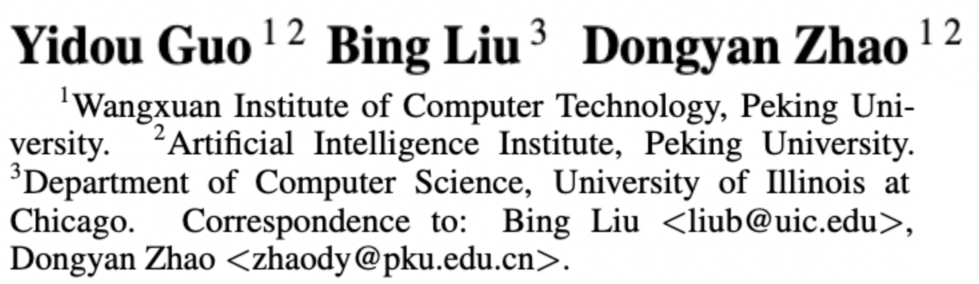
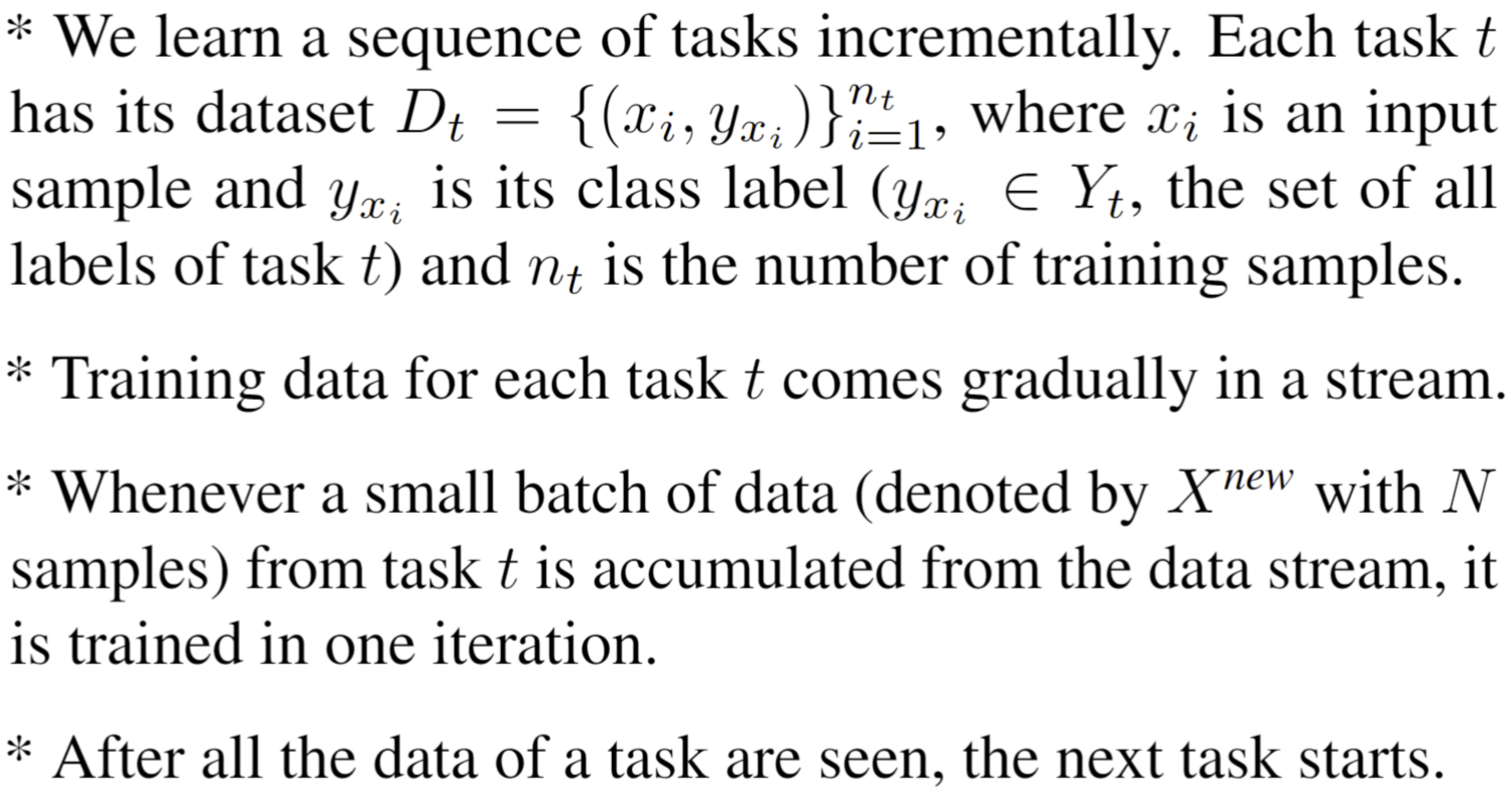
OCM: **O**nline **C**ontinual learning based on **M**utual information Maximization



Yidou Guo [1 2] **Bing Liu** [3] **Dongyan Zhao** [1 2]

[1]Wangxuan Institute of Computer Technology, Peking University. [2]Artificial Intelligence Institute, Peking University. [3]Department of Computer Science, University of Illinois at Chicago. Correspondence to: Bing Liu <liub@uic.edu>, Dongyan Zhao <zhaody@pku.edu.cn>.

Online CL: Problem statement



* We learn a sequence of tasks incrementally. Each task t has its dataset $D_t = \{(x_i, y_{x_i})\}_{i=1}^{n_t}$, where x_i is an input sample and y_{x_i} is its class label ($y_{x_i} \in Y_t$, the set of all labels of task t) and n_t is the number of training samples.

* Training data for each task t comes gradually in a stream.

* Whenever a small batch of data (denoted by X^{new} with N samples) from task t is accumulated from the data stream, it is trained in one iteration.

* After all the data of a task are seen, the next task starts.

Training is done in one epoch

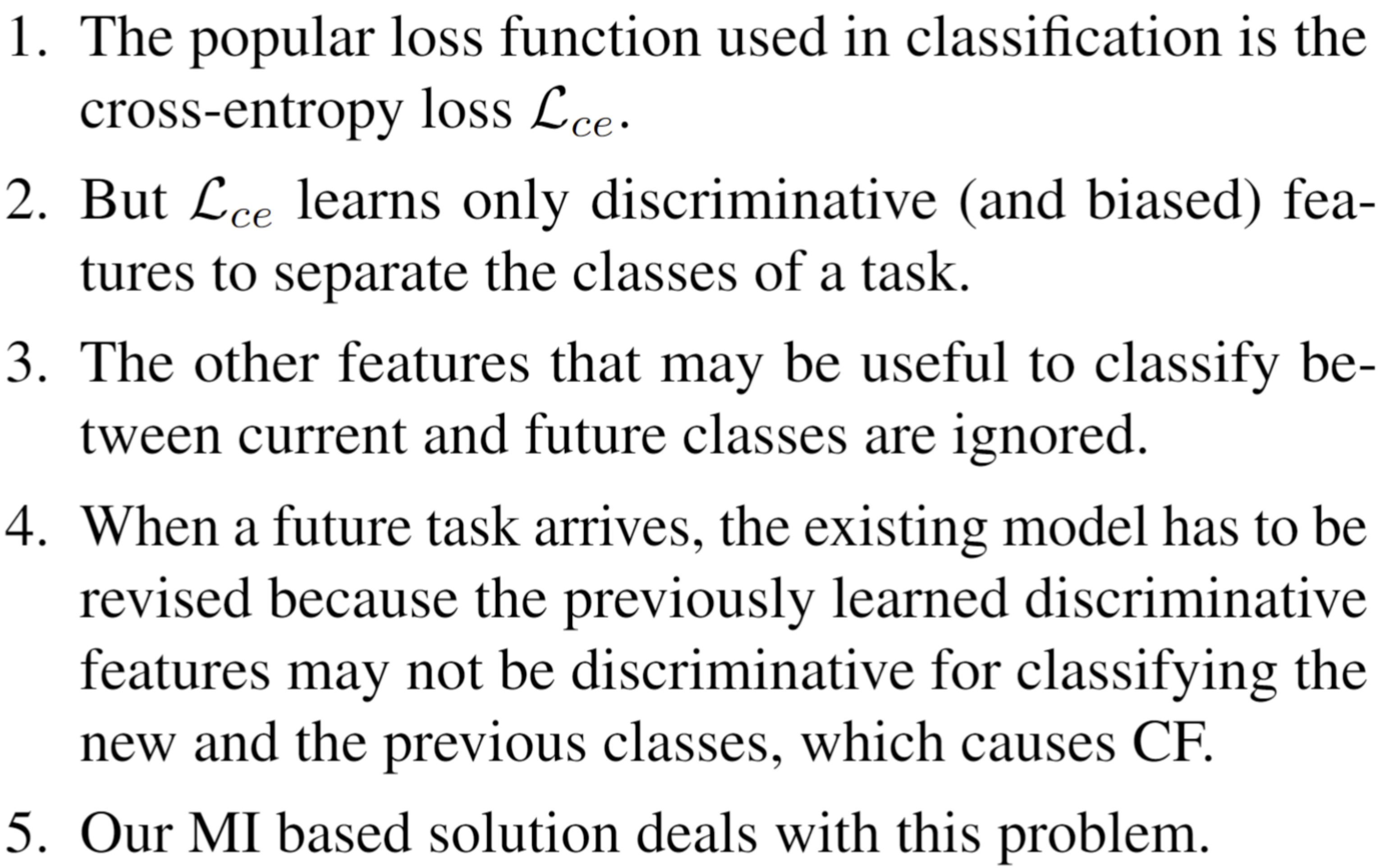
Online CL: Replay

* A mini-batch used in training consists of X^{new} and X^{buf}, where X^{buf} of size N_b is sampled from the memory buffer $\mathcal{M}$.

* $\mathcal{M}$ saves a small set of training samples of seen tasks.

* Note that, before seeing all the data of the current task t, $\mathcal{M}$ have already saved some data from task t sampled from the data stream of task t seen so far.

Problem with cross entropy loss



1. The popular loss function used in classification is the cross-entropy loss $\mathcal{L}_{ce}$.
2. But $\mathcal{L}_{ce}$ learns only discriminative (and biased) features to separate the classes of a task.
3. The other features that may be useful to classify between current and future classes are ignored.
4. When a future task arrives, the existing model has to be revised because the previously learned discriminative features may not be discriminative for classifying the new and the previous classes, which causes CF.
5. Our MI based solution deals with this problem.

Proposition 1. Minimal cross-entropy does not imply that all possible features are learned.

Proposition 2. Features not learned may cause CF in continual learning.

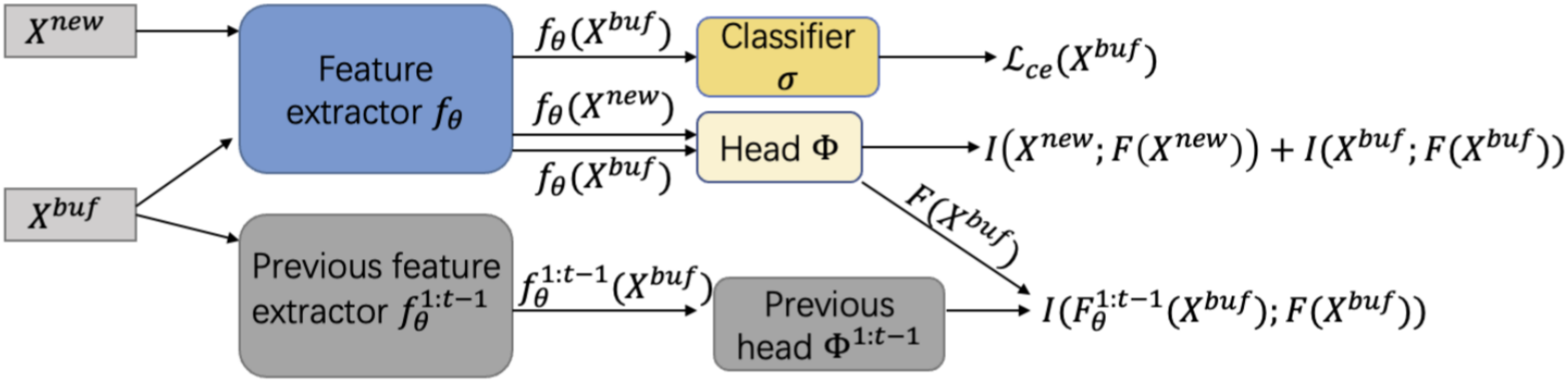
Remark: We need to learn & use as many features as possible, i.e., learning holistic representations.

Online CL: Mutual information (MI) maximization

- **OCM**: **O**nline **C**ontinual learning based on **M**utual information Maximization.
  - Both batch CL and online CL have not used mutual information (MI) to address *catastrophic forgetting* (CF)
- Objective: dealing with CF in the CIL setting using MI maximization
  - Preventing information/feature loss in feature learning
  - Preserving previously learned knowledge
- A new training strategy based on theoretical analysis
  - OCM also has a new data augmentation method called local rotation.

Guo, Liu, and Zhao. Online Continual Learning through Mutual Information Maximization. ICML-2022

OCMM architecture



X^{new}
X^{buf}
Feature extractor f_θ
Previous feature extractor $f_\theta^{1:t-1}$
$f_\theta(X^{buf})$
$f_\theta(X^{new})$
$f_\theta(X^{buf})$
$f_\theta^{1:t-1}(X^{buf})$
Classifier σ
Head Φ
Previous head $\Phi^{1:t-1}$
$\mathcal{L}_{ce}(X^{buf})$
$I(X^{new}; F(X^{new})) + I(X^{buf}; F(X^{buf}))$
$F(X^{buf})$
$I(F_\theta^{1:t-1}(X^{buf}); F(X^{buf}))$

Figure 1. Architecture of OCM. $\mathcal{L}_{ce}$: cross-entropy loss.

The final objective

$$
\begin{aligned}
&\max_{\theta,\sigma,\Phi} -\mathcal{L}_{ce}(X^{buf}) + \{I(X^{new}; F(X^{new})) + I(X^{buf}; F(X^{buf})) + \\
&I(F^{1:t-1}(X^{buf}); F(X^{buf}))\} \\
&\approx \max_{\theta,\sigma}\{\mathcal{L}_{ce}(\{x_i^b, y_{x_i^b}\}_{i=1}^{N_b})\} + \max_{\theta,\Phi}\{3\log(16) + 2\log(N_b) + \\
&\log(N) + \text{InfoNCE}(\{\{x_{i,c}, y_{x_{i,c}}\}_{c=1}^{16}\}_{i=1}^{N}; g^*) + \text{InfoNCE}(\{\{x_{i,c}^b \\
&, y_{x_{i,c}^b}\}_{c=1}^{16}\}_{i=1}^{N^b}; g^*) + \text{InfoNCE}(\{\{x_{i,c}^b, y_{x_{i,c}^b}\}_{c=1}^{16}\}_{i=1}^{N^b}; g^{'})
\end{aligned}
\qquad (11)
$$

where $x_{i,c}^b, x_{s,r}^b \in \{\{x_{i,c}^b, y_{x_{i,c}^b}\}_{c=1}^{16}\}_{i=1}^{N^b}$ and

$$
g^{'}(x_{i,c}^b, x_{s,r}^b) = e^{\frac{F(x_{i,c}^b)^T F^{1:t-1}(x_{s,r}^b)}{r}} \qquad (12)
$$

Experiment results

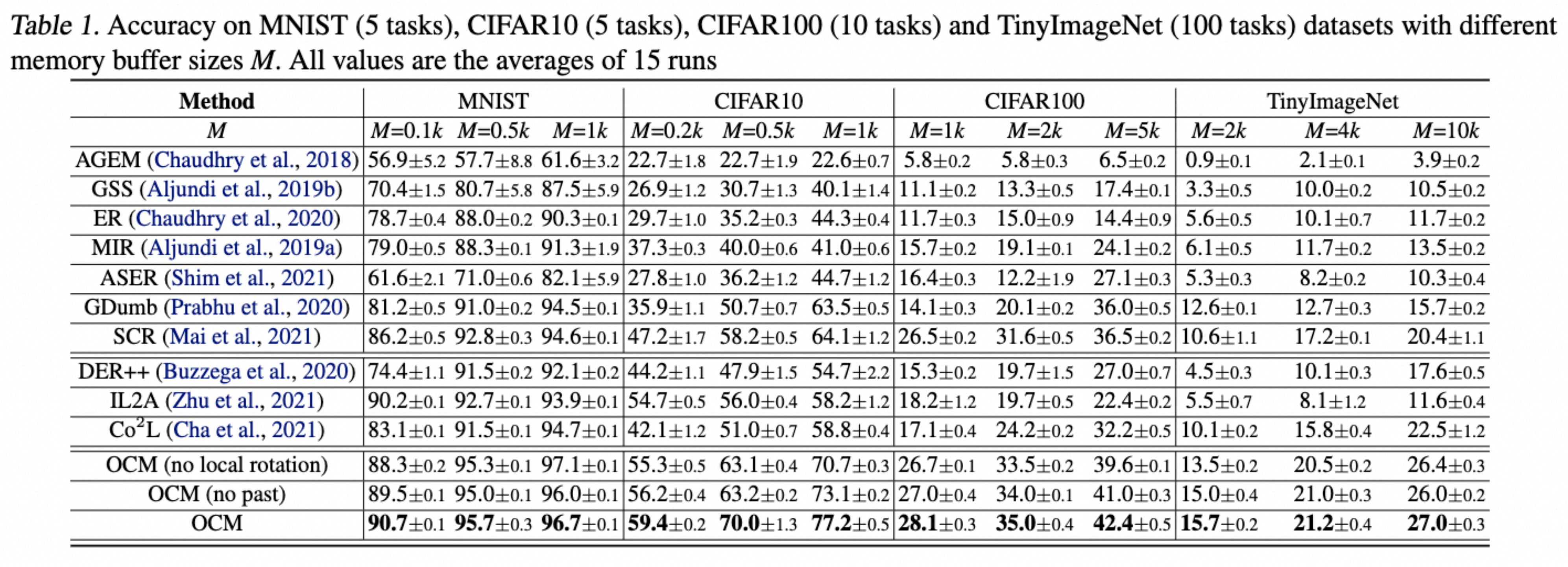


Table 1. Accuracy on MNIST (5 tasks), CIFAR10 (5 tasks), CIFAR100 (10 tasks) and TinyImageNet (100 tasks) datasets with different memory buffer sizes M. All values are the averages of 15 runs

Method	MNIST			CIFAR10			CIFAR100			TinyImageNet		
M	M=0.1k	M=0.5k	M=1k	M=0.2k	M=0.5k	M=1k	M=1k	M=2k	M=5k	M=2k	M=4k	M=10k
AGEM (Chaudhry et al., 2018)	56.9±5.2	57.7±8.8	61.6±3.2	22.7±1.8	22.7±1.9	22.6±0.7	5.8±0.2	5.8±0.3	6.5±0.2	0.9±0.1	2.1±0.1	3.9±0.2
GSS (Aljundi et al., 2019b)	70.4±1.5	80.7±5.8	87.5±5.9	26.9±1.2	30.7±1.3	40.1±1.4	11.1±0.2	13.3±0.5	17.4±0.1	3.3±0.5	10.0±0.2	10.5±0.2
ER (Chaudhry et al., 2020)	78.7±0.4	88.0±0.2	90.3±0.1	29.7±1.0	35.2±0.3	44.3±0.4	11.7±0.3	15.0±0.9	14.4±0.9	5.6±0.5	10.1±0.7	11.7±0.2
MIR (Aljundi et al., 2019a)	79.0±0.5	88.3±0.1	91.3±1.9	37.3±0.3	40.0±0.6	41.0±0.6	15.7±0.2	19.1±0.1	24.1±0.2	6.1±0.5	11.7±0.2	13.5±0.2
ASER (Shim et al., 2021)	61.6±2.1	71.0±0.6	82.1±5.9	27.8±1.0	36.2±1.2	44.7±1.2	16.4±0.3	12.2±1.9	27.1±0.3	5.3±0.3	8.2±0.2	10.3±0.4
GDumb (Prabhu et al., 2020)	81.2±0.5	91.0±0.2	94.5±0.1	35.9±1.1	50.7±0.7	63.5±0.5	14.1±0.3	20.1±0.2	36.0±0.5	12.6±0.1	12.7±0.3	15.7±0.2
SCR (Mai et al., 2021)	86.2±0.5	92.8±0.3	94.6±0.1	47.2±1.7	58.2±0.5	64.1±1.2	26.5±0.2	31.6±0.5	36.5±0.2	10.6±1.1	17.2±0.1	20.4±1.1
DER++ (Buzzega et al., 2020)	74.4±1.1	91.5±0.2	92.1±0.2	44.2±1.1	47.9±1.5	54.7±2.2	15.3±0.2	19.7±1.5	27.0±0.7	4.5±0.3	10.1±0.3	17.6±0.5
IL2A (Zhu et al., 2021)	90.2±0.1	92.7±0.1	93.9±0.1	54.7±0.5	56.0±0.4	58.2±1.2	18.2±1.2	19.7±0.5	22.4±0.2	5.5±0.7	8.1±1.2	11.6±0.4
Co^{2}L (Cha et al., 2021)	83.1±0.1	91.5±0.1	94.7±0.1	42.1±1.2	51.0±0.7	58.8±0.4	17.1±0.4	24.2±0.2	32.2±0.5	10.1±0.2	15.8±0.4	22.5±1.2
OCM (no local rotation)	88.3±0.2	95.3±0.1	97.1±0.1	55.3±0.5	63.1±0.4	70.7±0.3	26.7±0.1	33.5±0.2	39.6±0.1	13.5±0.2	20.5±0.2	26.4±0.3
OCM (no past)	89.5±0.1	95.0±0.1	96.0±0.1	56.2±0.4	63.2±0.2	73.1±0.2	27.0±0.4	34.0±0.1	41.0±0.3	15.0±0.4	21.0±0.3	26.0±0.2
OCM	**90.7**±0.1	**95.7**±0.3	**96.7**±0.1	**59.4**±0.2	**70.0**±1.3	**77.2**±0.5	**28.1**±0.3	**35.0**±0.4	**42.4**±0.5	**15.7**±0.2	**21.2**±0.4	**27.0**±0.3