

# Incremental Semantic Segmentation for Robotic Perception

Dewan Tauhid Rahman

University of Miami  
Department of Computer Science  
CSC752 – Fall 2025

**Abstract.** Robots deployed in open-world environments must continuously acquire new visual knowledge while retaining previously learned semantics. In practice, semantic segmentation models trained under standard supervised learning exhibit catastrophic forgetting when incrementally adapted to new classes. This paper presents a proof-of-concept study of rehearsal-free incremental semantic segmentation on the YCB-Video dataset [1] using SegFormer-B0 [2]. We evaluate four strategies: (1) naive incremental fine-tuning, (2) LoRA-based parameter-efficient adaptation, (3) modular adapter fusion with a frozen base model, and (4) LoRA combined with teacher–student knowledge distillation. Results show that naive and LoRA fine-tuning achieve strong new-class performance but collapse on base classes (base mIoU drops from 0.96 to  $\approx 0.18$ –0.19). Adapter fusion fully preserves base performance (0.96) but underfits new classes (0.42). The best stability–plasticity tradeoff is obtained with KD-LoRA, achieving base-class mIoU 0.8833 and new-class mIoU 0.9144. These experiments establish methodological grounding toward modular robotic perception systems that can expand class vocabularies online.

## 1 Introduction

Robots in household, industrial, and exploratory settings experience changing environments: new objects, tools, and scene configurations appear after deployment. Updating perception models by full retraining is often infeasible due to compute cost, limited access to historical data, and operational constraints (e.g., downtime, bandwidth, safety). These realities motivate *continual* or *incremental* semantic segmentation, where a model must learn new categories over time while preserving performance on previously learned classes.

A core barrier to continual learning is *catastrophic forgetting* [3, 4]: when a network is optimized on new data, parameter updates overwrite representations needed for earlier tasks. Given a model  $f_{\theta_t}$  after learning task  $t$ , learning task  $t+1$  updates parameters as

$$\theta_{t+1} = \theta_t - \eta \nabla_{\theta} \mathcal{L}_{t+1}, \quad (1)$$

where gradients are computed only using new-task data. In dense prediction tasks, forgetting can be amplified because gradients are aggregated across all

pixels, so feature drift affects large portions of the output map. In our baseline experiment (See Sec 4), naive fine-tuning reduces base-class mIoU from 0.96 to 0.18.

Our long-term vision is a modular robotic perception system that can expand its knowledge during deployment:

- detect novel objects and autonomously capture observations,
- generate semantic masks via an automatic annotation pipeline,
- train lightweight per-class adapters (e.g., LoRA) without retraining the full backbone,
- maintain a growing library of class-specific modules,
- select or fuse modules at inference time via a routing mechanism.

Implementing full online discovery and annotation is beyond a single-semester scope; therefore, this work focuses on foundational experiments that characterize incremental segmentation behavior and evaluate practical retention mechanisms.

## 2 Related Work

**Continual Learning and Catastrophic Forgetting:** Catastrophic forgetting was identified in early connectionist models [3,4]. Regularization-based methods such as Elastic Weight Consolidation (EWC) mitigate forgetting by constraining updates to parameters important for prior tasks [5]. Learning without Forgetting (LwF) introduced distillation constraints to preserve prior behavior without retaining old data [6]. While many early studies focused on classification, dense prediction tasks introduce additional challenges.

**Incremental Semantic Segmentation:** Incremental semantic segmentation is more challenging than classification because the model must preserve spatially grounded features and pixel-level decision boundaries. Prior work demonstrates that forgetting is often stronger in segmentation due to shared encoders and correlated outputs [7]. Rehearsal-based approaches can reduce forgetting but are memory-intensive and less suitable for long-term robotics scenarios.

**Parameter-Efficient Adaptation and Modularity:** Adapter methods add lightweight task-specific modules while freezing the backbone [8]. LoRA learns low-rank updates to weight matrices and has emerged as an effective parameter-efficient adaptation strategy [9]. Knowledge distillation [10] is commonly used to preserve model behavior and is central to rehearsal-free incremental learning strategies such as LwF [6]. This work evaluates these components in a unified incremental segmentation setting for robotic perception.

## 3 Methodology

**Dataset Preparation and Subset Selection:** We use YCB-Video [1], which provides RGB-D sequences with pixel-level masks for 21 household objects. The

Table 1: SegFormer model variants with parameter counts and FLOPs.

Model	Parameters (M)	FLOPs (G)
SegFormer-B0	7.72	3.30
SegFormer-B1	29.68	9.67
SegFormer-B2	56.41	24.83
SegFormer-B3	98.40	33.69
SegFormer-B4	129.88	40.34
SegFormer-B5	175.36	49.94

full dataset contains 92 sequences and approximately 133k frames. Since incremental learning experiments require multiple sequential training runs, and this project is a proof of concept, we use a representative subset: five sequences yielding 6,638 usable frames after filtering corrupted/incomplete samples. We split these into 5,974 training and 664 validation frames.

All RGB images and label masks are resized to  $384 \times 384$ . Pixel values are normalized using SegFormer preprocessing to maintain compatibility with pre-trained weights. To simulate incremental learning, classes are partitioned into base and new sets. During base training, pixels belonging to new classes are mapped to an ignore index; during incremental training, base-class pixels are ignored. This label-remapping strategy cleanly enforces phase-specific supervision without restructuring the dataset.

**Model Architecture:** We adopt SegFormer-B2 as our segmentation backbone due to its balance between expressiveness and efficiency. SegFormer employs a hierarchical transformer encoder and a lightweight MLP decoder, enabling effective global context modeling. The B2 variant contains 56.41M parameters and 24.83 GFLOPs (See Table 1), offering stronger representational capacity than smaller variants while remaining computationally practical.

The SegFormer architecture consists of a multi-stage transformer encoder that extracts hierarchical feature representations at progressively reduced spatial resolutions. These features are fused by a lightweight decoder, enabling efficient dense prediction while preserving spatial detail.

**Performance Metrics:** To evaluate the segmentation performance of the proposed SegFormer-B2 model, we use Intersection over Union (IoU) as the primary evaluation metric. IoU is a standard region-based measure in semantic segmentation that quantifies the spatial agreement between predicted masks and their corresponding ground-truth annotations. It is defined as the ratio of the overlapping area between the prediction  $Y$  and the ground truth  $T$  to the total area

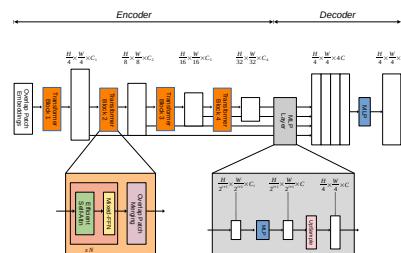


Fig. 1: SegFormer model architecture.

covered by both,

$$\text{IoU} = \frac{|Y \cap T|}{|Y \cup T|}.$$

By jointly accounting for false positives and false negatives, IoU penalizes misclassified regions and boundary inconsistencies, providing a stringent and informative assessment of segmentation quality.

**Experimental Setup:** All methods are optimized using AdamW with a learning rate of  $1 \times 10^{-4}$ . To ensure fair comparison across approaches, a consistent set of hyperparameters is used in all experiments. The batch size is fixed at 32, and each experiment is trained for 50 epochs. For parameter-efficient fine-tuning, the LoRA rank is set to  $r = 8$ , and when knowledge distillation is employed, a distillation temperature of 4 is used. All experiments are conducted on a single NVIDIA ADA 6000-series GPU, with batch size selected to fit within available GPU memory constraints.

## 4 Continual Learning Approaches

We consider a class-incremental semantic segmentation setting with two phases. In the *base phase*, the model is trained on a set of base classes  $C_{\text{base}}$  (plus background). In the *incremental phase*, only a disjoint set of new classes  $C_{\text{new}}$  is supervised. To enforce phase-specific supervision, labels are remapped such that pixels belonging to classes not present in the current phase are assigned an ignore index  $\kappa$  (i.e., excluded from the loss). Concretely, the supervised loss for a sample with image  $x$  and pixel labels  $y$  is computed as

$$\mathcal{L}_{\text{CE}}(x, y) = - \sum_{p \in \Omega : y_p \neq \kappa} \log \frac{\exp(z_{p,y_p})}{\sum_{c \in \mathcal{C}} \exp(z_{p,c})}, \quad (2)$$

where  $\Omega$  denotes the set of pixels,  $z_{p,c}$  are the model logits at pixel  $p$  for class  $c$ , and  $\mathcal{C}$  is the full label set (background + 21 object classes). Since SegFormer outputs logits at a lower spatial resolution, logits are bilinearly upsampled to the label resolution prior to computing the loss and metrics.

**Experiment 1: Naive Incremental Fine-Tuning:** As a baseline, we train SegFormer-B0 on base classes, then fine-tune *all* parameters on the incremental data stream containing only new-class supervision. Formally, if  $\theta$  denotes all model parameters, naive fine-tuning updates

$$\theta \leftarrow \theta - \eta \nabla_{\theta} \mathcal{L}_{\text{CE}}^{\text{new}}. \quad (3)$$

Because the optimization signal is computed exclusively from  $C_{\text{new}}$ , the shared encoder and decoder parameters drift toward representing new classes, often degrading feature subspaces that previously separated base-class regions. This yields the canonical catastrophic forgetting behavior in dense prediction:

- Base mIoU: 0.96 → 0.18
- New mIoU: 0.95

**Experiment 2: LoRA Incremental Fine-Tuning:** We next evaluate parameter-efficient adaptation using LoRA [9]. For a linear projection with weight matrix  $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$ , LoRA parameterizes an update as a low-rank decomposition:

$$W' = W + \Delta W, \quad \Delta W = \frac{\alpha}{r} BA, \quad (4)$$

where  $A \in \mathbb{R}^{r \times d_{\text{in}}}$  and  $B \in \mathbb{R}^{d_{\text{out}} \times r}$  are trainable,  $r \ll \min(d_{\text{in}}, d_{\text{out}})$  is the rank, and  $\alpha$  is a scaling factor. In our implementation, LoRA modules are injected into the transformer attention projections (query/key/value) and projection layers, while the remaining backbone is frozen. We also allow the final segmentation classifier head to update to accommodate new classes.

Although LoRA limits the number of trainable parameters and constrains updates to a low-dimensional subspace, forgetting remains severe:

- Base mIoU after incremental training: 0.19
- New mIoU: 0.93

This suggests that in semantic segmentation, even restricted updates to attention projections (combined with adapting the segmentation head) can significantly alter spatial features and decision boundaries for base classes.

**Experiment 3: Adapter Fusion:** To eliminate forgetting by construction, we decouple *retention* and *adaptation* through a modular fusion approach. We keep the base model  $\theta_{\text{base}}$  fixed after base training and train a separate LoRA-adapted model  $\theta_{\text{adapt}}$  on new classes only. At inference, we fuse logits channel-wise:

$$z_{\text{fused}}^{(c)}(x) = \begin{cases} z_{\text{base}}^{(c)}(x), & c \in C_{\text{base}} \cup \{0\}, \\ z_{\text{adapt}}^{(c)}(x), & c \in C_{\text{new}}. \end{cases} \quad (5)$$

This fusion is equivalent to a hard routing rule in which base classes are always predicted using the frozen base expert, and new classes are predicted using the adapter expert.

As expected, base performance is preserved exactly because  $\theta_{\text{base}}$  is never updated:

- Base mIoU: 0.96 (no forgetting)

However, new-class performance is limited:

- New mIoU: 0.42

A likely explanation is that the adapter expert is trained without the ability to reshape shared features learned during base training; it must learn new semantics largely through low-rank updates alone, which can underfit when new classes require substantial feature reorganization.

**Experiment 4: KD-LoRA (Knowledge Distillation + LoRA):** Finally, we combine LoRA adaptation with teacher–student knowledge distillation to directly constrain forgetting at the output level. The frozen base model acts as a teacher, producing logits  $z_t(x)$ , while the student is a LoRA-augmented model producing logits  $z_s(x)$ . The student is trained on new classes using cross-entropy, while simultaneously matching the teacher’s predictions on base classes via distillation.

Let  $S = C_{\text{base}} \cup \{0\}$  denote the base-class channel set including background. We compute distillation only over these channels (not the new-class channels), using temperature  $T$ :

$$\mathcal{L}_{\text{KD}} = T^2 \cdot \text{KL}\left(\text{softmax}\left(\frac{z_s^S}{T}\right) \parallel \text{softmax}\left(\frac{z_t^S}{T}\right)\right). \quad (6)$$

The full training objective is:

$$\mathcal{L} = \mathcal{L}_{\text{CE}}^{\text{new}} + \lambda \mathcal{L}_{\text{KD}}, \quad (7)$$

where  $\lambda$  controls the stability–plasticity tradeoff. Intuitively, the CE term promotes learning new semantic regions, while the KD term penalizes deviations from the teacher’s base-class decision function. This approach yields the best overall balance:

- Base mIoU: 0.8833
- New mIoU: 0.9144

Unlike adapter fusion, KD-LoRA permits adaptation to benefit from shared representations while explicitly discouraging destructive drift on the base-class outputs.

Table 2 provides a consolidated summary of base- and new-class performance across all incremental learning strategies evaluated in the preceding experiments.

Method	Base (Before)	Base (After)	New
Naive FT	0.96	0.18	0.95
LoRA FT	0.96	0.19	0.93
Adapter Fusion	0.96	0.96	0.42
KD-LoRA	0.96	0.8833	0.9144

Table 2: Mean Intersection-over-Union (mIoU) for base and new classes under different incremental learning strategies. **Base (Before)** and **Base (After)** report performance on base classes before and after incremental training, respectively, while **New** reports performance on newly introduced classes. Naive and LoRA fine-tuning suffer severe catastrophic forgetting, adapter fusion preserves base performance but underfits new classes, and KD-LoRA achieves the best stability–plasticity tradeoff.

## 5 Analysis and Discussion

Across all methods, the results reveal a clear stability–plasticity tradeoff in class-incremental semantic segmentation. Approaches that freely adapt shared parameters achieve strong performance on newly introduced classes but suffer from severe degradation on previously learned classes, while methods that explicitly preserve base representations limit their capacity to learn new semantics.

Naive fine-tuning exemplifies catastrophic forgetting in dense prediction: when optimization is driven solely by new-class supervision, shared encoder and decoder representations drift, erasing the spatial structures required for base classes. Although LoRA constrains updates to a low-rank subspace, it does not fundamentally resolve this issue. Even restricted modifications to attention projections, combined with adaptation of the segmentation head, are sufficient to significantly alter decision boundaries for base classes.

Adapter fusion eliminates forgetting by construction through complete isolation of base and new-class experts. However, this strict separation comes at the cost of adaptability: the new-class expert must operate largely on frozen features, which limits its ability to reorganize shared representations when new semantic concepts differ substantially from those seen during base training.

KD-LoRA offers a more balanced alternative by constraining behavior rather than parameters. Knowledge distillation anchors the student’s outputs on base classes to the frozen teacher, while LoRA provides controlled capacity for adaptation. This combination allows shared representations to evolve where necessary, without destructive drift on previously learned classes.

From a robotics perspective, these findings suggest that rehearsal-free continual learning should prioritize output-level stability over rigid parameter isolation. Methods such as KD-LoRA, which preserve safety-critical perception behavior while enabling incremental semantic expansion, are particularly well-suited for long-lived robotic systems operating in evolving environments.

## 6 Limitations and Future Work

This study is intended as a proof of concept for rehearsal-free incremental semantic segmentation in robotic perception, and several limitations arise from this scope. Experiments were conducted on a representative subset of the YCB-Video dataset rather than its full scale, and all evaluations were performed using the lightweight SegFormer-B0 backbone. While this choice enables efficient experimentation, it may limit absolute performance compared to larger architectures. In addition, training from scratch and extensive hyperparameter sweeps were not explored, as the focus of this work is on comparative behavior under incremental learning rather than peak accuracy.

Several components required for a fully autonomous continual learning system were also outside the scope of this study. In particular, online novelty detection, automatic annotation pipelines, and per-class adapter routing mechanisms were not implemented. Furthermore, all experiments were conducted offline, and

no evaluation was performed on physical robotic platforms or embedded hardware.

These limitations point directly to promising avenues for future work. Scaling the proposed methods to the full YCB-Video dataset and to larger segmentation backbones will help validate robustness at scale. Integrating online class discovery and automatic mask generation would enable true open-world learning during deployment. Training and managing per-class LoRA adapters, together with routing or gating networks, could facilitate modular and interpretable adaptation. Finally, deploying and benchmarking the system on real robotic platforms will be essential for assessing real-time performance, resource constraints, and long-term stability in practical settings.

## 7 Conclusion

This project studied rehearsal-free incremental semantic segmentation for robotic perception. Naive and LoRA fine-tuning show severe forgetting, adapter fusion preserves prior knowledge but underfits new classes, and KD-LoRA achieves the best stability–plasticity balance (0.8833 base mIoU, 0.9144 new mIoU). These findings provide a foundation for future modular robotic perception systems capable of continual class expansion.

## References

- Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, “Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes,” *Robotics: Science and Systems (RSS)*, 2018.
- E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, “Segformer: Simple and efficient design for semantic segmentation with transformers,” *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.
- M. McCloskey and N. J. Cohen, “Catastrophic interference in connectionist networks: The sequential learning problem,” *Psychology of Learning and Motivation*, vol. 24, pp. 109–165, 1989.
- R. M. French, “Catastrophic forgetting in connectionist networks,” *Trends in Cognitive Sciences*, vol. 3, no. 4, pp. 128–135, 1999.
- J. Kirkpatrick, R. Pascanu, N. Rabinowitz *et al.*, “Overcoming catastrophic forgetting in neural networks,” *Proceedings of the National Academy of Sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.
- Z. Li and D. Hoiem, “Learning without forgetting,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 12, pp. 2935–2947, 2017.
- F. Cermelli, M. Mancini, S. R. Bulo, E. Ricci, and B. Caputo, “Modeling the background for incremental learning in semantic segmentation,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 9233–9242.
- J. Pfeiffer, A. Rücklé, C. Poth *et al.*, “Adapterfusion: Non-destructive task composition for transfer learning,” in *European Conference on Computer Vision (ECCV)*. Springer, 2021, pp. 151–168.

9. E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, and W. Chen, “Lora: Low-rank adaptation of large language models,” *International Conference on Learning Representations (ICLR)*, 2022.
10. G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” *arXiv preprint arXiv:1503.02531*, 2015.