Incremental Learning in Semantic Segmentation

Anam ur Rehman
Politecnico di Torino
Student id: s283909
s283909@studenti.polito.it

Umar Farooq
Politecnico di Torino
Student id: s292448
s292448@studenti.polito.it

***Abstract*—We study the behaviour of semantic segmentation models in incremental learning setup. We introduce a new incremental learning technique which merges rehearsal and data regularization techniques to overcome catastrophic forgetting and task-recency bias. We also experiment with different sizes of exemplar memory to study their influence on model's performance. This leads to suggestions for ways to address different incremental learning challenges for semantic segmentation.**

I. Introduction

Semantic segmentation is the task of clustering parts of an image together which belong to the same object class. It is a form of pixel-level prediction because each pixel in an image is classified according to a category. From portrait mode of a modern cell phone camera to Self-driving cars, it has a vast range of applications. Often it is necessary to update a trained model to embed new knowledge while maintaining the old one. One simple method is to jointly train the model on all training data whenever an upgrade is required (joint training). Limitations like memory restrictions, privacy or security concerns demand a better, more sustainable approach. In incremental class learning we study how to efficiently use the resources by eliminating the need to retrain from scratch at the arrival of new data.

In our work, we first perform a segmentation task on the Pascal VOC dataset using BiSeNet [5] , a lightweight model for real time semantic segmentation. Afterwards, we modify our model to adopt an incremental learning protocol MiB [13]. MiB is based on data regularization techniques and it explicitly handles the semantic shift of the special background class. To further our analysis, we allow a small portion of data, called exemplars [11], to be stored and used in future training sessions. We also perform extensive experiments to understand how a model behaves when we change the cardinality of exemplar set.

II. Semantic Segmentation

Pixel level predictions require rich feature extraction from input images as it has to predict *where* an object is located and *which* category it belongs to. Furthermore, applications like self-driving cars and security scanners require the segmentation task to be performed in real-time. In the following sections we discuss BiSeNet [5] which was designed specifically to efficiently extract spatial features from images and provide semantic segmentation in real-time.

A. *Related work*

State-of-the-art scene parsing frameworks are mostly based on the fully convolutional network (FCN). Long et al. [1] proposed one of the first deep learning works for semantic image segmentation, using a fully convolutional network (FCN). FCN includes only convolutional layers for both, feature extraction and classification tasks. Despite its popularity and effectiveness, the conventional FCN model has some limitations as it is not fast enough for real-time inference and it lacks spatial details in the output.

Ronneberger et al. [2] proposed U-NET for segmenting biological microscopic images. They modified and extended FCN architecture using encoder-decoder settings. In order to learn high quality spatial details from the images they fused the hierarchical features of the backbone network. Although it increases the spatial resolution and achieves good results when trained over very few medical images as input, it is slow due to the extra computational cost of embedding features between two paths.

Another technique for segmentation is to use multi-scale and pyramid network-based models. Zhao et al. [3] developed the Pyramid Scene Parsing Network (PSPN), a multi-scale network to better learn the global context representation of an image. It uses a pyramid pooling module to distinguish patterns of different receptive fields from input and the outputs of these pyramid layers are concatenated to capture a rich global information. Although concatenation is simple yet powerful, it brings redundant information into the network.

Deeplab, developed by Chen et al., [4] is a family of state-of-the-art image segmentation approaches. The model mainly utilize three key features. First is the use of dilated convolution to address the issue of reduced feature resolution caused by sequential use of max-pooling and striding. Second is Atrous Spatial Pyramid Pooling (ASPP), which captures objects as well as image context at multiple scales to robustly segment objects at multiple scales. Third is the use of fully connected Conditional Random Fields (CRF) as a post processing step to improve the predictions near the borders of an object. Deeplab achieved excellent performances on test datasets in reasonable inference time but still it is not suitable for real-time inference.

B. *BiSeNet*

Bilateral Segmentation Network (BiSeNet) [5] is a lightweight segmentation network which can perform inferences in Real Time. It was designed to overcome the lack of

Backbone	Augmentation	Val. MIoU
Resnet 50 ✓	RC=320	67.4
Resnet 18	RC=320	60.7
Resnet 101	RC=320	70.7
Resnet 50	RC=352	68.6
Resnet 50	RC= *400* ✓	69.5
Resnet 50	RC=320 & Rotation	69.4

TABLE I: Hyperparameters tuning of BiSeNet model for random crop size of the input image, random rotation and comparing different backbone architectures for context path. All results are evaluated as percentage mean Intersection-over-Union on validation dataset. All other parameters of the network are unchanged for these experiments. We mark the optimal parameters for future usage.

spatial details in feature representations and provide a sizable receptive field without increasing inference time. The authors argued that most of the segmentation approaches compromise the accuracy to speed, which is inferior in practice. The network mainly consists of two paths. First is a 3-layered spatial path to preserve the spatial details of the original input image and encode affluent spatial information. Second path is named as Context path and it is designed to provide a sufficient receptive field to the classifier. It is critical for each output pixel to have a large receptive field, such that no important information is left out when making the prediction. Techniques like PSPP [3] and ASPP [4] use pyramid pooling to extract the enlarged receptive field, but these methods are computationally expensive. The context path in BiSeNet uses a lightweight backbone model to down sample and extract high level features with large receptive fields. A global average pooling (GAP) is applied on the tail of these features which encodes the maximum receptive field with global context information. Finally, the features of the backbone model and up-sampled GAP results are combined to form the output of the context path. In the Context Path, a specific Attention refinement Module (ARM) is used to refine the features of each stage. It employs global average pooling to capture global context and computes an attention vector to guide the feature learning. It demands negligible computation cost, and it can refine the output feature of each stage in the Context Path.

The output features for spatial path and context path are of different nature as one encodes low level, rich detailed information and the other provides high level context information, respectively. BiSeNet includes a Feature fusion module (FFM) which uses batch normalization on concatenated features to scale them appropriately.

C. *Experiments*

We test BiSeNet on the PASCAL Visual Object Classes (VOC) 2012 dataset which contains 20 object categories and one special category, the background. More than 12,000 images with pixel-level annotations are used to train the model. We used the batches of size 32 for training. Mean-intersection-over-union in percentage is used as the performance metric.

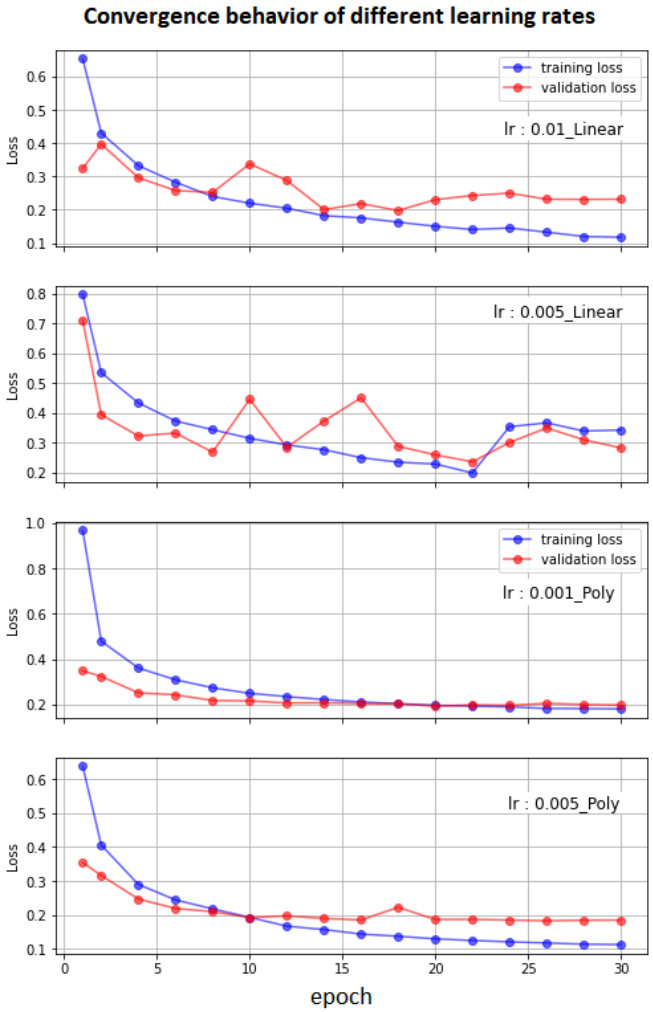

Fig. 1: learning curves of *BiSeNet* model for different learning rate settings. Linear stands for a constant learning rate for all epochs. Instead, poly indicates a dynamic learning rate which decreases after each epoch by a factor of $(1 - curr\ iter/tot\ iters)^p$ and p is to *0.9*.

In the context path, we replaced the originally proposed Xception39 network with ResNet50 [6] pre-trained on the Imagenet dataset. Model's performance on different learning rates is shown in Fig. 1. We select a learning rate of 0.005 with a polynomial scheduler of degree *0.9*. A learning rate decay helps the network learn complex patterns and prevents large gradient steps as the model gets closer to optimal solution. This learning rate provides faster and smoother convergence with higher MIoU percentage. It marginally over-fits on the training data but we can stop the training after 20 epochs to avoid overfitting as validation MIoU reaches a plateau after approximately 20 epochs. (Fig. 2)

Results of further hyperparameter tuning are shown in table I. ResNet101 has 44.5M parameters w.r.t 25.6M parameters of ResNet50 which brings additional computational cost in the network. The increase in performance with ResNet101 is almost negligible as compared to its computational overhead.

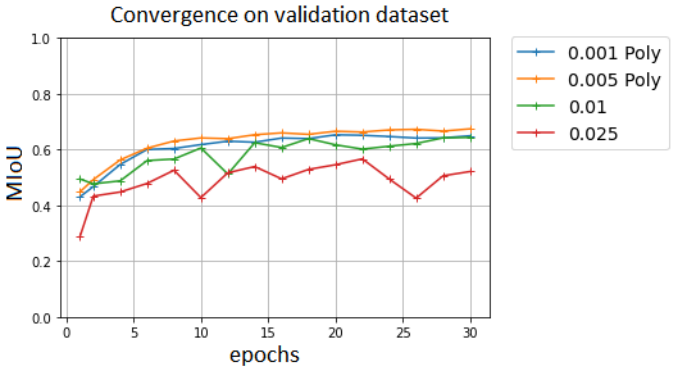

Fig. 2: Validation Learning Curves of BiSeNet for different learning rates.

Instead, ResNet50 yields much better performance as compared with ResNet18. (Table I)

Initially, we used random resized crop and random horizontal flip as data augmentation techniques. As we increase the random crop size of the input images, results slightly improve as more context information is available in larger crop sizes. A similar behavior is observed when we use random rotation of 5° in data augmentation. We used the marked hyperparameters table I for future usage. It provides comparable results without additional computational cost.

III. Incremental Learning

Incremental learning aims to develop artificially intelligent systems that can continuously learn new tasks from new data while preserving previously acquired knowledge. In most incremental learning (IL) scenarios, tasks are presented to a learner in a sequence of delineated training sessions. During these sessions, only data for new tasks is available for training. After each training session, the learner should be capable of performing all previously seen tasks on unseen data. This contrasts markedly with the prevailing supervised learning paradigm in which labeled data for all tasks is jointly available during a training session. Incremental learners only have access to data from a single task at a time while being evaluated on all tasks learned so far. The main challenge in incremental learning is to learn from new data additional tasks in a way that prevents forgetting of previously learned tasks. The naive approach of finetuning suffers from the lack of data from previous tasks and the resulting classifier is unable to retain previous knowledge. This drastic drop in performance on previously learned tasks is a phenomenon known as *catastrophic forgetting*.

A. Related work

Some incremental learning approaches use regularization terms together with the classification loss in order to mitigate catastrophic forgetting problems. EWC [7] regularizes the weights of neural network and estimates an importance metric for each parameter in the network. Instead, Learning Without Forgetting (LWF) [8] is a data regularization technique based on knowledge distillation [10]. It uses distillation loss to prevent activations drift. ILTSS [9] proposes a framework to learn semantic segmentation tasks, incrementally. Alongside the distillation loss on the output layer, they used a distillation loss on intermediate feature space before decoding stage to prevent drastic changes in feature representation.

	IL Tasks					
	15-5			15-5s		
Method	1 -15	16-20	all	1 -15	16-20	all
FT	4.6	29.6	10.5	1.1	4.4	2.6
LwF	47. 8	32.1	44.1	4.3	5.6	4.95
ILTSS	49.9	31.7	45.5	4.9	6.7	5.8
MiB	65.5	39.4	59.3	29.6	11.8	25.3
Joint	**75.3**	**68.9**	**73.8**	**75.3**	**68.9**	**73.8**

TABLE II: Performance overview of different incremental class learning methods on the Pascal-VOC 2012 dataset. Results are measured in MIoU percentage.

Other approaches include rehearsal methods which either store exemplars of each class from training data or generate them synthetically. These exemplars are later used to jointly train the model while learning future classes. ICARL [11] uses exemplars to build prototypes of old classes and classification is performed by nearest-mean-of-exemplars rule. Rehearsal based methods require a sampling strategy which must consider the approach by which exemplars are to be used in future. ICARL [11] uses herding approach which prefers data samples that are closer to mean feature representation of the class.

B. Modeling the Background for Incremental Learning

Modelling the background (MiB) [13] also uses distillation loss [10] as regularization technique on activations of the new network but its main contribution is to dynamically adopt the semantic shift of background class throughout the training phases. Authors argued that new training images may include pixels of classes from previous training sessions labeled as background. If the model learns to assign background class to these pixels it will cause dreadful results along with catastrophic forgetting. MiB employs an unbiased cross entropy loss which compares ground truth background labels with model predictions for that pixel being either background or one of the previously learned classes. Secondly, they observed that new classes may also appear in training data provided in previous training sessions. As distillation loss uses the previously learned model to make predictions on new training data, the Old model will probably assign the background class to all the pixels belonging to new classes. In MiB, a new unbiased distillation loss explicitly tackles this issue by summing probabilities of a pixel being one of the new classes when ground truth label is background. MiB also proposes a classifier initialization technique for new classes which makes sure that for a given pixel, the probability of being background is uniformly distributed among novel classes.

C. Experiments

We tested BiSeNet on Pascal VOC 2012 dataset to perform 2 different type of incremental tasks:

- 15-5. The model is jointly trained with standard cross entropy loss on the first 15 classes. In the next step, we train this model to additionally learn five new classes in a single step. The five new classes are *plant, sheep, sofa, train* and *tv-monitor*.
- 15-5s. This task is identical to 15-5 in all aspects but one. The five new classes are learned in 5 different training sessions, sequentially. The order of addition is same as mentioned above.

During training, at any given step, the model may receive images shared among different training sessions but all pixels belonging to either future or old classes are labeled as background. For incremental learning steps, unbiased cross-entropy and distillation losses are used.

Results obtained by joint training over the whole dataset serve as an upper bound. For baseline, we use fine tuning, LwF [8] and ILT [9]. LwF is a data regularization method which uses distillation loss alongside cross entropy loss to distill the knowledge from previously trained models. The ILTSS framework was specifically designed for incremental learning on semantic segmentation.

We trained the BiSeNet [5] model for 30 epochs using a learning rate of 0.005 with polynomial decay. For incremental steps we decreased the learning rate by a factor of 5.

D. *Results*

We report the results of different experiments in table II. Fine tuning (FT) serves as a baseline model and joint training can be seen as an upper bound for all incremental techniques as it trains the model statically on all classes in a single session. Performance is measured after the last training session in Mean Intersection-over-union over all the classes.

For 15-5, finetuning suffers a lot from catastrophic forgetting and we observed a major drop in performance as compared with upper bound. It is worth noticing that the performance is better on recently learned classes due to task-recency bias. Data regularization methods (LwF and ILTSS) yield better results than baseline, thanks to their knowledge distillation techniques. Both methods provide comparable results but due to additional regularization on encoder features, ILTSS performs slightly better on previously learned classes. MiB outperforms these methods with an overall 15% increase in MIoU with respect to ILTSS. Due to its ability to explicitly handle semantic drift of the special background class it is able to avoid conflicts and retain previously acquired knowledge.

Instead the 15-5s task is clearly more challenging as the model has additional training sessions. An increased number of gradient cycles allow models more time for drift in their activations and weights. FT almost completely loses its ability to classify pixels belonging to the first 15 classes. LwF [8] and ILTSS [9] also suffer a major drawback in performance and yield 40% lower performance w.r.t the 15-5 task. A similar behavior can also be observed for MiB which yields 35% lower score w.r.t 15-5 task but when compared with best baseline from the other methods, it outperforms all of them and provides 25% and 20% increase in overall and old classes, respectively. (table II)

IV. SOFT MIB

To study the role of exemplars in incremental learning we modify MiB settings and allow a fixed number of exemplars per class to be stored and used in future training sessions. As done in ILTSS [9], we assume that pixels belonging to old classes are not labeled as background but as their appropriate class in new training data. To merge rehearsal methods with data regularization and background modelling, we allow a fixed amount of exemplar memory to store the old training images for each class. These exemplars are used in future incremental learning tasks. The novel framework is similar to MiB but less restrictive, hence its name "Soft" MiB (SMiB). We further analyze how the cardinality of exemplars set influences the performance as we expand the exemplar memory. To avoid any additional overhead, we perform random sampling of exemplars for each class. An exemplar may include labels of more than one class from old training sessions, but it is counted as representative of only one of those classes.

SMiB ensures that new data will not label the old classes as background but as their original class. This small change in protocol has a major impact on the use of unbiased cross entropy loss [13]. Summing the probabilities for background class with one of the old classes is no more required and in practice it will provide poor results. We replace it with standard cross entropy loss. MiB's unbiased distillation loss is unchanged as during previous training sessions training images will label all future classes as background.

A. *Experiments*

Similar to section III-C, we performed our tests on the Pascal VOC 2012 dataset to incrementally learn 5 new classes. Initially we train a model jointly on data belonging to 15 classes. Then we perform two different incremental learning tasks 15-5 and 15-5s. While performing 15-5s, other models suffer extensively from catastrophic forgetting as shown in table II.

To analyze the impact of the number of exemplars per class on performance of SMiB, we test it with 5 different settings. In the first test we only allow new training data to include labels of old classes. It allows us to set a naive baseline for upcoming tests and eliminates the hypothesis that changes in performance are due to possible appearance of old classes in new training data. We test SMiB with exemplars of size 5,15,25 and 50 15-5 and 15-5s tasks of incremental learning. We also use the fine tuning method to build a model for 15-5 and 15-5s. The learning rate during incremental steps is decreased by a factor of 5 to further prevent the model from drifting away from the previously learned optimal solution.

B. *Results*

The results of SMiB with different sizes of exemplars are reported in table III . The naive baseline shows no major improvement over MiB. Which rejects the hypothesis that It

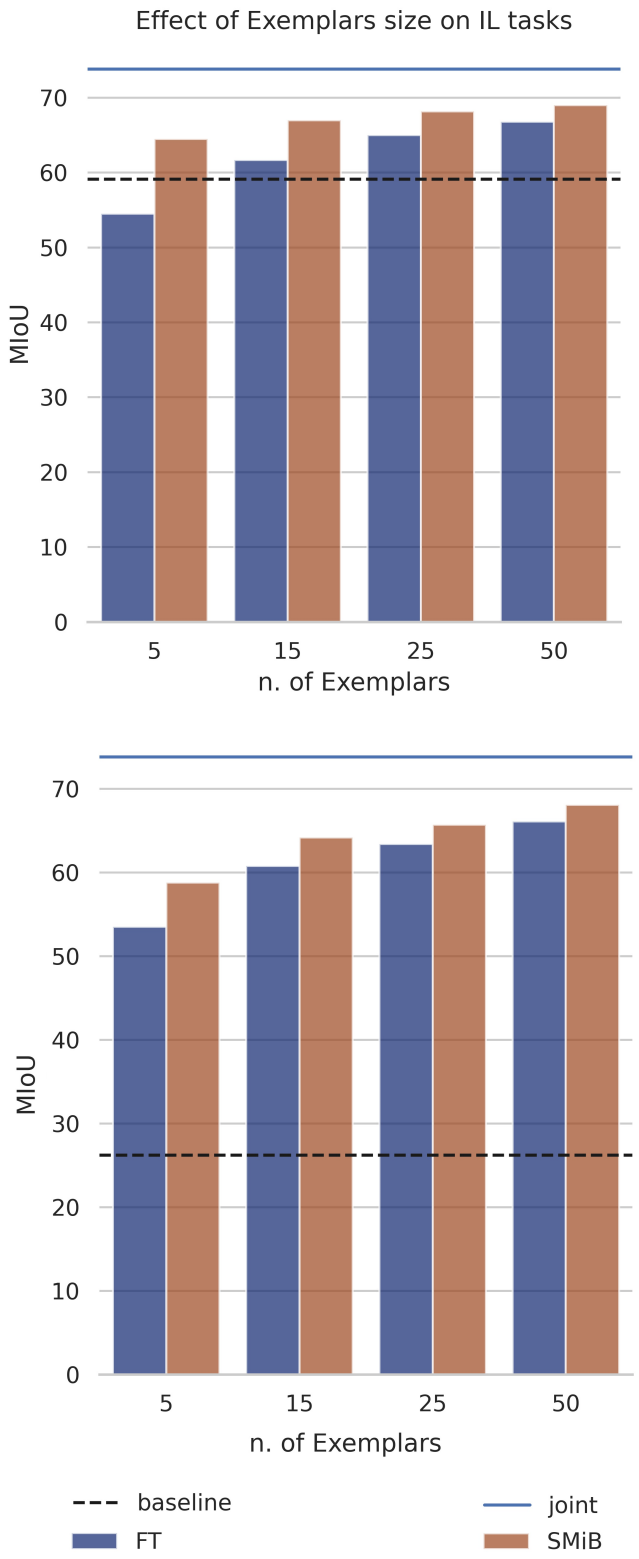

Fig. 3: Exemplars effect on models performance for 15-5 (top) and 15-5s (bottom) tasks. Baseline indicates the naive approach discussed in section IV-A.

is sufficient to assume that old classes will continue to appear in new training images and model will perform better. In incremental learning we make no such assumption on future tasks as an old class may never appear in training images of future sessions. On the other hand, when we allow our model to jointly train with exemplars stored from previous training sessions, an improved performance is observed for both incremental tasks. (Fig. 3)

The model trained with fine tuning gained a performance increase of 6% when we allowed 25 exemplars for the 15-5 task. Instead SMiB, due to it's explicit design to encode semantic shift of background class, exceeds the baseline performance by an overall 9% in MIoU for the same amount of exemplars. Task 15-5s was proved extremely challenging and all previously used methods suffer major drawbacks while performing it. Rehearsal methods on the other hand, yield prominent results on the 15-5s task. A model trained with fine tuning gains a performance increase of 27% by using just 5 exemplars per class. SMiB outperforms all non-rehearsal techniques for the 15-5s task and yields a 32% increase in MIoU w.r.t naive baseline.

As expected, in all cases, the performance increases as more exemplars are added. (Fig. 3). It's worth noticing that further Addition of exemplars becomes extrinsic after 25 exemplars per class in comparison to the gain in performance obtained. As an example, expanding the memory from 50 to 100 exemplars per class yields only 1.3% and 0.8% gain in MIoU on SMiB for 15-5 and 1-5s tasks, respectively. (table III)

C. Conclusions

We studied the incremental class learning problem for semantic segmentation, starting from MiB protocol. Almost all non rehearsal methods suffer from task-recency bias and catastrophic forgetting when they learn a small number of classes in multiple learning sessions. We address this issue by allowing a fixed exemplars memory for each class. Our proposed method (SMiB) provides prominent results even when a small number of exemplars are used. Such a small number of exemplars may introduce overfitting and class imbalance issues when used for image classification. However, the availability of pixel level annotations in semantic segmentation allows the model to retain information about old classes from a small number of exemplars. Exemplars help the model to avoid inter-task confusion and task-recency bias. As a future work, SMiB should be tested on other combinations of VOC dataset classes and other test data sets.

REFERENCES

[1] Jonathan Long, Evan Shelhamer, Trevor Darrell. **Fully Convolutional Networks for Semantic Segmentation.** Conference on Computer Vision and Pattern Recognition (2015).

[2] Olaf Ronneberger, Philipp Fischer, Thomas Brox. **U-Net: Convolutional Networks for Biomedical Image Segmentation.** Medical Image Computing and Computer Assisted Intervention (2015).

[3] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, Jiaya Jia. **Pyramid Scene Parsing Network.** Conference on Computer Vision and Pattern Recognition (2017).

[4] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, Alan L. Yuille. **DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs.** IEEE Transactions on Pattern Analysis and Machine Intelligence (2016).

[5] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, Nong Sang. **BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation**, 2018.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. **Deep Residual Learning for Image Recognition.** 2015.

[7] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka GrabskaBarwinska, et al. **Overcoming catastrophic forgetting in neural networks**. 2017

[8] Zhizhong Li and Derek Hoiem. **Learning without forgetting.** IEEE T-PAMI, 40(12):2935–2947, 2017

[9] Umberto Michieli and Pietro Zanuttigh. **Incremental learning techniques for semantic segmentation.** In ICCV-WS, pages 0–0, 2019.

[10] Geoffrey Hinton, Oriol Vinyals, Jeff Dean. **Distilling the Knowledge in a Neural Network.**

[11] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, Christoph H. Lampert. **iCaRL: Incremental Classifier and Representation Learning.**

[12] Marc Masana, Xialei Liu, Bartlomiej Twardowski, Mikel Menta, Andrew D. Bagdanov, Joost van de Weijer. **Class-incremental learning: survey and performance evaluation on image classification.**

[13] Fabio Cermelli, Massimiliano Mancini, Samuel Rota Bulò, Elisa Ricci, Barbara Caputo. **Modeling the Background for Incremental Learning in Semantic Segmentation,**

V. APPENDIX

A. Results of SMiB

IL Tasks							
		15-5			**15-5s**		
Method	**Exemplars**	1-15	16-20	all	1-15	16-20	all
MiB	-	65.5	39.4	59.3	29.6	11.8	25.3
MiB*	-	65.2	39.4	59.1	29.6	15.3	26.2
FT	5	59.5	38.3	54.5	59.4	34.5	53.5
SMiB	5	70	47.1	64.5	64.6	40.1	58.8
FT	15	67.2	44.2	61.7	65.9	44.5	60.8
SMiB	15	71.6	51.9	67	69.7	46.4	64.2
FT	25	69.7	50.1	65	68	48.7	63.4
SMiB	25	72.3	54.9	68.2	70.5	50.1	65.7
FT	50	71.2	52.7	66.8	70.2	52.7	66.1
SMiB	50	72.5	57.8	69	72.2	54.8	68.1
SMiB	100	73.2	61.2	70.3	72.1	58.5	68.9
Joint	-	**75.3**	**68.9**	**73.8**	**75.3**	**68.9**	**73.8**

TABLE III: Effect of exemplars memory size on performance of different models. MiB* refers to the naive approach discussed in section IV-A.

B. *Qualitative Results for IL*

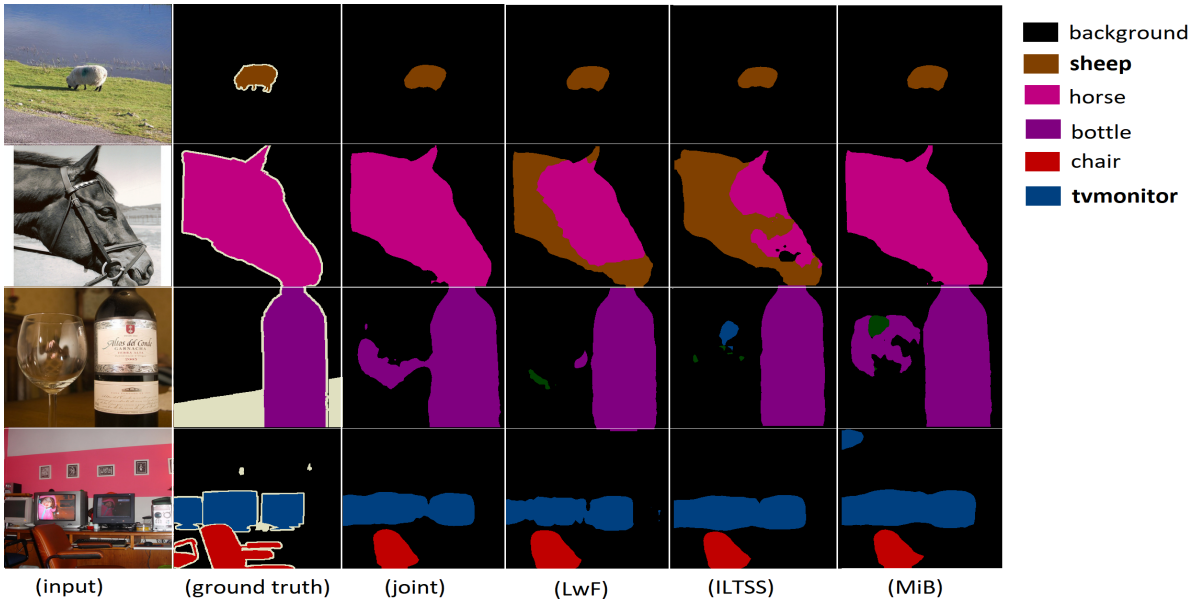

Fig. 4: Qualitative results of different semantic segmentation models performing **15-5** task. *Sheep* and *tvmonitor* are incrementally learned classes. In horse example, LwF and ILTSS are clearly effected by task-recency bias and inter-task confusion.

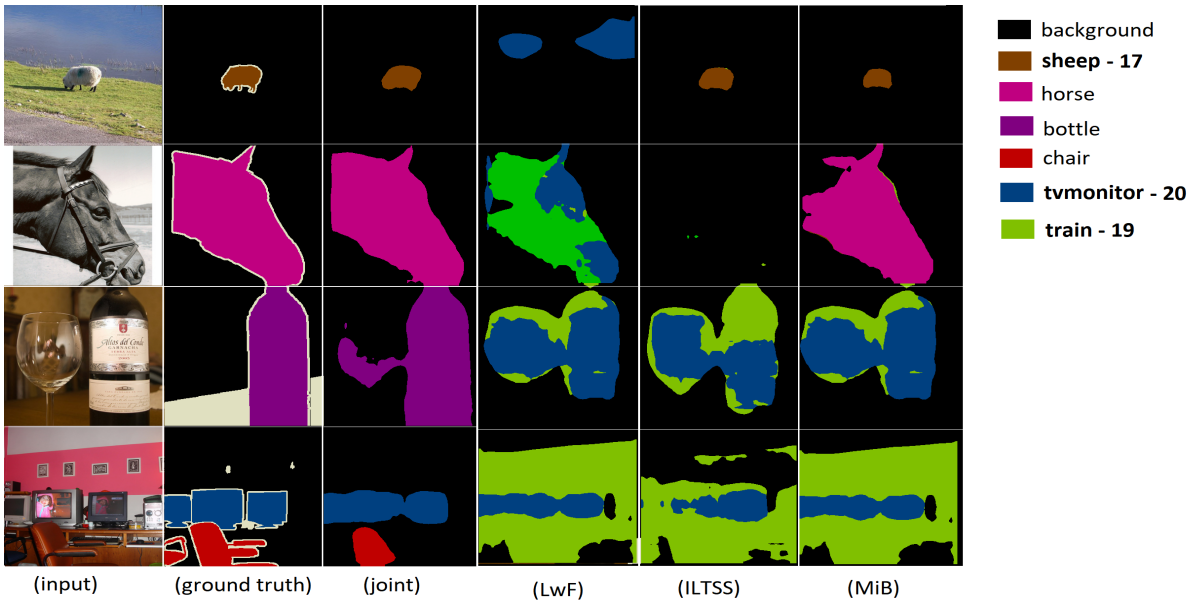

Fig. 5: Qualitative results of different semantic segmentation models performing **15-5s** task. *Sheep*, *train* and *tvmonitor* are incrementally learned classes in steps 2,4 and 5, respectively. Almost all models suffer from catastrophic forgetting and they have a major tendency to label pixels as newly learned classes.