmentation
≠
Classification

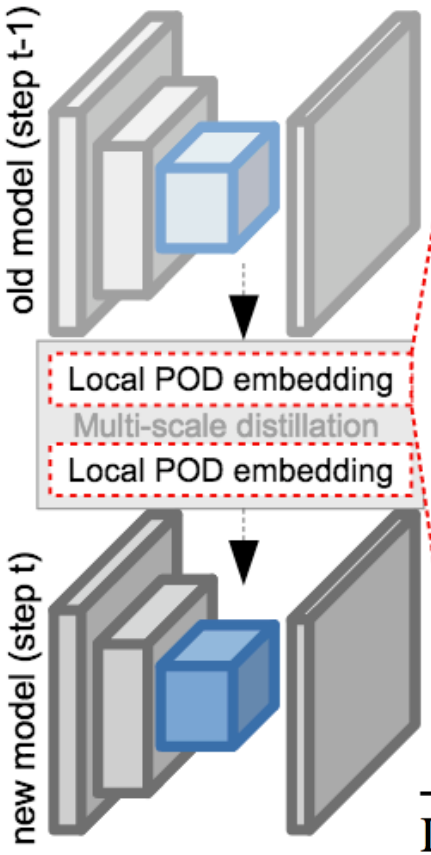
Distillation loss	0-15	16-20	*all*	*avg*
Knowledge Distllation	29.72	4.42	23.69	49.18
UNKD	34.85	5.26	27.80	46.39
POD	43.94	4.82	34.62	53.35

2. PLOP

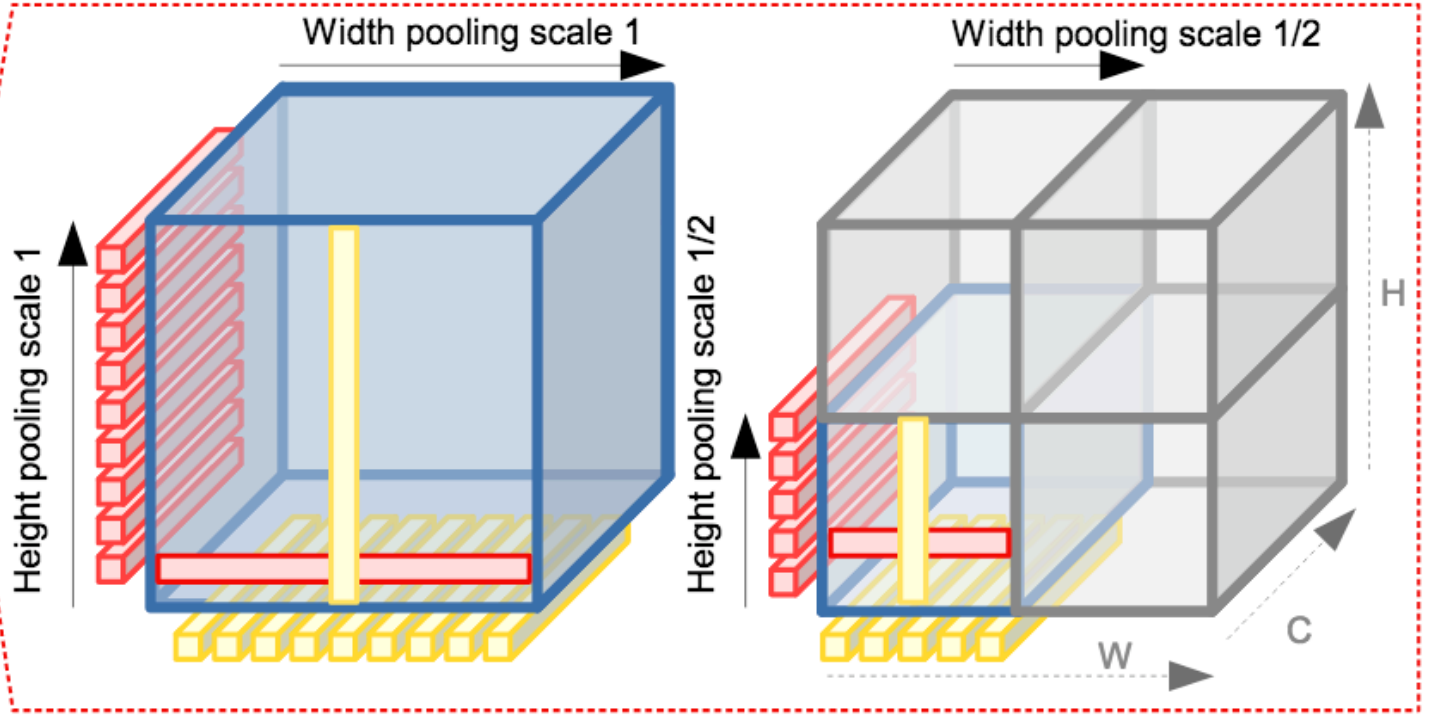
Local POD for the win!



2. PLOP



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**

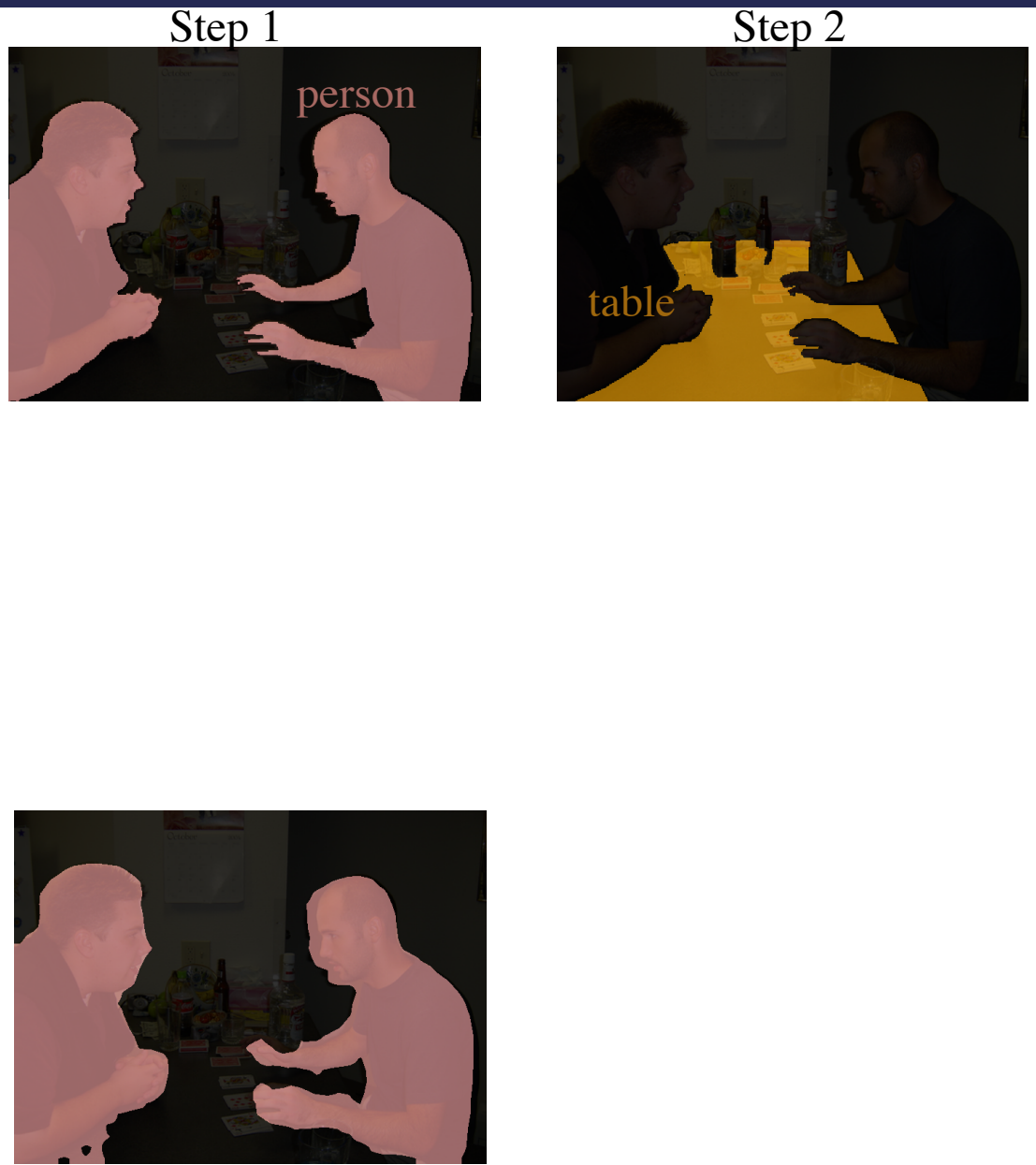
2. PLOP

Problems:

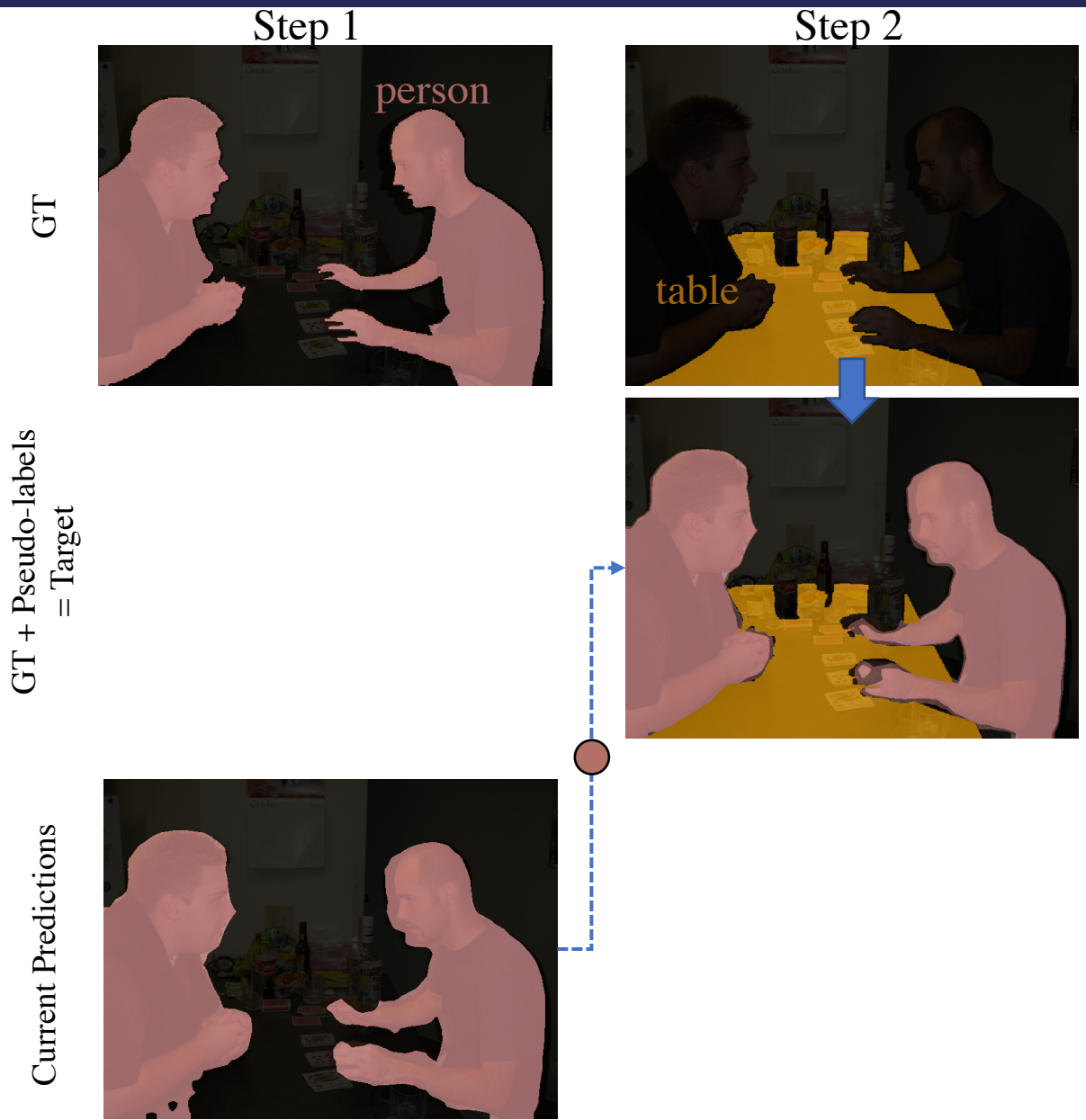
- Forgetting is particularly strong
- **Images at task t are partially labeled**

2. PLOP

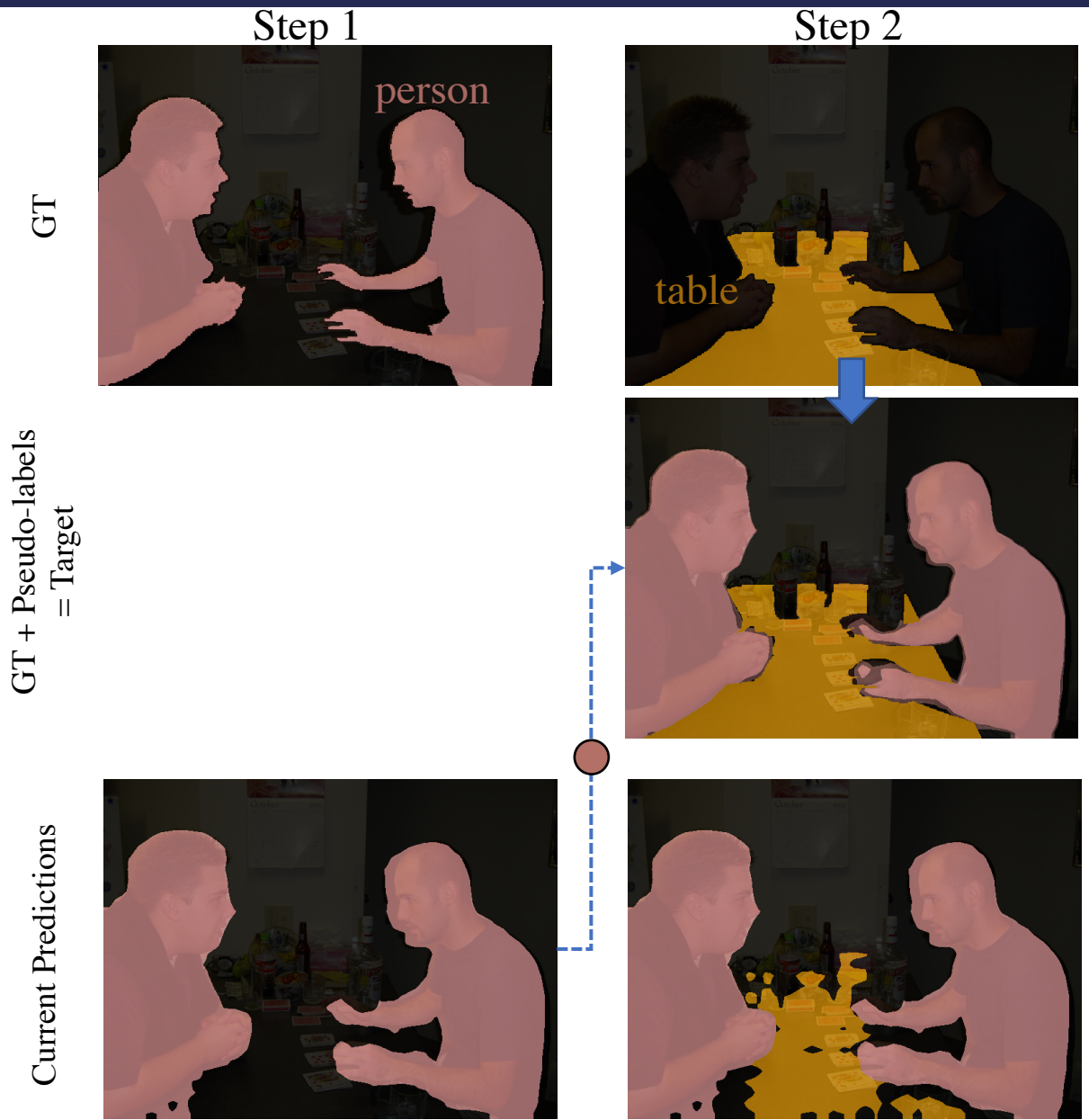
2. PLOP



2. PLOP


Step 1
Step 2
GT
person
table
GT + Pseudo-labels
= Target
Current Predictions

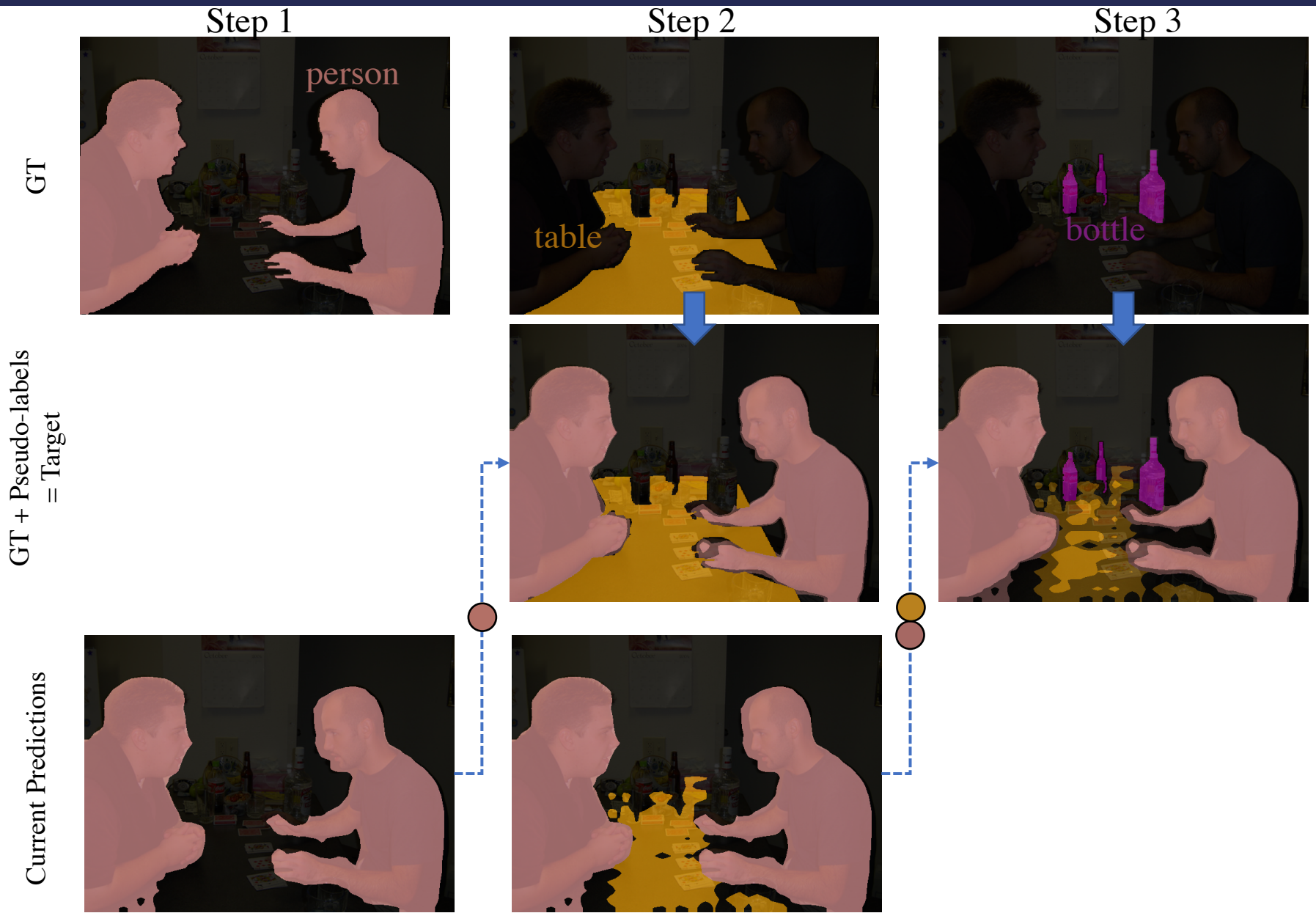
2. PLOP


Step 1
Step 2
GT
person
table
GT + Pseudo-labels
= Target
Current Predictions

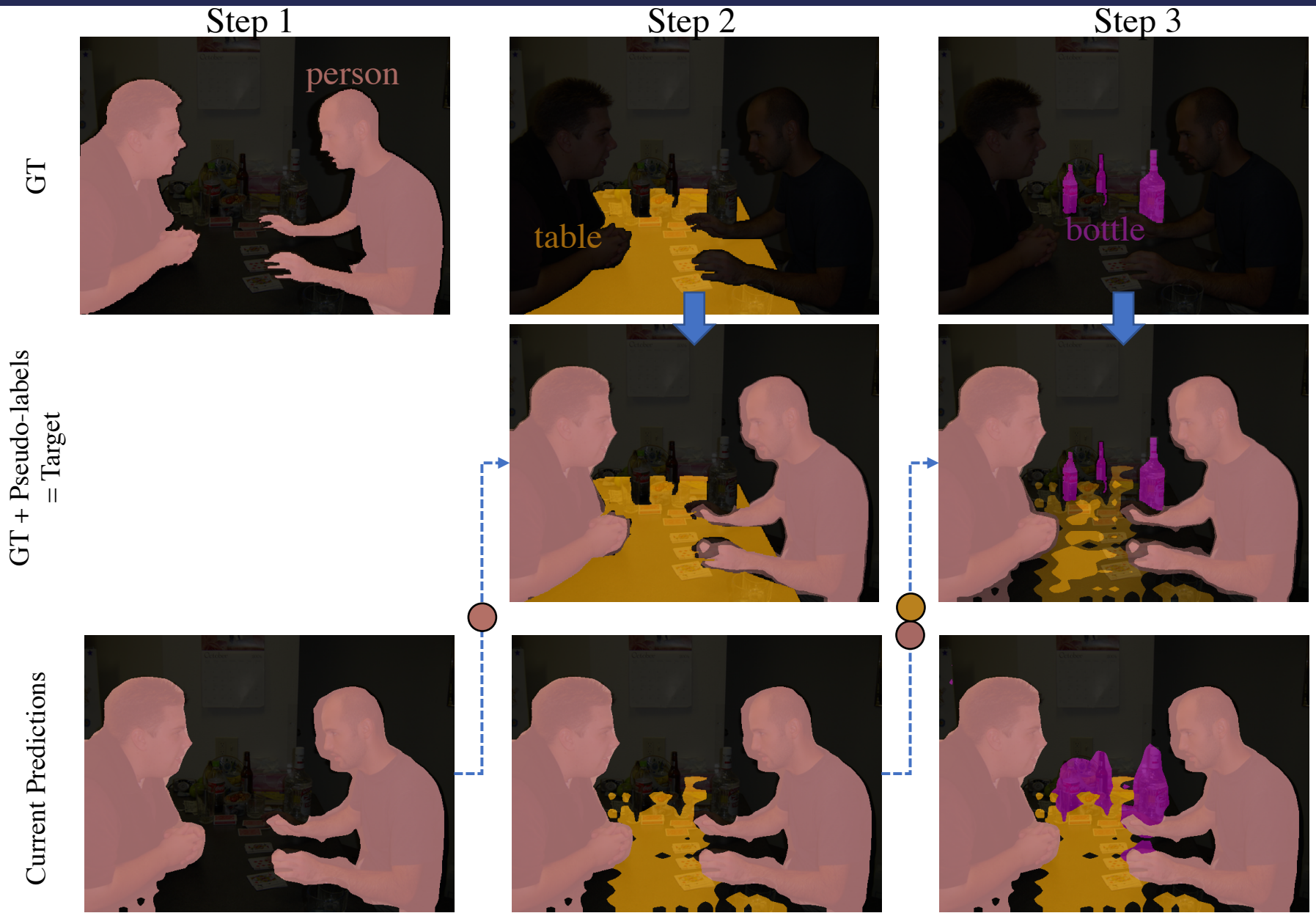
2. PLOP


Step 1
Step 2
Step 3
GT
person
table
bottle
GT + Pseudo-labels
= Target
Current Predictions

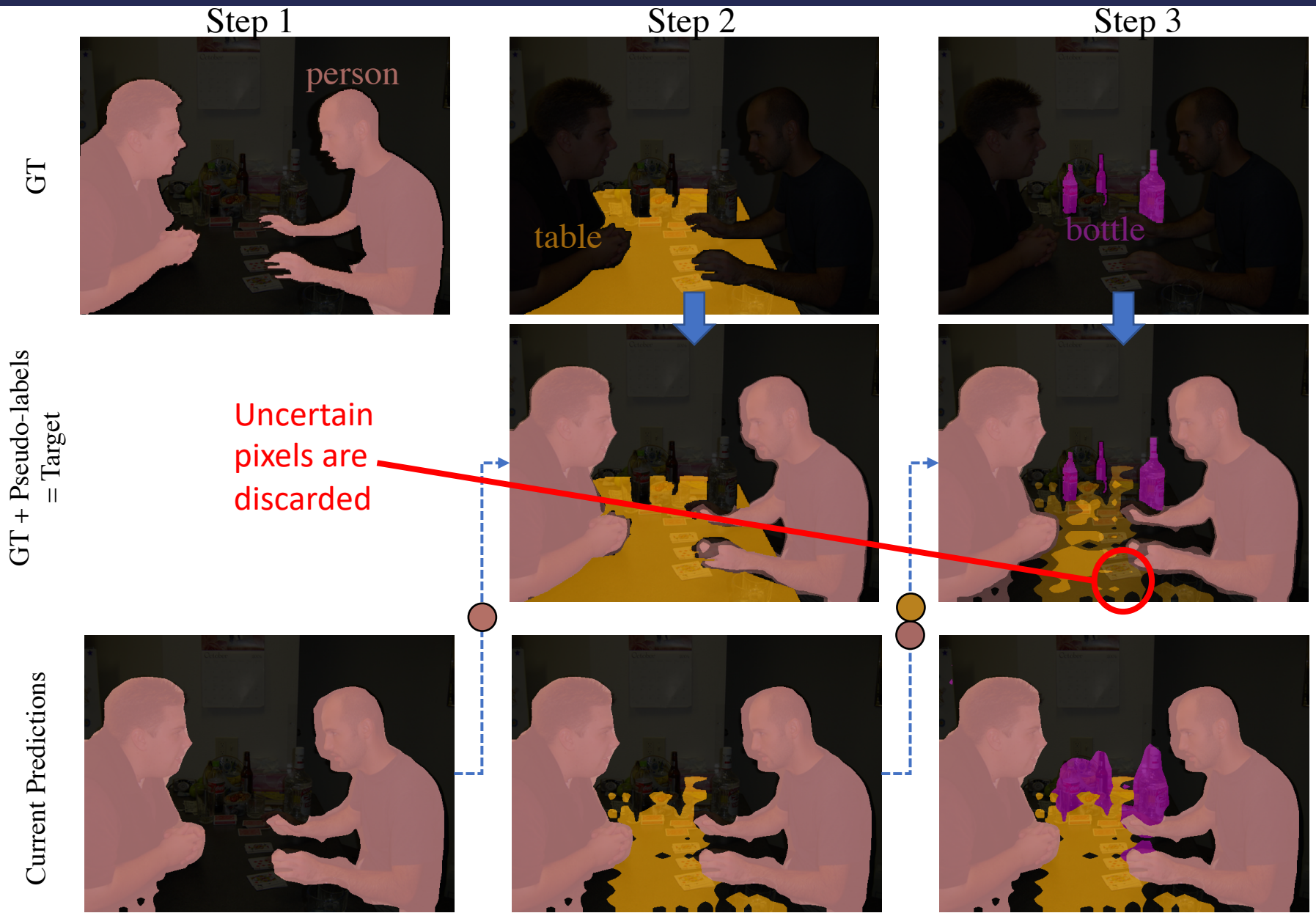
2. PLOP


Step 1
Step 2
Step 3
GT
person
table
bottle
GT + Pseudo-labels
= Target
Current Predictions

2. PLOP


Step 1
Step 2
Step 3
GT
person
table
bottle
GT + Pseudo-labels
= Target
Current Predictions

2. PLOP


Step 1
Step 2
Step 3
GT
person
table
bottle
GT + Pseudo-labels
= Target
Uncertain
pixels are
discarded
Current Predictions

2. PLOP

heuritech

SCIENCES SORBONNE UNIVERSITÉ

Discarding low-confidence samples to avoid overpredicting old classes

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

2. PLOP

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*

2. PLOP

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**

2. PLOP

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**

2. PLOP


Step 1
1-15
MiB
PLOP
MiB
PLOP
person
table
bird
Image
GT
Image
GT

2. PLOP

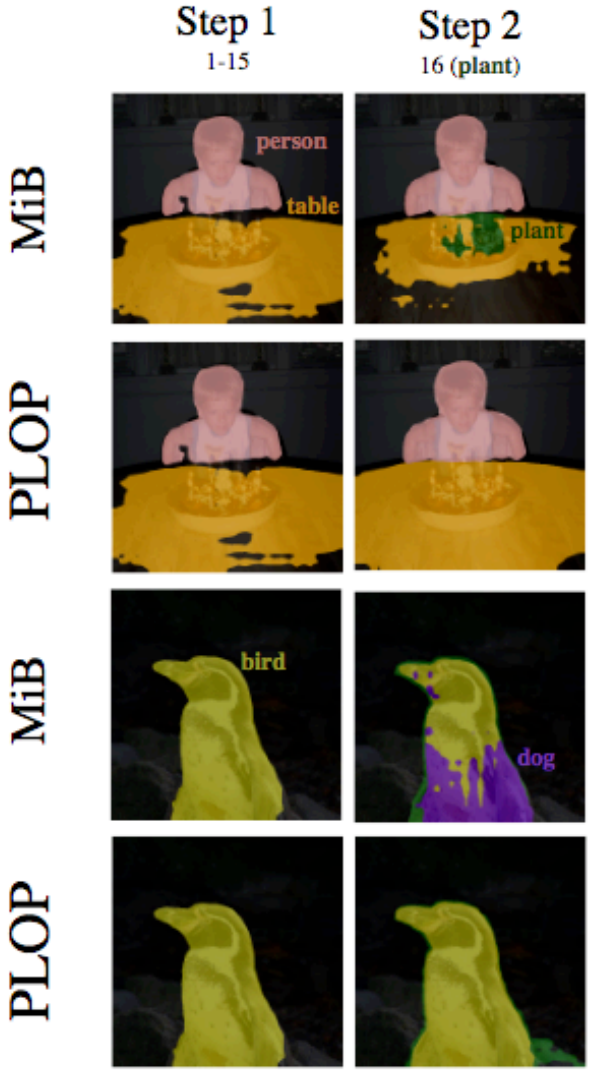
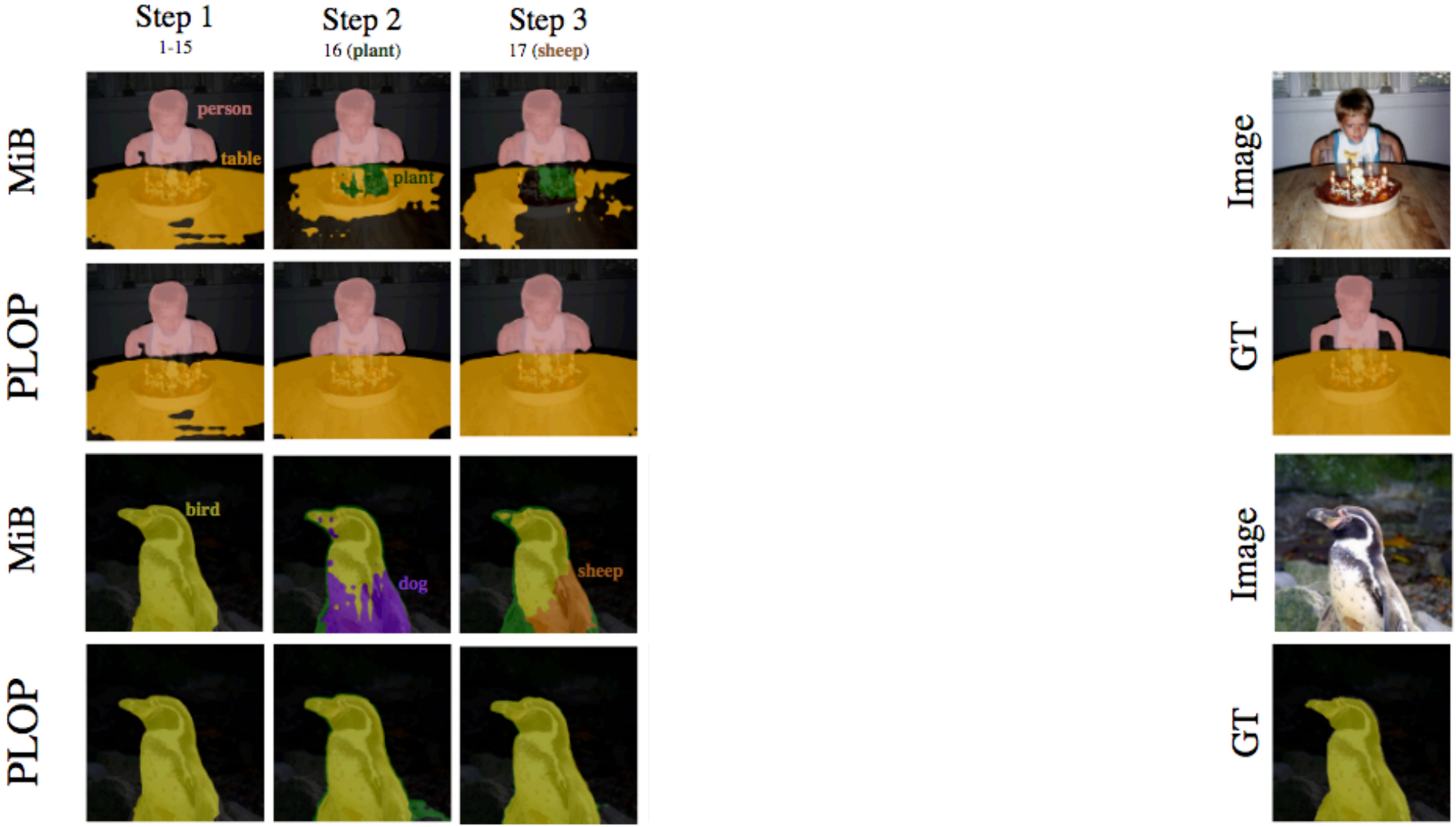
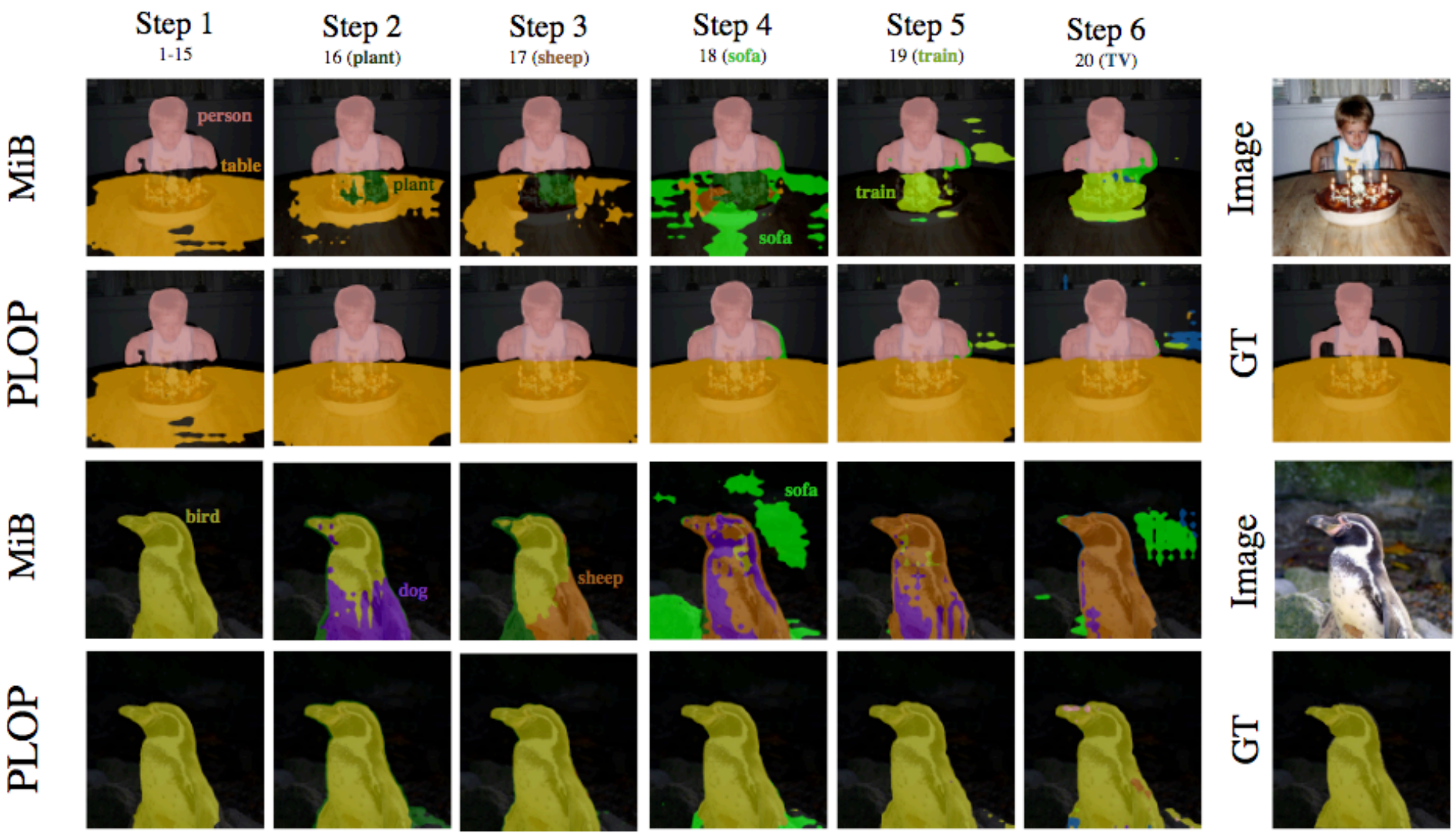

Step 1
1-15
Step 2
16 (plant)
MiB
person
table
plant
PLOP
MiB
bird
dog
PLOP


Image
GT
Image
GT

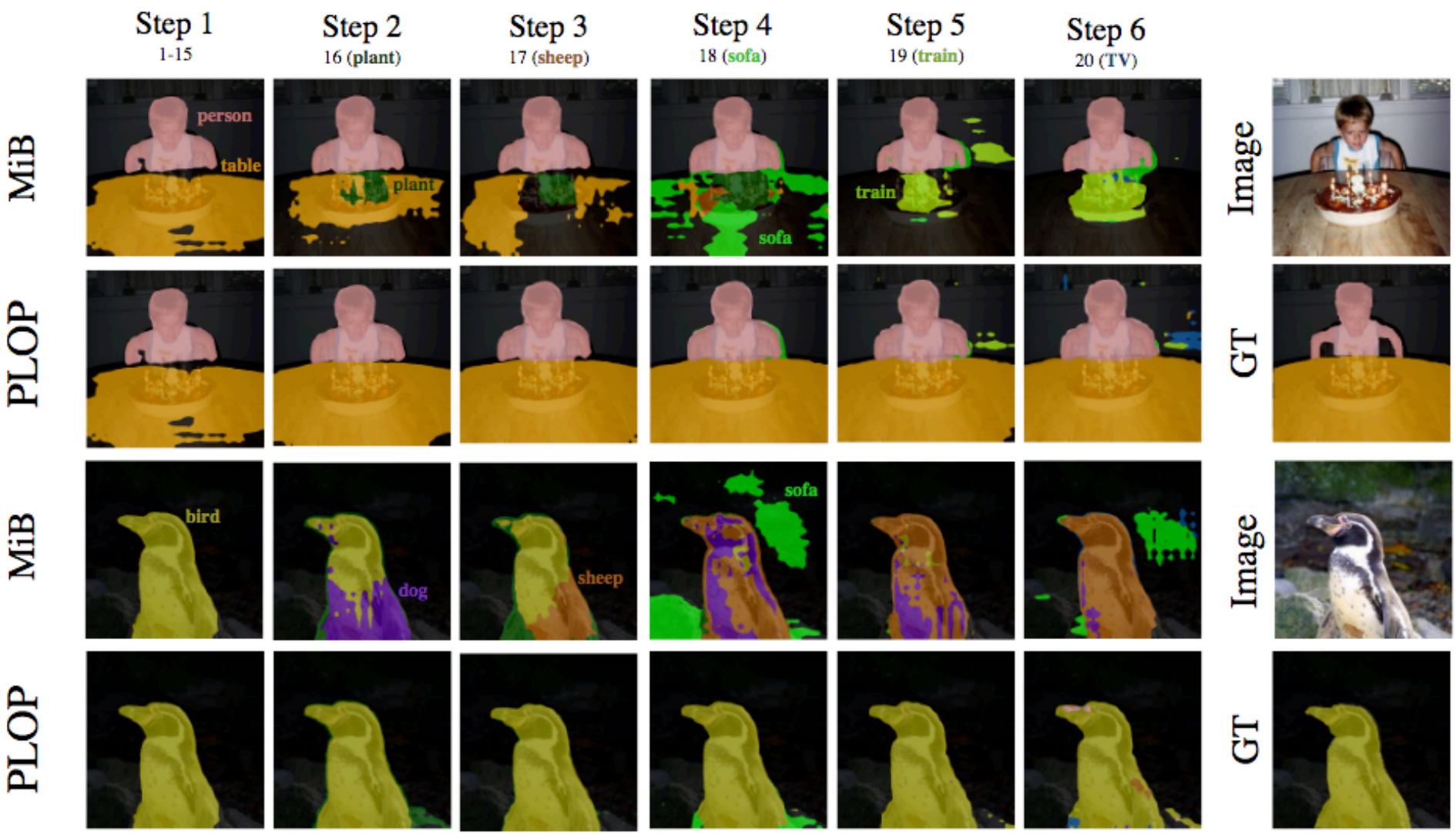
2. PLOP


Step 1
1-15
Step 2
16 (plant)
Step 3
17 (sheep)
MiB
person
table
plant
PLOP
MiB
bird
dog
sheep
PLOP
Image
GT
Image
GT

2. PLOP

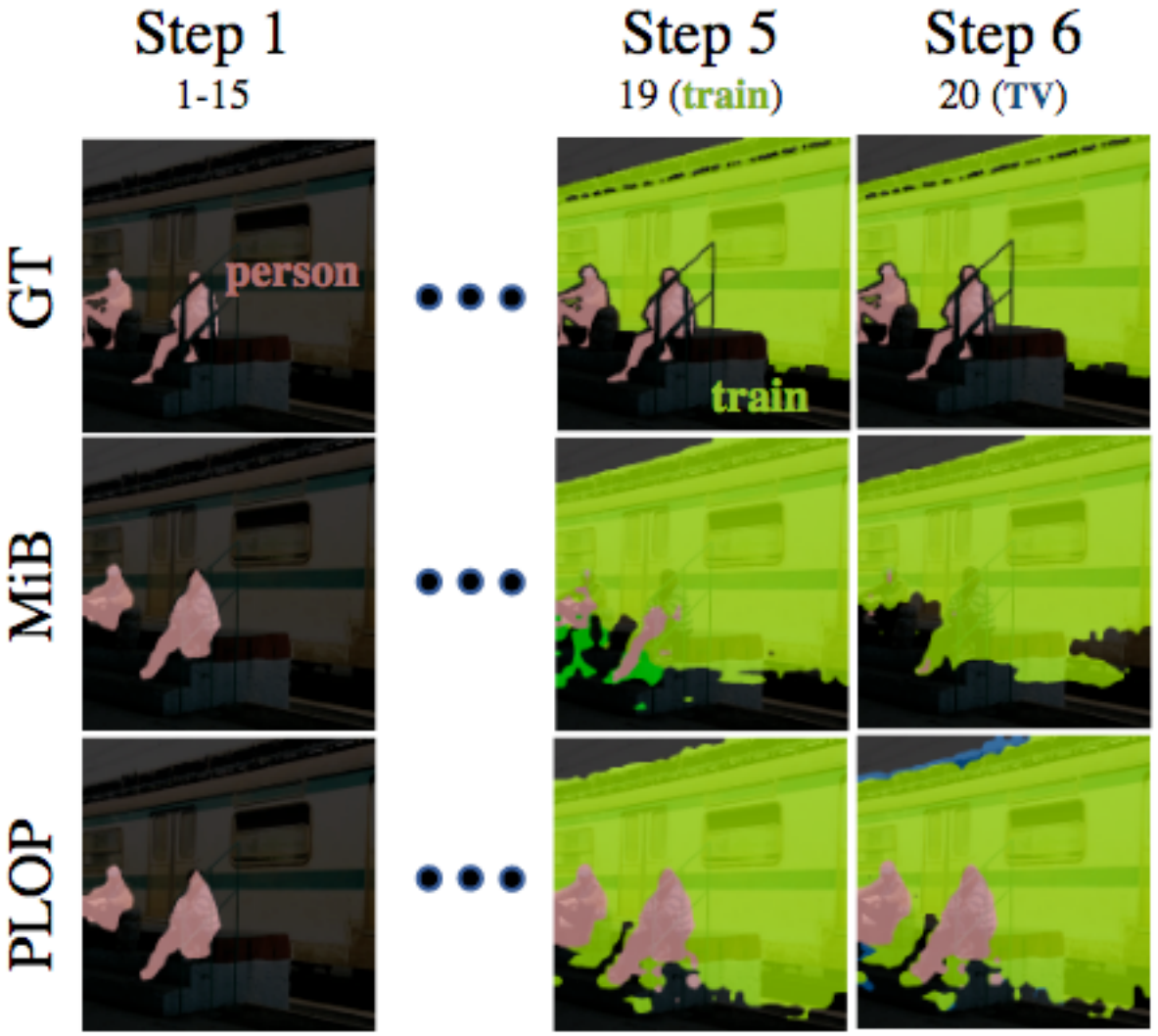

Step 1
1-15
Step 2
16 (plant)
Step 3
17 (sheep)
Step 4
18 (sofa)
Step 5
19 (train)
Step 6
20 (TV)
MiB
PLOP
MiB
PLOP
person
table
plant
sofa
train
bird
dog
sheep
sofa
Image
GT
Image
GT

2. PLOP


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
person
table
plant
sofa
train
bird
dog
sheep
sofa
Image
GT
Image
GT

2. PLOP

When a class appear only latter in the image → **background shift**



What are your questions?

References

References

heuritech

SCIENCES SORBONNE UNIVERSITÉ

[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