PAUSE: Post-Hoc Sequential Instance Unlearning in Panoptic Segmentation via Asymmetric Conflict Aware Gradient Projection

by

Moin Ahammed
24141107
Md. Azmain Adib
24141112
Md. Imdadul Islam
22301119

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
June 2026.

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Moin Ahammed

24141107

Md. Azmain Adib

24141112

Md. Imdadul Islam

22301119

Approval

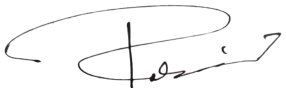
The thesis/project titled "PAUSE: Post-Hoc Sequential Instance Unlearning in Panoptic Segmentation via Asymmetric Conflict Aware Gradient Projection" submitted by

1. Moin Ahammed (24141107)
2. Md. Azmain Adib (24141112)
3. Md. Imdadul Islam (22301119)

of Spring, 2026 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science in June 13, 2026.

Examining Committee:

Supervisor:
(Member)



Md. Golam Rabiul Alam, PhD

Professor
Department of Computer Science and Engineering
Brac University

Thesis Coordinator:
(Member)

Md. Golam Rabiul Alam, PhD

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi

Chairperson
Department of Computer Science and Engineering
Brac University

Abstract

Panoptic segmentation models face increasing pressure to comply with privacy regulations and data-subject withdrawals, necessitating the post-hoc removal of specific training data. While machine unlearning bypasses the prohibitive cost of full retraining, it remains severely underexplored in dense prediction tasks. This challenge is particularly acute in transformer-based decoders, which couple individual object instances through shared query representations; consequently, suppressing a single target risks catastrophic degradation of its entire semantic class. This bottleneck intensifies under sequential deletion, where continuous updates often accumulate collateral damage or restore previously forgotten targets. To address this, we propose PAUSE (Post-Hoc Asymmetric Unlearning for Sequential Erasure), a post-hoc teacher-student framework tailored for query-structured architectures like Mask2Former. By integrating localized instance erasure with asymmetric gradient projection, damage-aware dynamic weighting, and a priority-ordered replay memory, PAUSE safely untangles the unlearning objective from shared features. Extensive evaluations on the Cityscapes dataset against NegGrad, NegGrad+, and SCRUB baselines demonstrate that PAUSE successfully processes sequential deletion queues, cleanly erasing targeted instances while bounding global utility degradation to a marginal ΔPQ of -0.4 percentage points. Finally, we formalize the fundamental tension between class-level retention and instance-level erasure, establishing a critical trade-off baseline for future dense-prediction unlearning systems.

Keywords: machine unlearning, panoptic segmentation, sequential erasure, transformer decoders, privacy compliance, Mask2Former, dense prediction

Acknowledgement

First and foremost, we want to sincerely praise and show gratitude to the Almighty Allah S.W.T. for providing us the strength, knowledge and patience to complete our thesis successfully. We would also like to express our gratitude to our supervisor, Dr. Md. Golam Rabiul Alam, sir, for his invaluable guidance and support throughout the course of this thesis. His insights and feedback were instrumental in shaping our work. Lastly, we are very much thankful to our parents, families, colleagues, and friends who have constantly supported us in fulfilling our goals. May Allah bless them all.

Table of Contents

List of Figures

List of Tables

Nomenclature

f_θ	Panoptic segmentation model with parameters θ.
θ_0, θ_t	Original (teacher) parameters; student parameters after request t.
$\mathcal{T}$, $\mathcal{S}$	Frozen teacher model and trainable student model.
(I_t, μ_t, c_t)	The t-th unlearning request: image, target mask, class label.
μ, $\bar{\mu}$	Target (forget) region and its complement (retain region).
$\mathcal{Q}_F$	Forget-query set: object queries responsible for the target.
$\mathcal{L}_{\text{erase}}$	Composite forget loss suppressing the target instance.
$\mathcal{L}_R$	Aggregate retention loss anchoring the student to the teacher.
$\mathcal{L}_{\text{mem}}$	Replay loss re-suppressing previously forgotten targets.
g_F, g_R	Forget and retain gradients.
$g_R^\perp$	Retain gradient projected orthogonally to g_F (forget-priority projection).
ρ	Forget-term share of total gradient magnitude (dynamic weighting).
PIV	Predicted-instance validity (per-query forgetting score).
PQ, SQ, RQ	Panoptic Quality with Segmentation and Recognition components.
ΔPQ	Change in global panoptic quality after unlearning.
Rem.	Panoptic-removal rate: targets absent from the rendered output.
$R_{\max}(\tau)$	Maximum late resurgence of a previously forgotten target τ.

Chapter 1

Introduction

As dense vision systems increasingly permeate privacy-sensitive domains, the ability to selectively forget learned information has transitioned from a theoretical concept to a critical operational imperative. Driven by data protection regulations such as the GDPR's right to erasure [14], the technical capacity to eradicate individual data influences from trained networks is now an indispensable prerequisite for responsible AI deployment [2], [18]. To address the formidable challenge of unlearning highly localized visual entities, we introduce **PAUSE** (**P**ost-Hoc **A**symmetric **U**nlearning for **S**equential **E**rasure). Our framework is a research pipeline engineered for post-hoc, sequential, instance-level unlearning within panoptic segmentation models. Constructed upon the Mask2Former architecture [29] with a Swin Transformer backbone [32], PAUSE selectively excises specific object instances without necessitating comprehensive retraining. We structure our pipeline within a teacher–student framework: a frozen teacher network provides reference predictions, whilst we incrementally update a trainable student network to suppress designated target instances. By examining this paradigm within a highly constrained environment—where target and retained regions share spatial dimensions and decoder representations—we establish a foundational approach to dense instance-level forgetting.

1.1 Background

To contextualise the technical contributions of our work, this section establishes the fundamental theoretical and architectural paradigms upon which our PAUSE framework is constructed. We first outline the task of panoptic segmentation and the structural mechanics of the Mask2Former architecture that serve as our baseline model. Subsequently, we review the core principles of machine unlearning, tracing the evolution from exact data deletion to modern approximate methods. Finally, we synthesise these domains by examining the acute complexities of instance-level unlearning within dense prediction environments, thereby framing the specific technical hurdles our methodology addresses.

1.1.1 Panoptic Segmentation

Panoptic segmentation [16] amalgamates semantic and instance segmentation to provide a cohesive representation of visual scenes. Each pixel is allocated a semantic

class and, for countable objects, a distinct instance identity. The task differentiates between *thing* classes (countable entities like vehicles and pedestrians) and *stuff* classes (amorphous regions like skies and roadways). By unifying these tasks, models must simultaneously reason about amorphous textures and distinct object boundaries. Performance is predominantly quantified via Panoptic Quality (PQ), which evaluates both detection precision (Recognition Quality) and mask overlap fidelity (Segmentation Quality). This RQ/SQ decomposition is particularly valuable for our unlearning evaluation, as it allows us to diagnose whether a model has forgotten an object's existence (low RQ) or merely degraded its spatial boundaries (low SQ). In our evaluation, PQ and its class-specific derivations serve as the primary barometers for retained model utility.

1.1.2 Mask2Former

Mask2Former [29] is a versatile architecture capable of executing panoptic, instance, and semantic segmentation within a unified framework. It deploys learnable object queries, decoded via transformer layers, to generate categorical and binary mask predictions. A pivotal feature is *masked attention*, which confines cross-attention to the predicted mask region from the preceding decoder layer. This localised feature extraction is highly advantageous for our unlearning framework, as it inherently restricts the spatial receptive field of each query. The architecture comprises a hierarchical backbone network [32] for multi-scale feature extraction, a pixel decoder to generate high-resolution embeddings, and a transformer decoder that processes the queries. Because query assignments are resolved via bipartite matching during training, we must employ a corresponding query-matching protocol during unlearning to identify the specific parameters responsible for a target instance. In our PAUSE strategy, we freeze the backbone and pixel decoder, isolating parameter updates exclusively to the transformer decoder.

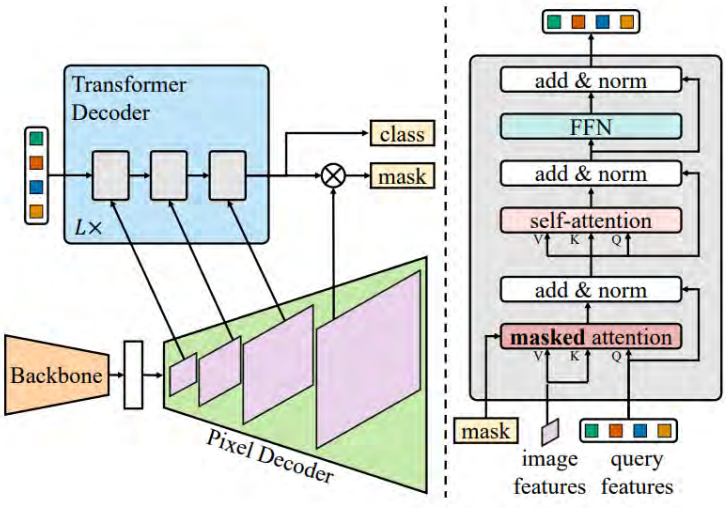


Figure 1.1: Mask2Former [29] architecture diagram

1.1.3 Machine Unlearning

Machine unlearning eradicates the influence of designated training data from a pre-trained model without exhaustive retraining [2]. Because exact unlearning methods (e.g., SISA [18]) scale poorly on massive models due to the requisite data partitioning and retraining, current research emphasises *approximate unlearning*. This approach adjusts a trained model using gradient manipulation to replicate the behaviour of a retrained baseline, balancing the fundamental tension between forgetting the target data and retaining core model utility. Prominent methodologies in this space include SCRUB [42], Bad Teacher [37], and SalUn [40]. The privacy vulnerabilities necessitating these frameworks are well-documented in dense prediction; membership inference attacks against segmentation models confirm that these architectures internalise specific characteristics of training samples [21].

1.1.4 Instance-Level Unlearning in Dense Prediction

Unlike image classification, where forgetting occurs at the broader class or image level, instance-level unlearning targets a *specific object instance* within a single image. In panoptic segmentation, this is exceptionally challenging. Target pixels are spatially contiguous with retain pixels, creating a severe risk of boundary leakage where the erasure signal damages surrounding non-target objects. Furthermore, the query-based mechanism means suppressing one query may disrupt adjacent or identically classified queries via shared self-attention pathways, leading to same-class collateral damage. Eradicating a singular vehicle risks degrading all vehicle representations unless the unlearning signal is meticulously isolated. Sequential unlearning introduces the additional threat of catastrophic accumulation, wherein successive iterations progressively compound these errors, compromising general utility or inadvertently reactivating previously suppressed instances. We address these obstacles through localised erasure protocols, asymmetric gradient projection, and dynamic loss weighting.

1.2 Rationale of the Study and Motivation

Panoptic segmentation models trained on sophisticated urban scene datasets, such as Cityscapes [5], inherently encode granular, instance-level representations of the objects they categorise. In real-world deployment scenarios—including autonomous vehicle navigation and smart city surveillance—these models process environments containing identifiable individuals and personal assets. The demonstrated capacity of deep segmentation architectures to memorise specific training permutations, as evidenced by successful membership inference attacks [21], renders the operationalisation of selective data removal an urgent technical necessity.

While efficacious in classification domains, existing approximate unlearning methodologies lack direct applicability to panoptic segmentation architectures. Traditional classification unlearning operates holistically across entire classes or discrete images; conversely, our requirement demands the precise suppression of a singular instance within an image whilst preserving all surrounding contextual data. Dense, pixel-level predictions generate complex spatial interdependencies absent in standard classification tasks. Furthermore, the query-based operational mechanics of Mask2Former

exacerbate these difficulties, as queries constantly interact and compete via self-attention mechanisms; the suppression of one query inherently risks redistributing its contextual evidence to adjacent queries. To the best of our knowledge, no prior literature has systematically investigated instance-level unlearning for panoptic segmentation within a sequential operational context. This represents a significant research gap and a pressing practical requirement that our work directly addresses.

1.3 Problem Statement

We formally define the instance-level panoptic unlearning paradigm as follows. Let $f_{\theta_0} : \mathcal{I} \to \mathcal{Y}$ be a pretrained panoptic segmentation model (the *teacher*) that maps an image $I \in \mathcal{I}$ to a set of Q query predictions:

$$f_{\theta_0} : I \;\longmapsto\; \big\{(p_q,\; m_q)\big\}_{q=1}^{Q}, \tag{1.1}$$

where $p_q \in \Delta^C$ denotes a distribution over C semantic classes (plus a "no object" class $\varnothing$), $m_q \in [0,1]^{H\times W}$ is a soft per-pixel mask, and a fixed, non-learnable fusion operator $\Pi(\cdot)$ collapses the query set into a panoptic map. A removal request is a *target instance* specified as a tuple:

$$\big(I_t,\; \mu_t,\; c_t\big), \tag{1.2}$$

where I_t is the image, $\mu_t \in \{0,1\}^{H\times W}$ is the binary mask of the target, and c_t is its class. Given a stream of T such requests, our goal in **sequential instance-level unlearning** is to produce a chain of *in-place* parameter updates $\theta_0 \to \theta_1 \to \cdots \to \theta_T$, applied by an operator $\mathcal{U}$,

$$\theta_t \;=\; \mathcal{U}\big(\theta_{t-1};\, I_t, \mu_t, c_t\big), \qquad \mathcal{F}_t \;=\; \mathcal{F}_{t-1} \cup \{(I_t, \mu_t, c_t)\}, \quad \mathcal{F}_0 = \varnothing, \tag{1.3}$$

where θ_t is never obtained by retraining from θ_0, and $\mathcal{R} = \mathcal{D}\backslash\bigcup_t\{(I_t, \mu_t, c_t)\}$ denotes the retain set. The chain of updates must satisfy the following conditions:

1. Forgetting Condition: For every removed instance, no segment of its class may survive in the rendered output; formally, for all $(I, \mu, c) \in \mathcal{F}_t$, there exists no segment $s \in \Pi(f_{\theta_t}(I))$ with $\text{class}(s) = c$ and $s \cap \mu \neq \varnothing$. The condition is enforced on the panoptic output, not on internal query scores.

2. Retention Condition: Predictions on the retain set $\mathcal{R}$ must remain behaviourally close to the teacher f_{θ_0}, i.e. $\mathbb{E}_{I\sim\mathcal{R}}\, d\big(f_{\theta_t}(I), f_{\theta_0}(I)\big) \leq \epsilon$ for a divergence d and tolerance $\epsilon \geq 0$. Global utility is quantified via the absolute change in Panoptic Quality (ΔPQ).

3. Sequential Non-Resurgence: The forgetting condition must hold over the *cumulative* set $\mathcal{F}_t$, so that for any previously processed target $j < t$, the instance erased at step j does not reactivate at step t.

1.4 Objectives

Our overarching objective is the conceptualisation and empirical evaluation of PAUSE. Our precise objectives are:

1. To investigate instance-level forgetting complexities in query-based models, addressing spatial codependencies and sequential metric drift.

2. To adapt Mask2Former into a teacher–student paradigm, facilitating selective, decoder-level updates.

3. To suppress target instances using localised erasure signals whilst safeguarding non-target data through same-class preservation algorithms.

4. To evaluate an asymmetric gradient projection and dynamic loss weighting system to stabilise the forget–retain equilibrium.

5. To engineer mitigation strategies against resurgence via replay memory and student-side query re-matching.

6. To assess unlearning efficacy, retention fidelity, and sequential stability via panoptic-output metrics and per-class analyses.

7. To transparently evaluate the framework's limitations and the fundamental forget–retain trade-off.

1.5 Methodology in Brief

To implement the PAUSE framework, we adopt the versatile Mask2Former architecture [29] equipped with a Swin-B backbone [32] to leverage its powerful multi-scale feature extraction capabilities. Operating within a structured teacher–student paradigm, our training configuration deliberately freezes both the backbone and the pixel decoder networks. This constraint serves to safeguard foundational low-level visual features against distortion, effectively isolating all subsequent parameter updates exclusively to the transformer decoder head where high-level semantic object queries are processed.

For each specified target instance designated for removal, a precise *query matching* procedure is deployed to identify the exact object queries responsible for the target's prediction. Once isolated, these matched queries are subjected to specialized *erase losses* designed to aggressively drive both the target's mask probabilities and its associated class presence scores toward zero. To counterbalance this destructive process and maintain model stability, we introduce localized *retain losses*. These losses are engineered to preserve predictions in the immediate spatial vicinity adjacent to the erased region, prevent collateral degradation of non-target instances belonging to the same semantic class, and maintain the model's global predictive validity across the remainder of the image.

Given that erasure and retention represent fundamentally competing objectives, optimizing them simultaneously can induce severe gradient conflict. We resolve these mathematical contradictions between the aggregated forget and retain gradients by integrating an asymmetric *gradient projection* algorithm [26] alongside a *dynamic loss weighting* framework [9] to dynamically harmonize the optimization trajectory. This update routine is executed sequentially across two operational phases: an aggressive initial erasure phase (Stage A) that prioritizes rapid target suppression, followed by a parameter-expanded repair phase (Stage B) dedicated to restoring baseline performance and smoothing decision boundaries. Finally, a dedicated *replay memory* matrix logs previously processed targets over time, acting as a structural anchor to prevent any downstream resurgence of forgotten concepts. The complete end-to-end pipeline is illustrated in Figure 1.2.

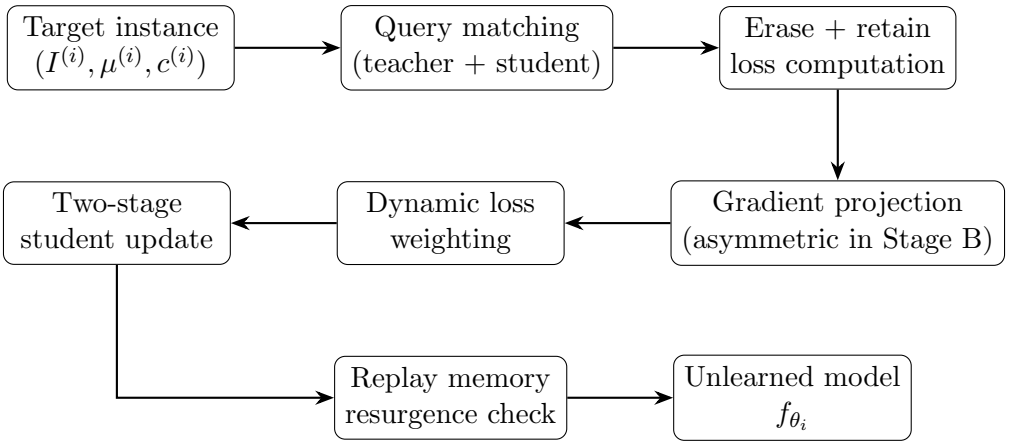


Figure 1.2: High-level overview of our PAUSE instance-level unlearning pipeline.

1.6 Scope and Challenges

Our architectural design and analytical interpretation are governed by several fundamental scope limitations and technical challenges:

1. **Framework Specificity:** Our empirical investigation is strictly confined to the Mask2Former architecture utilising a Swin-B backbone on the Cityscapes dataset. Generalisation to disparate architectures or datasets remains unvalidated. Furthermore, we restrict our unlearning targets exclusively to *thing-class* instances (e.g., vehicles, pedestrians).

2. **The Forget–Retain Trade-off:** Suppressing a target instance unequivocally requires altering decoder representations structurally shared with retained predictions. While our localised erasure protocols mitigate this, they cannot fundamentally eradicate all same-class collateral degradation.

3. **Sequential Accumulation:** Processing unlearning requests iteratively introduces the acute risk that successive updates will systematically compound utility erosion, or conversely, cause historically suppressed data to structurally resurface.

4. **Query Matching Drift:** Continuous student parameter drift creates operational uncertainty. As the student dynamically updates, its internal routing mechanics may reallocate an instance to an entirely distinct query than initially matched by the static teacher model.

5. **Conservative Metric Evaluation:** Determining if an instance is genuinely "forgotten" requires stringent metrics. While per-query suppression metrics provide insight, we prioritise panoptic-output removal metrics, as they critically reflect the model's actual functional output and serve as a more statistically rigorous assessment tool.

6. **Computational Constraints:** The raw computational overhead necessitated by continuous sequential unlearning on a standard single GPU framework (RTX 4090, 24 GB) strictly limits the step allowances and batch complexities we can permit per target instance.

1.7 Thesis Outline

The remainder of this thesis is organized into five chapters:
Chapter 2 presents a detailed literature review covering panoptic segmentation, machine unlearning, gradient-based optimization strategies, and the core challenges of instance-level forgetting in dense prediction systems.
Chapter 3 discusses the requirements, impacts, constraints, and practical considerations associated with the proposed framework, including ethical, societal, environmental, and computational aspects of sequential unlearning.
Chapter 4 introduces the proposed PAUSE framework and describes its complete methodology, including the teacher–student architecture, target-instance matching, localized erase and retain objectives, asymmetric gradient projection, replay-guided sequential repair, and evaluation protocol.
Chapter 5 presents the experimental setup and result analysis, including utility preservation, forgetting quality, sequential stability, qualitative behavior, ablation studies, statistical analysis, and comparisons with existing unlearning baselines.
Finally, **Chapter 6** summarizes the major findings of this work, discusses its contributions to dense-prediction machine unlearning, and outlines potential directions for future research.

Chapter 2

Literature Review

We explain the conceptual and technological literatures that the proposed framework PAUSE will rely on, in this chapter. We start with the well-defined operating regime of dense-prediction then we followed by the panoptic segmentation problem. After that we demonstrate machine unlearning and the specific issue of granularity, which introduced instance deletion as one of the best-studied options. We then analyze why the structure(s) of the panoptic models are hard to selectively edit, overview the algorithmic families that have been suggested for associated forgetting problems, and finally merge the evaluation rules adopted in the literature. Each section is designed to reveal by its completion, the part of design space that has yet to be developed and that is the focus of the remainder of this thesis.

2.1 Foundational Concepts in Instance-Level Unlearning

The slogan "making the model forget" generally refers to unlearning in binary classification.Unlearning in dense vision tasks is not as well known as "making the model forget." Unlike the classical, sample-based formulation, the requests that the deployed segmenter will be given for each image will be spatially structured. It analyze a single annotated object in a valid image, while the remaining annotations in that image should be correctly predicted and also the rest of the objects of the same class in the rest of the images throughout the dataset should be predicted with the same accuracy. This section outlines the vocabulary, assumptions, and notation needed to be carried on by the rest of the chapter, and which will be also formally inscribed in the PAUSE framework.

2.1.1 Panoptic Segmentation

A dense prediction task with each pixel of an image assigned a semantic class label and, if applicable, an instance identity, is said to be panoptic-segmentation [16]. Formally, given an input image $x \in \mathbf{R}^{H \times W \times 3}$, the desired output is a mapping

$$\Phi : \{1, \ldots, H\} \times \{1, \ldots, W\} \longrightarrow \mathcal{C} \times \mathbf{N}, \tag{2.1}$$

that sends each pixel s to a pair (c_s, k_s) consisting of a class label $c_s \in \mathcal{C}$ and an instance identifier k_s. The class set $\mathcal{C}$ is partitioned into two disjoint subsets,

$\mathcal{C} = \mathcal{C}_{\text{stuff}} \cup \mathcal{C}_{\text{things}}$, where *stuff* classes (e.g. sky, road, vegetation) describe uncountable regions and admit no instance identity, while *things* classes (e.g. person, car, bicycle) correspond to countable, separable objects to which a unique k_s must be assigned. By including these two seemingly separate tasks of semantic segmentation that outputs only a class map and instance segmentation that outputs only an instance partition for things classes, panoptic segmentation subsumes the two tasks [16].
There are three families of architectures that are important to this thesis:

- **Encoder–decoder semantic backbones.** Early panoptic systems were created by combining a semantic-segmentation branch (FCN [1], U-Net [4], or atrous-convolution networks of the DeepLab style [15]) and an instance-segmentation branch using Mask R-CNN [10]. Heuristics were added on post-hoc, fusing the two ends with end-to-end optimization of the panoptic objective being limited.

- **Query-based universal architectures.** Some recent work is trying to rephrase the panoptic segmentation task as a set-prediction problem following the paradigm of DETR [24]. MaskFormer [30]. MaskFormer [30] adopted the new idea of mask classification: Producing a class distribution and binary mask of every object query in a fixed-size set of N learned object queries $Q = \{q_1, \ldots, q_N\}$ from a transformer decoder. Then Mask2Former [29] generalized this further using masked attention and multi-scale feature aggregation, enabling it to perform well in all three tasks on this well-known dataset, panoptic, instance and semantic tasks in one architecture. OneFormer [38] bundled together the three tasks in task-conditioned joint training. This family includes a model unlearned in this thesis: a Mask2Former network, which was pre-trained on Cityscapes [5], with a Swin-B backbone.

- **Mask classification as the operating substrate.** Under the mask-classification view, the model outputs, for each query q_i, a class distribution $p_i \in \Delta^{|\mathcal{C}|}$ and a per-pixel mask logit $m_i \in \mathbf{R}^{H \times W}$. The panoptic prediction at pixel s is then obtained by a matching step

$$\Phi(s) = \big(\arg\max_{c \in \mathcal{C}} p_{i^\star(s)}(c),\ k_{i^\star(s)}\big), \qquad i^\star(s) = \arg\max_i \sigma(m_i)(s) \cdot p_i(\hat{c}_i), \quad (2.2)$$

  where σ is the sigmoid function and $\hat{c}_i$ the most likely class for query i. This indirect, query-mediated assignment is the structural feature on which the difficulty of instance-level unlearning in panoptic models ultimately rests.

The standard evaluation metric for panoptic segmentation is the Panoptic Quality (PQ), introduced together with the task itself [16]. Given, for each class c, the sets $\mathrm{TP}_c, \mathrm{FP}_c, \mathrm{FN}_c$ of matched, unmatched-predicted, and unmatched-ground-truth segments (under the rule that a pair is matched when their IoU strictly exceeds $1/2$), PQ admits the multiplicative decomposition

$$\mathrm{PQ}_c = \underbrace{\frac{\sum_{(p,g) \in \mathrm{TP}_c} \mathrm{IoU}(p,g)}{|\mathrm{TP}_c|}}_{\mathrm{SQ}_c} \cdot \underbrace{\frac{|\mathrm{TP}_c|}{|\mathrm{TP}_c| + \frac{1}{2}|\mathrm{FP}_c| + \frac{1}{2}|\mathrm{FN}_c|}}_{\mathrm{RQ}_c}, \quad (2.3)$$

into Segmentation Quality (SQ), defined as the average of the mask IoU evaluated in the pairs of mask images that are matched, and Recognition Quality (RQ), an F_1-like metric for the detection. In the panoptic literature, reporting is always published with SQ and RQ together and and any unlearning procedure that claims to preserve utility must do so under these aggregate measures restricted to the retain set.

2.1.2 Machine Unlearning

Machine unlearning is the problem of producing, from a model $f_{\theta_{\text{orig}}}$ already trained on a dataset D, and a designated forget subset $D_f \subset D$, an updated set of parameters θ_{unl} such that the resulting model behaves as if D_f had never been part of training [39]. Writing $D_r = D \setminus D_f$ for the retain set, the canonical statement of the problem is

$$\theta_{\text{unl}} \;\approx\; \theta_{\text{retrain}}, \qquad \theta_{\text{retrain}} = \arg\min_{\theta} \frac{1}{|D_r|} \sum_{(x,y)\in D_r} \ell\big(f_\theta(x), y\big), \tag{2.4}$$

where the notion of approximation in (2.4) can be made precise in different ways and gives rise to the two principal categories of unlearning methods. The motivation for the problem is partly regulatory and partly engineering. The European General Data Protection Regulation and analogous frameworks in other jurisdictions establish a right to erasure that, when the data has been used to train a model, is widely interpreted as requiring removal of the data's influence from the model and not merely from storage [14]. A complementary engineering motivation is the correction of training data discovered to be mislabeled, corrupted, or adversarially poisoned, where retraining is again the conceptual gold standard but is operationally prohibitive [39].
Following the taxonomy now standard in the literature, unlearning methods divide into two families [18], [39]:

- **Exact unlearning** guarantees that θ_{unl} is statistically indistinguishable from θ_{retrain}, typically through a structural argument over the training pipeline. The SISA framework is the archetype: by partitioning D into disjoint shards, training a constituent model on each, and aggregating predictions at inference, the deletion of any sample reduces to retraining a single shard [18]. Exact methods provide strong, verifiable guarantees but incur storage and accuracy costs that scale poorly with model size.

- **Approximate unlearning** relaxes the indistinguishability requirement and asks only that the influence of D_f on the parameters or outputs of the model be reduced below a tolerable bound [19], [20]. The bound may be defined parametrically (a distance in weight space), behaviorally (a difference in predictive distributions on a probe set), or adversarially (resistance to a membership inference attack [8]). Approximate methods sacrifice provable indistinguishability for tractability and are the dominant family for large, modern vision models. PAUSE is an approximate method in this sense.

Both families share the same two-sided objective structure: a forgetting term defined on D_f and a retention term defined on D_r. The difficulty of operationalising this structure for dense, query-based panoptic models is the central technical concern of this thesis.

2.1.3 Unlearning vs. Retraining

Retraining from scratch on D_r remains the conceptual yardstick against which every unlearning procedure is measured: by construction it is exact, and certifiability follows trivially from the training recipe. For a panoptic model in the Mask2Former family, however, this baseline is operationally untenable. A single full training run on COCO or Cityscapes consumes tens to hundreds of GPU-hours, and the deletion requests that a deployed system must handle are typically small, frequent, and streamed online rather than batched into a single event [39]. Re-executing the full training pipeline for every incoming request would dominate the operational cost of the system and is incompatible with the latency expectations attached to legal compliance.

Approximate unlearning trades this guarantee for tractability. Rather than recomputing $\arg\min_\theta \mathcal{L}_{D_r}(\theta)$ from a fresh initialisation, it perturbs the existing parameters so as to undo the marginal contribution of D_f, typically through a small number of carefully chosen gradient steps,

$$\theta_{t+1} \;=\; \theta_t \;-\; \eta\big(\nabla\mathcal{L}_r(\theta_t) - \lambda_f \nabla\mathcal{L}_f(\theta_t)\big), \tag{2.5}$$

or, in some cases, through a closed-form Hessian-based correction derived from influence functions [12], [20]. The cost of this trade is paid in certifiability: an approximate procedure cannot, in general, prove that no trace of the forgotten data remains, which has led the field to a parallel literature on auditing and verification [8], [35]. Dense prediction sharpens the trade-off, because the parameters that encode a single instance are entangled with those encoding many others, and an update aggressive enough to satisfy the first term in (2.5) is also aggressive enough to disturb the second.

2.1.4 Granularity of Unlearning Requests

A second axis along which the unlearning literature is organised is the granularity at which the deletion request is specified. Each granularity induces a different forget–retain partition and, accordingly, a different evaluation protocol [39]:

1. **Sample-level unlearning.** The forget set D_f consists of entire training examples; this is the classical setting for which SISA [18] and influence-function approaches [12] were developed.

2. **Class-level unlearning.** An entire semantic category is removed from the model's effective vocabulary, so that the model no longer recognises, say, all instances of the class person [34], [44]. In dense prediction this manifests as the suppression of supervision for a class across the entire dataset [25].

3. **Instance-level (region-level) unlearning.** A single annotated object within an image, identified by a binary mask $m \in \{0,1\}^{H\times W}$, is to be erased while the remainder of the image, including other instances of the same class, must be retained verbatim [43]. The forget and retain partitions in this case share images and differ only in the spatial support of supervision.

Instance-level unlearning is at once the finest and the least studied of these granularities, as illustrated in Figure 2.1. The reasons are structural rather than incidental.

Class-level methods can afford to be coarse because the boundary between forget and retain coincides with a semantic category that the model already represents compactly, and sample-level methods exploit the fact that a training example is a discrete, identifiable input. Instance-level requests in panoptic segmentation share neither property: the target lives inside a shared image alongside retain-set annotations, it shares its class with retain-set objects, and the parameters that produced it are the same parameters that produce its neighbours. The chapter is organised so as to make the cost of this coincidence explicit.

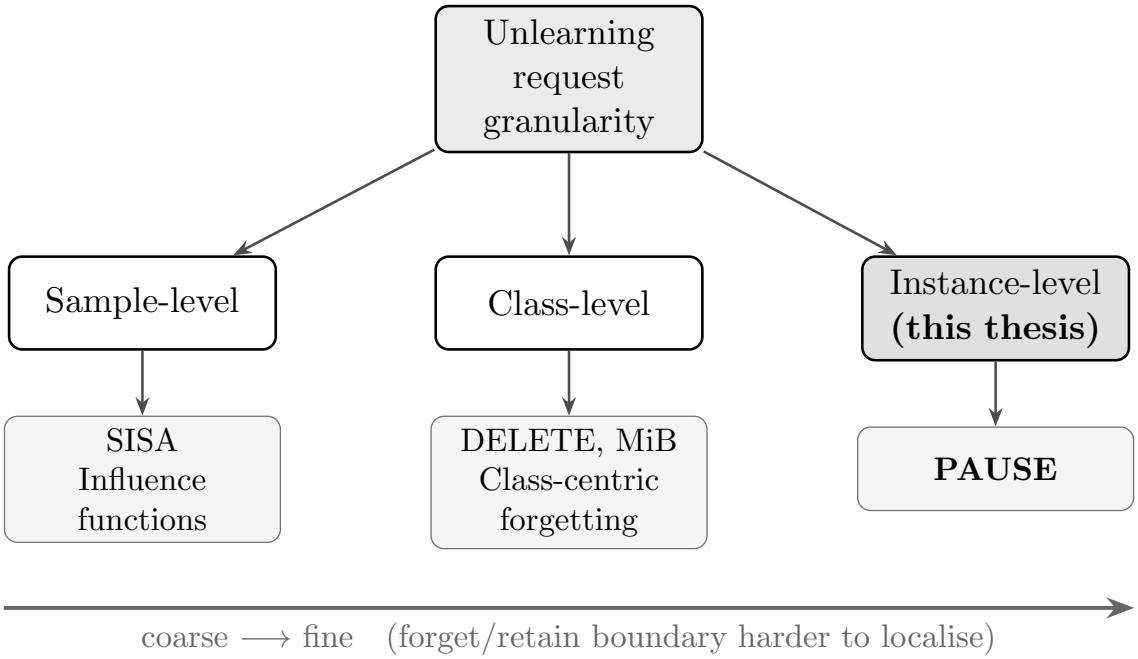


Figure 2.1: The granularity axis of unlearning requests, from coarse to fine. *Sample-level* deletion removes entire training examples; *class-level* deletion removes a whole semantic category from the model's vocabulary; *instance-level* deletion, the setting of this thesis, removes a single annotated object while every other object, including other objects of the same class in the same image, must be retained. Representative methods are placed at the granularity they natively operate on (SISA and influence functions at sample level; DELETE, MiB, and class-centric forgetting at class level; PAUSE at instance level). The forget/retain boundary grows progressively harder to localise as granularity refines, because at the instance level the forget and retain targets share images, classes, and parameters.

2.2 Core Challenges in Instance Unlearning for Panoptic Segmentation

Although interest in machine unlearning has grown rapidly, selectively removing information from an already-trained panoptic segmentation model remains technically demanding. Unlike coarse, image-level deletions, instance-level requests must remove a single object embedded in a valid image while leaving every other pixel-level prediction intact. This section identifies three structural difficulties that any practical instance-unlearning method must contend with, and that directly motivate

the region-aware, retention-anchored design adopted by PAUSE.

2.2.1 Where Panoptic Models Store Instance Information

The first difficulty is representational. In a query-based panoptic architecture such as Mask2Former [29], the shared backbone and pixel decoder produce a multi-scale feature pyramid that is consumed by every prediction in the image, and a transformer decoder then iteratively refines a fixed set of object queries. The mechanism by which a query specialises to a region is masked attention: at decoder layer ℓ, the cross-attention of query q_i is restricted to the pixels its own intermediate mask prediction $m_i^{(\ell-1)}$ already activates, so that

$$\mathrm{Attn}_i^{(\ell)}(s) \;=\; 0 \quad \text{wherever} \quad \sigma\big(m_i^{(\ell-1)}(s)\big) < \tfrac{1}{2}, \tag{2.6}$$

confining each query's update to its current footprint [29]. This is what makes the architecture efficient and accurate, but it is also why instance editing is non-local: a query's footprint is defined by the same shared pixel features that define every other query's footprint, so suppressing one footprint perturbs the feature substrate that the others read from. The mask logit for a single object makes this concrete, emerging as a dot product between a per-query embedding and the high-resolution pixel features,

$$m_i(s) \;=\; \langle q_i^{(L)},\, F(s)\rangle, \qquad i \in \{1,\dots,N\}, \tag{2.7}$$

where $q_i^{(L)}$ denotes the query embedding at the final decoder layer and $F(s) \in \mathbf{R}^d$ the pixel feature at location s. Two observations follow directly from (2.7). First, the pixel features F are shared across the entire image: the same parameters that delineate a particular instance also delineate every adjacent object and every same-class object elsewhere. Second, the query embeddings $q_i^{(L)}$ are themselves the output of a stack of attention and feed-forward layers whose parameters are tied across all queries, so that an intervention on the parameters that produced query $i^\star$ at inference time is, with high probability, also an intervention on the parameters that produce every other query.

The set-prediction training objective compounds this entanglement. Because queries are assigned to ground-truth segments through a bipartite matching rather than a fixed correspondence [24], [29], no query is the permanent owner of any instance; the query that claims a target at one optimisation step may relinquish it at the next. Target-level unlearning research in the dense-prediction setting has accordingly framed the problem explicitly as one of localisation: identifying which components of the model are responsible for a given target is harder than removing them once identified [43]. For panoptic instance unlearning the conclusion is sharper still: there is no single weight to delete, no contiguous subnetwork to prune, and no per-instance representation to suppress. Any operation that hopes to erase a single instance must instead reshape the joint behaviour of a representation that was never trained to keep instances separable in parameter space.

2.2.2 Forgetting–Retention Trade-off and Boundary Leakage

The representational entanglement just described produces the central optimisation tension of the problem, the forgetting–retention trade-off. Formally, the unlearning

objective can be written as a weighted combination of a forgetting and a retention term,

$$\mathcal{L}_{\text{unl}}(\theta) \;=\; \lambda_f \, \mathcal{L}_{\text{forget}}(\theta;\, \Omega_f) \;+\; \lambda_r \, \mathcal{L}_{\text{retain}}(\theta;\, \Omega_r), \tag{2.8}$$

where Ω_f is the spatial support of the target instance and $\Omega_r = \Omega \backslash \Omega_f$ its complement within the image [44]. The difficulty is that the parameters supporting $\mathcal{L}_{\text{forget}}$ and $\mathcal{L}_{\text{retain}}$ overlap substantially, so that the gradients of the two terms, $g_f = \nabla \mathcal{L}_{\text{forget}}$ and $g_r = \nabla \mathcal{L}_{\text{retain}}$, are typically far from orthogonal. The cosine alignment

$$\alpha(\theta) \;=\; \frac{\langle g_f(\theta),\, g_r(\theta) \rangle}{\|g_f(\theta)\|_2 \, \|g_r(\theta)\|_2} \tag{2.9}$$

is in practice non-trivially negative on the parameters that the two objectives contest, which is precisely the conflicting-gradient regime that motivates the gradient-surgery and null-space update schemes developed for multi-task and continual learning [17], [26], [31], [33]. The choice of how to resolve this conflict, treated symmetrically as in PCGrad or asymmetrically in favour of forgetting, is a central design decision that Section 2.3.1 develops and that PAUSE answers in a manner specific to the asymmetry of the unlearning objective.

A particularly visible manifestation of this trade-off in dense prediction is boundary leakage. The pixels immediately adjacent to a forgotten instance must continue to receive a coherent label, typically the surrounding stuff class or a contiguous neighbouring instance, and a poorly constrained unlearning step may instead carve a hole, hallucinate spurious instances around the deletion site, or shift the boundaries of overlapping objects [6], [44]. The continual-learning literature, which has long grappled with the stability–plasticity dilemma, provides several of the mechanisms now imported into the unlearning setting precisely to mitigate this leakage: knowledge distillation from a frozen teacher [3], [7], elastic weight consolidation [6], and the gradient-projection schemes already mentioned. The design space these mechanisms open up, in particular the combination of a region-restricted distillation anchor with a forget-priority projection of the gradient, is exactly the space that PAUSE inhabits.

2.2.3 Privacy Vulnerabilities and Relearning Risks

Even when the forgetting–retention trade-off appears to be managed at the level of standard accuracy metrics, the privacy guarantees of an approximate unlearning procedure can be substantially weaker than they first appear. Membership inference attacks, which attempt to determine from a model's outputs whether a particular example was part of its training set, remain the canonical diagnostic for residual information about forgotten data [8], [35]. In the dense-prediction setting, the natural per-instance analogue is to compare the distribution of confidences, losses, or gradient norms on previously forgotten instances against those on instances the model has genuinely never seen; an unlearning procedure that leaves the two distributions statistically distinguishable has not actually unlearned.

For the sequential setting that motivates this thesis, in which deletion requests arrive one after another over the lifetime of a deployed model, a second and underappreciated risk emerges: the *resurgence* of previously forgotten instances. Consider a sequence of forget requests $\{D_f^{(1)}, D_f^{(2)}, \ldots, D_f^{(T)}\}$ producing intermediate parameter states $\theta^{(0)} \to \theta^{(1)} \to \cdots \to \theta^{(T)}$. Resurgence at step t is the phenomenon in which

an earlier forget target $\tau \in D_f^{(j)}$, $j < t$, becomes once again predictable under $\theta^{(t)}$, measured as a rebound in its instance IoU,

$$\mathrm{Resurge}(\tau, t) \;=\; \mathrm{IoU}\big(\hat{m}_\tau^{(t)},\, m_\tau^\star\big) \;-\; \mathrm{IoU}\big(\hat{m}_\tau^{(j)},\, m_\tau^\star\big), \tag{2.10}$$

where $m_\tau^\star$ is the original target mask and $\hat{m}_\tau^{(\cdot)}$ the model's prediction at the corresponding step. Sequential unlearning has only recently been recognised as a setting distinct from one-shot unlearning, and the continual-learning literature that studies the analogous problem of catastrophic forgetting under streamed updates [6], [7], [36] offers few mechanisms that guarantee monotone non-increasing resurgence across requests. Bounded replay of previously forgotten material, deployed as an anchor against drift, is one of the few instruments to have begun to appear in this space, and underlies the memory component of PAUSE. A complementary failure mode peculiar to instance-level forgetting, in which a broad class-preservation objective inadvertently reintroduces an already-suppressed target, is examined in the discussion of results rather than here, as it is specific to the proposed method.

2.3 Algorithms and Methodologies for Instance Unlearning in Panoptic Segmentation

We now review the algorithmic families that have been proposed for related forgetting problems and assess their suitability for panoptic instance unlearning. Throughout this section a common notation is used.

Notation (panoptic mask classification). We let the training set be denoted $D = \{(x_i, Y_i)\}_{i=1}^N$, in which each annotation Y_i is a collection of instance masks and class labels $\{(m_{i,k}, c_{i,k})\}_k$. A query-based panoptic model f_θ produces N query outputs $(p_i, m_i)_{i=1}^N$, where $p_i \in \Delta^{|\mathcal{C}|}$ and $m_i \in \mathbf{R}^{H\times W}$. An instance-level unlearning request specifies a target binary mask $M \in \{0,1\}^{H\times W}$ within a particular image x, identifying the pixels of the instance to be forgotten. The complementary retain mask is $\bar{M} = 1 - M$. For any per-pixel loss $\ell(\cdot,\cdot)$, the masked empirical forget and retain risks are

$$\begin{aligned} \mathcal{L}_f(\theta) \;&=\; \frac{1}{|M|}\sum_s M(s)\,\ell\big(f_\theta(x)(s), y(s)\big), \\ \mathcal{L}_r(\theta) \;&=\; \frac{1}{|\bar{M}|}\sum_s \bar{M}(s)\,\ell\big(f_\theta(x)(s), y(s)\big). \end{aligned} \tag{2.11}$$

The post-unlearning parameters θ' are sought so that the model's behaviour on M is erased while its behaviour on $\bar{M}$, and on all other images in D, is retained.

2.3.1 Gradient-Driven Target Erasure and Gradient Conflict

A first family of approximate methods casts unlearning as a bi-objective optimisation over $(\mathcal{L}_f, \mathcal{L}_r)$, solved through gradient updates that amplify the forgetting signal while counteracting collateral drift on the retain set. The Lagrangian relaxation of

the constrained problem

$$\min_{\theta} \mathcal{L}_r(\theta) \quad \text{s.t.} \quad \mathcal{L}_f(\theta) \;\geq\; \tau \tag{2.12}$$

admits the unconstrained surrogate $J(\theta) = \mathcal{L}_r(\theta) - \lambda_f \mathcal{L}_f(\theta)$, whose gradient step is given by (2.5). The negative sign on $\mathcal{L}_f$ is what gives this family its colloquial name, NegGrad, whose pure form simply ascends the forget loss,

$$\mathcal{L}_{\text{NegGrad}}(\theta) \;=\; -\,\lambda_f\, \mathcal{L}_f(\theta), \tag{2.13}$$

with no retention term, projection, or memory. Its retention-augmented variant NegGrad+ adds a teacher-distilled retain term and is a standard unlearning baseline [42],

$$\mathcal{L}_{\text{NegGrad+}}(\theta) \;=\; \lambda_r \underbrace{\mathrm{KL}\big(f_{\theta_T}(x) \,\|\, f_\theta(x)\big)\big|_{\bar{M}}}_{\text{distil teacher on retain}} - \lambda_f\, \mathcal{L}_f(\theta), \tag{2.14}$$

where θ_T are the frozen teacher parameters. A distillation-based refinement, SCRUB, alternates a maximisation step that pushes the student away from the teacher on the forget set with a minimisation step that distils the teacher on the retain set [42],

$$\underbrace{\max_{\theta}\; \mathrm{KL}\big(f_{\theta_T}(x) \,\|\, f_\theta(x)\big)\big|_{M}}_{\text{MAX step (forget)}}, \qquad \underbrace{\min_{\theta}\; \mathrm{KL}\big(f_{\theta_T}(x) \,\|\, f_\theta(x)\big)\big|_{\bar{M}} + \mathcal{L}_{\text{task}}(\theta)\big|_{\bar{M}}}_{\text{MIN step (retain)}}, \tag{2.15}$$

the two steps interleaved every m iterations; SCRUB is among the most consistent approximate unlearners across forget-quality and utility metrics. The teacher–student view is shared by methods that distil from a deliberately incompetent (randomly initialised) teacher on the forget set to induce forgetting while a competent teacher anchors retention [37]. PAUSE adopts the same teacher–student instantiation, with the student initialised from the teacher and the teacher providing the held-out distillation signal that anchors the retain objective; the baselines of (2.13)–(2.15) are the comparators against which it is evaluated in Chapter 5.

The harder problem these methods expose is what to do when g_f and g_r point in opposing directions, the negative-alignment regime of (2.9). The multi-task and continual-learning literatures offer three responses. PCGrad resolves a conflict symmetrically by projecting each task's gradient onto the normal plane of the other when their cosine similarity is negative [26]; for two terms this replaces g_f by

$$g_f' \;=\; g_f - \frac{\langle g_f, g_r \rangle}{\|g_r\|_2^2}\, g_r \quad \text{if } \langle g_f, g_r \rangle < 0, \tag{2.16}$$

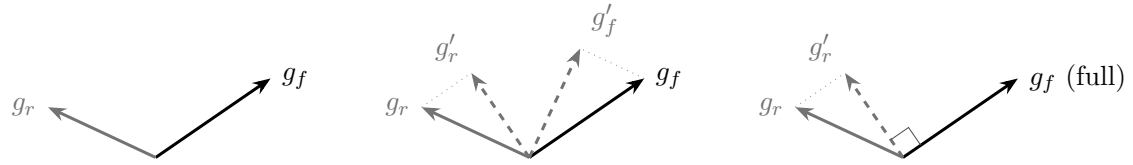
and symmetrically for g_r, so that neither objective is privileged. CAGrad instead seeks an update that minimises the average loss while guaranteeing a worst-case per-task decrease within a trust region around the average gradient [31]. Null-space methods take a stricter line: Adam-NSCL projects each candidate update into the approximate null space of the feature covariance of previous tasks, so that the update provably does not disturb what has already been learned [36], a strategy anticipated by gradient projection memory [33] and orthogonal weights modification [17].

For unlearning, the symmetry of PCGrad is a poor fit: the two objectives are not co-equal tasks but an erasure that must succeed and a retention that should be preserved only where it does not obstruct erasure. This observation motivates an

asymmetric resolution in which the forgetting gradient is delivered in full and the retain gradient is projected onto the orthogonal complement of g_f, contributing only its component that does not oppose forgetting:

$$g_r' \;=\; g_r - \frac{\langle g_r, g_f\rangle}{\|g_f\|_2^2}\, g_f \quad \text{if } \langle g_r, g_f\rangle < 0, \qquad g_f' = g_f. \tag{2.17}$$

This forget-priority projection is the gradient-surgery component of PAUSE, and Chapter 4 derives it in full; null-space projection of the forgetting gradient was also considered but, as reported in the results, was found unsuitable for a panoptic decoder whose modules are close to full rank, leaving the asymmetric scheme of (4.43) as the operative choice. Selective-update strategies that restrict the gradient to a single most-responsible layer, in the spirit of SLUG [43], address the cost concern but do not by themselves resolve the matching instability of (2.2) or the boundary leakage of Section 2.2.



(a) Conflict $\langle g_f, g_r\rangle < 0$

(b) PCGrad (symmetric) both gradients altered

(c) PAUSE (asymmetric) g_f in full; $g_r' \perp g_f$

Figure 2.2: Resolving the forget/retain gradient conflict. **(a)** Conflicting gradients: the forget gradient g_f and retain gradient g_r have negative cosine similarity, $\langle g_f, g_r\rangle < 0$. **(b)** Symmetric PCGrad (2.16) projects *each* gradient onto the normal plane of the other, treating the two objectives as co-equal tasks; both are altered. **(c)** The asymmetric forget-priority projection (4.43) used by PAUSE delivers g_f in full and projects only g_r onto the orthogonal complement of g_f (right angle marked), so that retention contributes solely the component that does not oppose forgetting. The asymmetry matches the structure of unlearning, where erasure must succeed and retention is preserved only where it does not obstruct erasure.

2.3.2 Dynamic Loss Weighting

Resolving the direction of the update leaves open the question of its relative magnitude: how large λ_f should be relative to λ_r in (2.8), and how this balance should evolve as forgetting proceeds. A fixed weighting is brittle, because the gradient magnitudes of the forget and retain terms vary by orders of magnitude over the course of an unlearning run and across targets. The multi-task literature has developed principled answers. Uncertainty weighting derives per-task weights from a homoscedastic noise model, scaling each loss by the inverse of a learned variance [11]. GradNorm instead operates directly on gradient magnitudes, adapting each weight w_k so that the per-term gradient norm is driven towards a common target [9]. Writing $g_k(\theta) = \left\|\nabla_\theta\, w_k \mathcal{L}_k(\theta)\right\|_2$ for the weighted gradient norm of term k and $\bar{g} = \frac{1}{K}\sum_k g_k$ for their mean, the target balances each term against the mean scaled

by its relative training rate r_k,

$$g_k(\theta) \;\longrightarrow\; \bar{g}(\theta)\,\big[r_k(\theta)\big]^{\alpha}, \qquad r_k(\theta) = \frac{\tilde{\mathcal{L}}_k(\theta)}{\frac{1}{K}\sum_j \tilde{\mathcal{L}}_j(\theta)}, \tag{2.18}$$

where $\tilde{\mathcal{L}}_k$ is the loss of term k relative to its initial value and α controls how strongly imbalance is penalised. PAUSE builds on this principle, equalising the per-term gradient magnitudes to a reference and then renormalising so that the forget term holds a fixed fraction of the total gradient magnitude, augmented with a damage-sensitive controller that throttles the forgetting weight when collateral on the target's class or boundary exceeds a tolerance and a retain floor that prevents the retention objective from being overwhelmed. The detailed schedule is deferred to Chapter 4; what matters for the present survey is that magnitude balancing, as much as direction, is an established and necessary instrument for managing competing objectives, and that it has not previously been specialised to the forget/retain asymmetry of unlearning.

2.3.3 Representation Pruning and Weight Saliency

A second family bypasses the loss landscape and intervenes directly on the model's internal representation. The shared intuition is that the information about a particular concept, class, or sample is carried by an identifiable subset of channels, attention heads, or parameter groups, and that surgically suppressing this subset will remove the concept while leaving the rest of the network intact [41]. For a parameter group θ_j, one defines mask-restricted forget and retain saliencies

$$S_j^f \;=\; \mathbf{E}_{(x,y,M)\sim D}\left[\sum_s M(s)\,\Big|\,\frac{\partial \ell\big(f_\theta(x)(s), y(s)\big)}{\partial \theta_j}\Big|\right], \quad S_j^r \;=\; \mathbf{E}\left[\sum_s \bar{M}(s)\,\Big|\,\frac{\partial \ell(\cdot)}{\partial \theta_j}\Big|\right], \tag{2.19}$$

and selects the prunable or update-eligible set

$$\mathcal{P} \;=\; \big\{\, j \;:\; S_j^f \;\geq\; \beta_f \;\wedge\; S_j^r \;\leq\; \beta_r \,\big\}. \tag{2.20}$$

Weight-saliency unlearning makes this concrete: SalUn identifies the parameters most salient to the forget objective by thresholding the magnitude of the forget gradient, and confines the unlearning update to that mask, narrowing the gap to exact retraining in classification and generation [40]. Sparsity-driven approaches show, more broadly, that pruning the model before or during unlearning makes approximate forgetting both cheaper and more thorough [41], and single-layer methods restrict the update to one critical layer chosen by importance and gradient-alignment diagnostics [43].

For class-level forgetting in classification networks, where a class can plausibly be identified with a small set of output-adjacent channels, these methods are competitive. For instance-level forgetting in panoptic segmentation the obstacle is (2.20) itself: a single instance does not, in general, correspond to a stable parameter subset. Two instances of the same class share the channels that encode that class, and even within a single image the object queries that delineate adjacent objects share their refinement pathway through the decoder. A pruning mask aggressive enough to drive S_j^f below the unlearning threshold will almost certainly drag S_j^r down with it, and the literature contains no demonstration of tight localisation at this granularity.

2.3.4 Decomposition and Exact-Style Methods via SISA

The SISA framework partitions the training data into S disjoint shards $\{D^{(s)}\}_{s=1}^{S}$ and, within each shard, into K sequential slices $\{D^{(s,k)}\}_{k=1}^{K}$ [18]. Training proceeds slice by slice,

$$\theta^{(s,k)} \;=\; \mathrm{Train}\big(\theta^{(s,k-1)},\, D^{(s,k)}\big), \qquad k = 1, \ldots, K, \tag{2.21}$$

and predictions are aggregated across shards. For panoptic outputs an aggregator at the segment-set level would replace the simple probability average; a candidate is to union the per-shard segment predictions and re-resolve the panoptic format by majority vote on overlapping supports. When a deletion request arrives, only the shards containing the affected samples need be retrained, and only from the earliest slice impacted by the deletion:

$$\theta^{(s,k_0)} \text{ retained}, \qquad \theta^{(s,k)} \;=\; \mathrm{Train}\big(\theta^{(s,k-1)},\, D^{(s,k)} \setminus D_f^{(s,k)}\big), \quad k = k_0 + 1, \ldots, K. \tag{2.22}$$

Importing this scheme into panoptic instance unlearning is technically possible but practically unattractive. The aggregation step in (2.21) assumes that each constituent model produces an independently interpretable prediction, an assumption that holds for classification logits but breaks for panoptic outputs, which are sets of segments with overlapping spatial supports that must be reconciled. The training cost of each shard remains that of a full panoptic model, so the cost saving is modest unless S is large, in which case per-shard data becomes too small to train a competitive model. Finally, instance-level requests cut across shards by their very nature: an image annotated with many instances may be drawn upon by deletion requests targeting any of them, so the granularity of partitioning fails to align with the granularity of deletion. Refining the partition to the patch or region level recovers some alignment but multiplies the bookkeeping cost and introduces spatial-context discontinuities at shard boundaries, a difficulty also observed in graph-structured extensions of the sharding idea [28].

2.3.5 Class-Centric Unlearning and Adaptive Deletion

A substantial portion of the segmentation-unlearning literature targets the removal of an entire class rather than a single instance. Class-centric methods typically combine a class-conditioned forget loss with a distillation term that anchors the model's behaviour on the remaining classes [25], [34]. A representative formulation, following the DELETE framework [44], decomposes the KL divergence between teacher and student distributions into a target-facing and a non-target-facing component. For a target class u and a pair of distributions $p, q \in \Delta^{|\mathcal{C}|}$, write

$$p^{(b)} = (p_u,\, p_{\setminus u}), \quad q^{(b)} = (q_u,\, q_{\setminus u}), \qquad \hat{p}_i = \frac{p_i}{p_{\setminus u}}, \quad \hat{q}_i = \frac{q_i}{q_{\setminus u}}, \quad i \neq u, \tag{2.23}$$

where $p_{\setminus u} = \sum_{i \neq u} p_i$. The KL divergence then decomposes as

$$\mathrm{KL}(p \,\|\, q) \;=\; \mathrm{KL}\big(p^{(b)} \,\|\, q^{(b)}\big) \;+\; p_{\setminus u}\, \mathrm{KL}\big(\hat{p} \,\|\, \hat{q}\big), \tag{2.24}$$

where the first term depends only on q_u, the probability the student places on the forget class, and so drives forgetting, while the second supervises relative probabilities among non-target classes and so enforces retention [44]. In dense prediction this

decomposition is applied pixelwise, optionally masked by $M(s)$ to focus forgetting pressure on the target region.
For instance-level unlearning the decomposition (2.24) does not transfer directly. Treating every pixel of the target class as part of the forget partition over-erases the model's behaviour on other instances of the same class, which is precisely what instance-level unlearning is required to preserve. Attempts to make class-centric losses more selective by spatially restricting the forget term to $M(s) = 1$ reintroduce the boundary-leakage problem of Section 2.2. Adaptive deletion approaches, which schedule the strength of the forgetting signal as a function of how much of the target has already been removed [34], mitigate over-erasure on average but provide no mechanism for accumulating deletions across a long sequence of requests. The scaling axis from class to instance and from one-shot to sequential is therefore visible in the literature but, at the time of writing, largely untravelled.
Table 2.1 consolidates the analysis of this section by scoring each algorithmic family against the requirements that sequential instance-level panoptic unlearning imposes. The judgements are *capability* assessments derived from the discussion in Sections 2.3.1–2.3.5, not head-to-head empirical results; the experimental comparison against the realised baselines is reported in Chapter 5. Read as a column, the table shows that no prior family satisfies all five requirements simultaneously, and that the instance granularity and panoptic-readiness columns are where existing methods are weakest.

Table 2.1: Suitability of unlearning method families for the requirements of sequential instance-level panoptic unlearning. Entries denote native capability: ✓ supported, ✗ not supported, ∼ partial or requires non-trivial adaptation. *Granularity* is the finest deletion unit the family natively addresses; *Sequential* indicates a mechanism against cross-request resurgence; *Utility* is the strength of retain-set preservation; *No retrain* indicates the method avoids full retraining; *Panoptic* indicates readiness for dense, query-based set-prediction outputs. Judgements follow the analysis of Sections 2.3.1–2.3.5.

Method family	**Granularity**	**Sequential**	**Utility**	**No retrain**	**Panoptic**
Retraining (gold standard)	any	—	exact	✗	∼
SISA / sharding [18]	sample	∼	exact	✗	✗
NegGrad / NegGrad+ [42]	sample/class	✗	weak	✓	∼
Distillation (SCRUB, Bad-T) [37], [42]	sample/class	✗	moderate	✓	∼
Saliency / pruning (SalUn, SLUG) [40], [43]	class/concept	✗	moderate	✓	✗
Class-centric (DELETE, MiB) [25], [44]	class	✗	strong	✓	∼
PAUSE (this work)	**instance**	✓	strong	✓	✓

2.4 Evaluation of Instance Unlearning in Panoptic Segmentation

Evaluating instance-level unlearning in panoptic models is, by its nature, multidimensional: the same parameter update must be assessed for the thoroughness with which it has erased the target instance, for its impact on every other prediction the model is asked to make, for its resistance to adversarial probes, and for its stability across a sequence of subsequent requests. This section formalises a protocol that combines these dimensions into a single reporting convention, drawn from the conventions of the panoptic, continual-learning, and unlearning-verification literatures,

and that the experimental chapters of this thesis adopt.

2.4.1 Forget / Retain-Set Construction

Let a panoptic dataset be $D = \{(x_i, Y_i)\}_{i=1}^{N}$, where $Y_i = \{(m_{i,k}, c_{i,k})\}_k$ enumerates the ground-truth instance masks and class labels in image x_i. An instance-level unlearning request is specified by a pair $(i^\star, k^\star)$, designating instance $k^\star$ in image $i^\star$, and induces a binary forget mask $M_{i^\star} = m_{i^\star,k^\star}$ with complementary retain mask $\bar{M}_{i^\star} = 1 - M_{i^\star}$. Crucially, the forget and retain partitions share images and differ only in the spatial support of supervision:

$$\begin{aligned} D_f &= \big\{(x_{i^\star},\, Y_{i^\star},\, M_{i^\star})\big\}, \\ D_r &= \big(D \setminus \{x_{i^\star}\}\big) \,\cup\, \big\{(x_{i^\star},\, Y_{i^\star} \setminus \{(m_{i^\star,k^\star}, c_{i^\star,k^\star})\},\, \bar{M}_{i^\star})\big\}. \end{aligned} \tag{2.25}$$

This split is region-aware rather than image-aware, in deliberate contrast to sample-level formulations [18], [39], and it is what enables the request to be honoured without discarding the supervisory signal carried by the surrounding annotations. In sequential settings, the protocol additionally records the order in which requests arrive, $(i_t^\star, k_t^\star)_{t=1}^{T}$, and retains, for each t, a held-out forget-evaluation probe consisting of the previously forgotten targets.

2.4.2 Forgetting-Quality Metrics

Three quantities, jointly, capture how thoroughly an instance has been forgotten. The first is the residual mask agreement between the post-unlearning prediction and the original target mask. Let $\hat{m}_{i^\star}^{\theta'}$ denote the model's predicted mask at $i^\star$ for the query that, prior to unlearning, claimed the target, and define

$$\mathrm{IoU}_f(\theta') \;=\; \frac{|\hat{m}_{i^\star}^{\theta'} \cap M_{i^\star}|}{|\hat{m}_{i^\star}^{\theta'} \cup M_{i^\star}|}. \tag{2.26}$$

A successful unlearning step drives $\mathrm{IoU}_f \to 0$. The second is the residual class-confidence mass within the target region, measuring whether the model still assigns appreciable probability to the original class on pixels formerly belonging to the forgotten instance:

$$\mathrm{Conf}_f(c^\star;\, \theta') \;=\; \frac{1}{|M_{i^\star}|} \sum_{s} M_{i^\star}(s)\, p_{\theta'}(c^\star \mid x_{i^\star},\, s). \tag{2.27}$$

The third is the confidence of any instance the model proposes within the same spatial neighbourhood, which captures the case in which the original instance has been suppressed but a near-identical replacement is being produced from another query.

A measurement subtlety, central to this thesis, distinguishes a *per-query* verdict from a *whole-output* verdict. Equations (2.26)–(2.27) interrogate the single query that originally claimed the target; but whether the instance actually disappears from the model's rendered panoptic output depends on the full assignment of (2.2), under which a different query may step in to re-explain the region. The two verdicts can disagree, and the honest headline measure of forgetting is therefore the one computed after the same panoptic post-processing that the deployed model uses,

namely whether any segment of the target's class survives over the target region. The thesis reports this panoptic-output removal rate as the primary forgetting metric and the per-query measures of (2.26)–(2.27) as secondary, explaining any divergence between them rather than reporting whichever is more favourable.
For sequential settings, these per-request measurements are supplemented by a longitudinal resurgence statistic. Following (4.35), for each previously forgotten target τ we recompute $\mathrm{IoU}_f(\theta^{(t)})$ at every subsequent step $t > j$ and report the maximum rebound,

$$\mathrm{R}_{\max}(\tau) \;=\; \max_{t>j} \Big(\mathrm{IoU}_f(\theta^{(t)};\,\tau) - \mathrm{IoU}_f(\theta^{(j)};\,\tau)\Big), \tag{2.28}$$

together with the fraction of past targets for which $\mathrm{R}_{\max}$ exceeds a configurable tolerance. Consensus on a single aggregate forgetting score comparable to PQ for utility has not yet emerged in the literature [35], [42], and the protocol adopted here reports the components jointly.

2.4.3 Utility-Preservation Metrics

Utility preservation is reported in terms of the panoptic metrics introduced in Section 2.1.1, restricted to the retain partition. Writing $\mathrm{TP}_c^r, \mathrm{FP}_c^r, \mathrm{FN}_c^r$ for the matched, unmatched-predicted, and unmatched-ground-truth segments of class c when matching is performed on $\bar{M}$, the retain-region panoptic quality is

$$\mathrm{PQ}_c^r(\theta') \;=\; \mathrm{SQ}_c^r(\theta') \cdot \mathrm{RQ}_c^r(\theta'), \tag{2.29}$$

with SQ^r and RQ^r defined exactly as in (2.3) but evaluated on $\bar{M}$. Aggregate reporting is class-stratified, since an unlearning step that surgically damages one or two classes can leave overall PQ almost unchanged while substantially harming the prediction quality for the class to which the target instance belonged [25]. Reporting only a global PQ that has been diluted across many untouched classes would therefore overstate utility preservation, and the per-class breakdown is treated as mandatory rather than supplementary. A retain-region output-drift score quantifies subtle distributional shifts not captured by PQ:

$$\mathrm{Drift}_r(\theta, \theta') \;=\; \frac{1}{\sum_i \sum_s \bar{M}_i(s)} \sum_i \sum_s \bar{M}_i(s)\, \mathrm{KL}\big(p_\theta(\cdot \mid x_i, s) \,\big\|\, p_{\theta'}(\cdot \mid x_i, s)\big), \tag{2.30}$$

where a low value indicates a localised update.
A single scalar that summarises the utility–forgetting balance, useful when comparing configurations within the same model family, is the unlearning-to-retention ratio

$$U2R(\theta, \theta') \;=\; \frac{\mathrm{PQ}^r(\theta')/\mathrm{PQ}^r(\theta)}{\mathrm{FQ}(\theta')/\mathrm{FQ}(\theta)}, \qquad \mathrm{FQ}(\theta') = 1 - \mathrm{IoU}_f(\theta'), \tag{2.31}$$

larger values of which indicate that a given forgetting improvement is achieved with proportionally smaller utility degradation [39].

2.4.4 Privacy Verification and Benchmarking Gaps

Beyond geometric and probabilistic measures of forgetting, the unlearning literature increasingly relies on adversarial probes to verify that no exploitable residue

remains in the model [8], [35]. The natural panoptic analogue of a membership inference attack compares, between forgotten instances and instances the model has genuinely never seen, the distribution of a chosen score: per-instance loss, query-level confidence, or the norm of the gradient of the loss with respect to the query embedding. A statistically indistinguishable pair of distributions, measured for example by a two-sample test or by the area under the attacker's ROC curve near 0.5, is a stronger forgetting certificate than (2.26) alone. A complementary metric, drawn from the continual-learning literature, is the relearning time

$$t_{\text{relearn}}(\tau) \;=\; \min\big\{t \geq 0 \;:\; \text{IoU}_f(\theta^{(t)};\, \tau) \;\geq\; \delta\big\}, \tag{2.32}$$

the number of standard fine-tuning steps required to push the predicted mask back to a target IoU δ after unlearning. Large t_{relearn} indicates that the erased instance does not re-emerge under benign continued training, which is desirable when subsequent updates are expected.

The picture that emerges from the evaluation literature, taken as a whole, is one of methodological fragmentation. There is no standard benchmark for instance-level unlearning in panoptic segmentation: no agreed dataset split, no canonical forget-request schedule, no shared implementation of the privacy probes, and no consensus reporting format. Authors working on adjacent problems, class-level forgetting in classification [34], class-centric segmentation unlearning [44], continual segmentation [25], each reuse their own evaluation infrastructure, with the result that cross-paper comparisons are essentially impossible. The protocol assembled above, defined by (2.25)–(2.32), is the protocol against which the framework proposed in this thesis is evaluated.

2.5 Key Findings and Research Gaps

The bodies of work surveyed in this chapter provide a substantial toolkit, but one that has been assembled around problems adjacent to, rather than coincident with, sequential instance-level unlearning in panoptic segmentation. The unlearning literature contributes a principled definition of the problem, a clear taxonomy of exact and approximate methods, and a growing collection of gradient-based, distillation-based, and saliency-based recipes [12], [18], [39], [40], [42], [44]. The dense-prediction literature contributes strong query-based panoptic architectures and a mature evaluation metric in the PQ decomposition of (2.3) [16], [29]. The multi-task and continual-learning literatures contribute distillation, gradient projection, and magnitude balancing as mechanisms for controlling interference between competing objectives or successive parameter updates [6], [7], [9], [11], [26], [31], [36]. Each strand is necessary, but none, on its own, is sufficient for the problem the present thesis takes on.

Four concrete gaps remain. First, almost the entire unlearning literature targets classification or, at most, semantic segmentation; panoptic segmentation, with its mixed stuff–things vocabulary and its query-based representational substrate of Section 2.2.1, is essentially absent. Second, the granularity at which existing methods operate is class-level or sample-level, not instance-level; mechanisms for erasing a single annotated object from a model that contains many other objects of the same class in the same image are not part of the standard repertoire, and the bi-objective

tension of (2.8) acquires its sharpest form at this granularity. Third, sequential unlearning, in which a long stream of deletion requests must be honoured without resurgence (4.35) and without cumulative degradation of utility, is recognised as a setting but has very few methodological commitments attached to it; bounded replay of previously forgotten material and re-identification of drifting forget queries are not standard practice. Fourth, the field lacks a shared evaluation protocol for panoptic instance forgetting that encompasses, under a single benchmark, the forgetting-quality measures of (2.26)–(2.28), the utility measures of (2.29)–(2.30), and the privacy and relearning measures of Section 2.4.4, and in particular one that reports the honest panoptic-output removal rate rather than a more favourable per-query proxy.

PAUSE is positioned to address these gaps directly. It treats panoptic segmentation as the target domain rather than as an afterthought, operates at the granularity of the individual annotated instance, resolves the forget/retain gradient conflict asymmetrically in favour of forgetting (4.43) while balancing the competing terms by gradient magnitude, localises the erasure to the instance region to limit class-level collateral, and accommodates sequential requests through a bounded replay memory and a student re-matching procedure designed to suppress resurgence and drift. The detailed mechanics of the framework, including its teacher–student instantiation, its retain-anchored distillation terms, its forget-priority projection, and the negative result on null-space projection that justifies that choice, are deferred to the chapters that follow; the role of the present chapter has been to establish that the gap they address is real and that the surrounding literature, while rich, does not close it.

Chapter 3

Requirements, Impacts and Constraints

In this chapter we refrain from the technical aspects of the development of the framework and examine the conditions and impacts of its implementation. We first examined the technical, resource and methodological requirements that the design of PAUSE needed to meet, before considering the impact of the design on a range of societal, environmental and ethical issues, then look at some of the standards it was subject to, followed by an examination of the main risks of the project and then an economic analysis of the costs of compliance for the framework. The framing of this paper is conservative. As PAUSE tackles a hard and under-studied problem, sequential instance-level unlearning in panoptic segmentation and the usefulness of this process is best seen as a good trade-off, rather than something that has been solved.

3.1 Final Specifications and Requirements

We divided requirements that determined PAUSE in three categories: technical and resource requirements of the operating regime; software requirements from the system which was implemented; methodological requirements to get an unlearning result that is considered defendable as opposed to a superficially benefiting result.
Technical and resource requirements: PAUSE operates on top of HuggingFace Mask2Former which is based on the Swin-B backbone pretrained on Cityscapes panoptic [5], [29] and the student is initialized from the teacher's weights. A key requirement imposed for development considerations by both the limited 24GB of VRAM in a consumer-grade GPU and by a preference for an ability to localize the parameter update, is to keep the Swin backbone network and the pixel decoder *frozen*; and solely train an additional trainable transformer decoder layers and head, a class predictor, a mask-embedding projection, and the object queries. By this we ensures the number of trainable parameters and per-request compute small enough to warrant running a complete deletion cycle on one workstation and also limits the update to the portion of the network under control of query-mediated assignment, a part where instance-level changes need to be effected. Requests for deletion are addressed over two stages, a "forgetting" stage and a "repair" stage, with a fixed step budget for each instance of the deletion process so that the cost of satisfying a deletion request is fixed before the work starts, and is not left to open-ended

optimizing.
Software and reproducibility requirements: We used PyTorch for the implementation of PAUSE, on the basis of hugging-face's transformers implementation of Mask2Former. Due to the specificity of Cityscapes one additional correctness condition applies: The uniqueness of the discrepancy between the label-ids of the panoptic annotation and train-ids of the model, in the specific classes to be unlearned must be handled with care. To compare with previous methods meaningfully, all of the frameworks and baselines must use the same experimental runner, share a common stage freezing mechanism, early stopping rule, step budget, forget-query matching and metrics, ensuring that only the per-step loss is different between methods [42].
Methodological requirements: There are three 'non-negotiable' commitments. First, utility preservation is required, since an instance-level edit results in a critical destruction of one or two thing classes can have a little or no impact on the global PQ, but have a significant impact on the class that included the target [16], [25]; so there's a strict need for the breakdown by per-class. Second, the forgetting should be calculated by the honest panoptic-output removal rate of, after the same panoptic post-processing that the deployed model uses, rather than the more favorable per-query proxy, which could disagree with the output rate of the actual panoptic renders. Third, as PAUSE is an approximate unlearning approach [39], an adversarial assessment of the residual influence must be confirmed, and naturally this should be an evaluation of the difference between the two that is performed by a membership-inference style probe, which compares the forgotten and never-seen cases. Our three requirement ensures that a claimed success is considered a removal from the rendered output and not a relabelling or a diluted aggregate.

3.2 Societal Impact

Our motivation for unlearning at the instance level comes significantly from a societal aspect. For data-protection laws, like the European General Data Protection Regulation (GDPR), the right to erasure is broadly interpreted as the right to remove the influence of this data rather than merely delete the data from storage of the model, after it has been used to train the model [14]. If a model is trained on the same type of settings as ours ones presented here (urban street scenes), where there are so many people and cars to be distinguished [5], it is a particularly relevant situation for the study of this right. Exactly those single pedestrian or single vehicle deletion runs that PAUSE can support, without the almost prohibitively high cost of retraining an operation, are a direct positive step towards individual privacy for deployed vision systems.
The same feature has a dual-edge sword character that needs clearly articulated. In an adversarial view, a surgical removal of a selected object from a model's perception -viewed as a mechanism of privacy, one may also use it to remove evidence of specific objects, persons and events from a system's output -may also be called a mechanism of selective suppression. The very behavior that makes PAUSE useful when it is applied to a specific occurrence makes it difficult to spot if it is being misused, as the surrounding scene is left intact and the whole class isn't deleted. We feel that it is more of a procedure than a technology: If a deletion request is in response to a safety relevant perception model, it should be logged, auditable and require an authorization, as well as the proper reporting of what the method is removing and

not removing (Section 3.4) should be a step towards protecting from over-ambitious reporting of what has been achieved. In addition there are safety aspects that are unique to dense perception. A per-class utility preservation as needed in Section 3.1 is as much a safety constraint as a quality control one. Because in an autonomous-driving or surveillance context, detecting an object of a class is a safety critical requirement, while an unlearning procedure that impacts the class as a whole, but not simply the specifically requested instance, could introduce a hazard.

3.3 Environmental Impact

This is our environmental argument for unlearning, and it rests on the comparison with its conceptually alternative idea. Removing the influence of a datum requires retraining of the model from scratch based on the data without the datum, which takes tens to hundreds of GPU hours for one full training run for panoptic model in the Mask2Former family [29], [39]. If that pipeline is rerun each time a request wants to delete something, the energy cost will be even larger, and can be economically and environmentally unsustainable [22], [23].
In these respects, PAUSE is a deliberately lightweight solution, in keeping with the aims of efficient, or "Green", AI [22]. The framework train deletion in less than an hour using just a single consumer GPU, as opposed to re-running a multi-GPU training job; freezing the Swin backbone and the pixel decoder and only updating the transformer decoder head; combining each request into a short forgetting stage and a lengthy repair stage under a fixed step budget. We are careful not to overstate this: it is still a method with iterative gradient updates, and requests are processed sequentially which means the per-request cost can be incurred over a series of deletions, thus only relative. Honestly, the marginal price of honouring a deletion request reduced by PAUSE by a large factor compared to retraining, which is the regime in which the deletion is likely to arise as a result of compliance.

3.4 Ethical Issues

There are three ethical points that are fundamental in this research. The first, and the more obvious, is the danger of giving the impression of security when in fact you're not. PAUSE is an approximation of an entire method, and approximation of unlearning cannot in general certify that no information or trace of the forgotten instance remains in the parameters [35], [39]. When a user is told an instance has been "forgotten," they may wrongly assume he is being assured that it has been deleted similar to a deletion of data from a database. We are addressing it in two ways: (1) adversarial measure of residual influence using three types of membership-inference style of probe, not just output behavior, and (2) honest measure of forgetting in terms of the panoptic output-removal rate which reveals the forgotten cases. The second is the familiar paradox addressed in Section 3.2: there is a tension between the need for privacy and the need for censorship when there is such an ability of precision. We assume that the research artefact should reflect this risk rather than consider the potential of benign use. The third is about the data itself. The dataset consists of cityscapes that carry real city scenes with identifiable subjects and vehicles [5]; the data is used in this work as part of published research under the research

terms of the dataset, and the targets evaluated in this work are not objects collected for this work, but rather the object instances of this published dataset. We believe all these factors collectively support our honest reporting, as well as limited and traceable use and explicit recognition of the boundaries of approximate erasure as a suitable attitude towards such work.

3.5 Standards

We have maintained several external standards and conventions frame the work. On the regulatory side, the right to erasure articulated in Article 17 of the GDPR is the legal standard that motivates the deletion capability and that informs the requirement to remove influence rather than merely storage [14]. On the evaluation side, our work adheres to the established panoptic-segmentation reporting convention, the Panoptic Quality metric together with its segmentation and recognition quality decomposition introduced with the task [16], and reports it in the class-stratified form that the literature on segmentation editing treats as necessary [25]. On the data side, We used Cityscapes in its standard official split, with the canonical 500-image validation set reserved for utility evaluation. So that the measurements are comparable to other work on the same benchmark [5]. On the software side, our implementation follows standard reproducibility practice within the PyTorch and HuggingFace ecosystem: fixed random seeds, a single shared runner for the framework and all baselines, and a faithful reimplementation of the comparator methods in their published form, with the components specific to PAUSE deliberately not retrofitted into the baselines so that the comparison remains fair [42].

3.6 Project Management Plan

The project followed a five-phase management plan spanning approximately sixteen weeks. Because PAUSE is organized as a sequential request-processing pipeline built on top of a frozen-backbone Mask2Former teacher, later stages depended directly on the outputs of earlier stages: the forgetting stage could not be designed before the panoptic evaluation harness was in place, and the repair stage could not be tuned before the forgetting stage produced measurable removal. This meant that phases could not overlap arbitrarily, although parallel work was still possible within each phase wherever dependencies allowed. Table 3.1 presents the overall project timeline, major activities, and expected deliverables.

Table 3.1: Project timeline for the development of PAUSE.

Phase	Duration	Activities	Deliverables
1	Weeks 1–3	Literature review, problem formulation, Mask2Former baseline setup	Design document and PyTorch environment
2	Weeks 3–6	Cityscapes data pipeline, per-class PQ evaluation harness, teacher–student initialization	Evaluation harness and frozen-backbone student
3	Weeks 6–9	Forgetting-stage design, forget-query matching, and instance-region masking	Working forgetting stage and removal results
4	Weeks 9–13	Repair stage, dynamic-weight controller, forget-priority projection, and baseline integration	Trained PAUSE pipeline and baseline models reimplementations
5	Weeks 13–16	Adversarial evaluation, ablation studies, and thesis writing	Ablation reports and completed thesis

The phased development approach ensured systematic progress, where each stage built upon the outputs of the previous phase. This structured workflow minimized integration risks and allowed parallel development of independent components such as the evaluation harness and the forget-query matcher. The timeline also enabled iterative refinement through feedback cycles, ensuring that critical issues such as the disagreement between per-query and post-processed forgetting measurements, and the absence of a usable null space in the decoder, were resolved during development. Figure 3.1 complements Table 3.1 by visualizing how the five phases were distributed across the sixteen-week schedule. The chart makes the sequential dependencies between phases easier to interpret, while also showing where limited parallel work was feasible within the broader plan. Together, Table 3.1 and Figure 3.1 provide both a tabular and visual view of the project-management structure used for PAUSE.

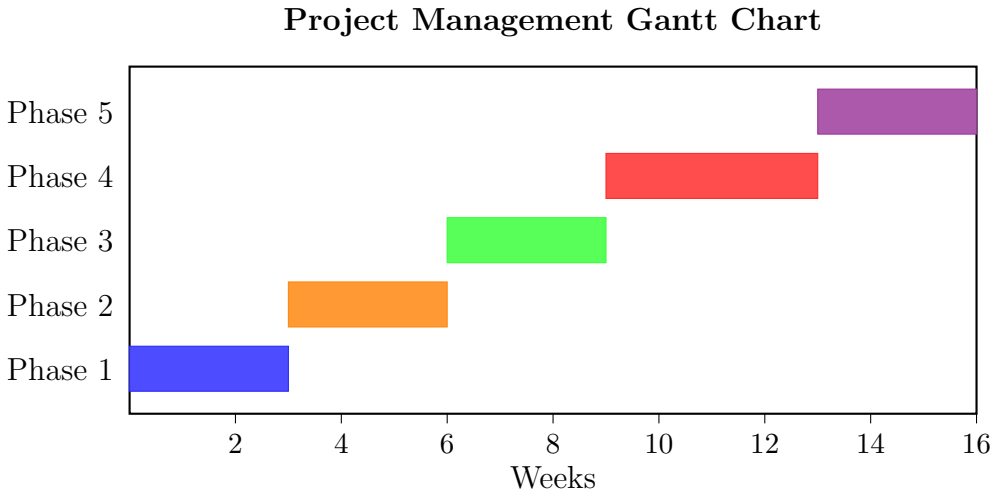


Figure 3.1: Gantt chart of the PAUSE project timeline.

Concurrency within phases helped keep the project on schedule. During Phase 2, baseline reproduction of the Mask2Former teacher on Cityscapes ran overnight on the single 24 GB GPU, freeing daytime hours for evaluation-harness design and forget-target identification. During Phase 4, the dynamic-weight controller and the forget-priority projection were developed in parallel, communicating through a scalar mixing coefficient applied when combining the forget and retain gradients.
Version control and reporting used a single-branch Git workflow with checkpointed experiment directories. Per-class PQ/SQ/RQ reports, ablations, and adversarial probe results were stored in numbered folders, keeping each iteration traceable and reducing confusion during sequential-deletion experiments where every request produces a new student checkpoint.
Two subsystems required mid-project revision. First, the original null-space projection was abandoned after evidence that the Mask2Former decoder modules are close to full rank and admit little usable null space [36]; it was replaced by a forget-priority projection that applies the forget gradient in full while projecting the retain gradient onto its orthogonal complement. Second, an early query-level forgetting metric disagreed with the actual panoptic output because suppressed queries could be revived during post-processing; it was replaced by an honest output-removal rate measured after the same post-processing the deployed model uses.

3.7 Risk Management

The most important risks for the success of this project are technical and follow directly from the difficulty of the problem; each of the risks is found in parallel with the design decision that contributes to mitigating it.

- **Incomplete forgetting:** Not all the marked target instances are removed. The easiest case to comprehend and dissect is a target that has been collateral suppressed in the past; possibly before its deletion step but is located in some region that the teacher asserts it with confidence and therefore, reintroduced by the broad class-preservation objective (which cannot tell the difference between "a car to keep" and "a car that was already forgotten"). This the key forget/retain stress. It can be mitigated (not eliminated), but not hidden thanks to the localization of the erasure to the instance region and the forget-priority projection, which ensures that the forget gradient is added in full.

- **Class-level collateral damage:** Censoring or suppressing one occurrence can damage the whole class in a rich and dense representation. When erasure is not localized on the instance-region level and without any certain retention term protecting all countable classes, the panoptic quality of the unlearned class dropped considerably even at the early design phase; this can be avoided by applying instance-region-restricted erasure and a thing-class retention term protecting all countable classes [25].

- **Resurgence and drift under sequential requests:** A deleted instance can re-appear in a stream after it has been forgotten, and the query to which an instance had been deleted can change as the student is changed. We mitigated

this by having bounded replay memory to forget previously encountered targets and by also re-suppressing forget queries for the current student without eradicating the teacher identified query.

- **Hardware limitation:** However, the development was limited to using a single 24 GB GPU so we can't optimize it to full model across relevant resolution. We mitigate this and reduce the computation by having a frozen-backbone, decoder-only design which is both a memory accommodation and also an deliberate localization of the update.

- **Limited generality:** The framework is validated in but for a single set of data and architecture; and the dynamic-weighting controller sets two parameters (hyperparameter) which would need to be tuned again on other data and architecture. This was identified as a limitation in scope and not mitigated and it is marked for further work. Another related design risk namely that a null-space projection would have avoided the forget/retain conflict a bit more gracefully led to an investigation on evidence but came up with a negative result. We found the panoptic decoder is close to full rank and few of its modules admit a usable null space, which inspired us to use forget-priority projection [36].

3.8 Economic Analysis

The economic case for our PAUSE is similar to its environmental argument, and is based on the cost of the other treatment option. While existing methods for retraining can differ in their complexity, they require a full training run in the range of tens to hundreds of GPU-hours per deletion, and such a training run, on commercial cloud machines, is not trivial. It is also more expensive on every deletion request than a straightforward evaluation of the model [23], [29], [39]. This "retrain per request" approach would be economically unfeasible with an operator having to tackle a stream of erasure requests.
Unlike this, our PAUSE will accept a request but respond with a limited number of gradient steps on the decoder head only, completing even on one consumer-level GPU. All the costs are kept at a minimum, the one that dominates is access to the pre-trained teacher, which the operator already owns; the marginal costs of every deletion thus are merely minutes of single-gpu compute, versus yet another full training run. Therefore, compliance is within the reach of operators with limited training facilities. Instead of being a free operation, a reduction in marginal costs as compared with retraining is an honest qualification in line with the spirit of Section 3.3. The value of our framework is that it allows a tractable per-query saving to be expressed as a small and predictable per-quarter value, which is the appropriate economic property.

Chapter 4

Proposed Methodology

4.1 Methodological Overview

This chapter develops **PAUSE** (*Post-Hoc Asymmetric Unlearning for Sequential Erasure*), the framework we propose for *sequential, instance-level unlearning* in a panoptic segmentation model. Given a model already trained on a panoptic dataset, PAUSE removes the influence of *individual object instances*—for example, one particular car in one particular image—one request at a time, while leaving the model's behaviour on every other instance, every other class, and the global scene parsing essentially unchanged.
We treat unlearning as a constrained optimisation problem and design every component as a response to that problem. The forgetting requirement is the *objective*; the requirement to preserve everything else is a *constraint*; and the shared, query-based representation of a panoptic decoder is what couples the two so tightly that they must be arbitrated explicitly rather than traded off with a single fixed weight.

4.1.1 Design Objectives and Constraints

PAUSE is designed against three objectives and four constraints, and every component in section 4.3 answers one of them.

Objectives.

O1 Instance-level removal. The targeted instance must disappear from the model's *rendered* panoptic output, not merely have its confidence reduced.

O2 Utility preservation. Global panoptic quality, and per-class quality on the classes involved, must remain close to the original model's; collateral on neighbouring instances and classes must be small.

O3 Sequential stability. An instance removed at step t must stay removed as later requests are processed, despite the model being edited again.

Constraints.

C1 Shared representation. The decoder has no per-instance parameter to delete; any edit acts through queries shared by many instances, so forgetting and retention are intrinsically coupled.

Table 4.1: Notation used in this chapter.

Symbol	Meaning
f_θ, θ	panoptic model and its parameters
θ_0	original (pretrained) parameters; the *teacher* $\mathcal{T} = f_{\theta_0}$
θ_t	student parameters after processing t requests; $\mathcal{S} = f_{\theta_t}$
Q, C	number of object queries; number of classes (incl. "no object" $\varnothing$)
z_q, $p_q \in \Delta^C$	class logits and class distribution of query q
$m_q \in [0,1]^{H\times W}$	soft mask of query q
$\Pi(\cdot)$	fixed panoptic post-processing / fusion operator
$\mathcal{D}$	dataset the model was trained on
(I_t, μ_t, c_t)	request t: image, binary instance mask $\mu_t \in \{0,1\}^{H\times W}$, class c_t
$\mathcal{F}_t$	cumulative forget set after t requests
$\mathcal{R}$	retain set (everything the model should still predict as $\mathcal{T}$)
$\mathcal{Q}_\mathcal{F}$	forget-query set (queries responsible for the target)
$g_\mathcal{F}$, $g_\mathcal{R}$	forget and retain gradients

C2 No retraining from scratch. The original training set is not assumed available for full retraining, and the compute budget is a single consumer GPU; the method must be a lightweight post-hoc edit.

C3 Bounded edit surface. To limit cost and to protect scene-level (stuff) parsing, only the decoder head is editable; the perception backbone and pixel decoder are frozen.

C4 Honest measurement. Success is judged by what the full panoptic post-processing actually renders, since a per-query score can fall while the instance still appears in the output.

4.1.2 Details of the Problem Formulation

Model and notation. Let $f_\theta : \mathcal{I} \to \mathcal{Y}$ be a panoptic segmentation model with parameters θ, mapping an image $I \in \mathcal{I}$ to a set of Q object predictions. For an image I, query $q \in \{1, \dots, Q\}$ emits a class distribution and a soft mask,

$$p_q(\cdot \mid I; \theta) = \sigma\big(z_q(I;\theta)\big) \in \Delta^C, \qquad m_q(\cdot \mid I; \theta) \in [0,1]^{H\times W}, \tag{4.1}$$

and a fixed, non-learnable fusion operator $\Pi(\cdot)$ converts the query set into a panoptic map. The symbols used throughout are collected in Table 4.1.

The request stream. A removal request is an instance specified by an image I_t, a binary mask $\mu_t \in \{0,1\}^{H\times W}$ delineating the instance, and its class c_t. Requests arrive as a stream and are applied *in place* by an update operator $\mathcal{U}$,

$$\theta_t = \mathcal{U}\big(\theta_{t-1};\, I_t, \mu_t, c_t\big), \qquad t = 1, \dots, T, \qquad \mathcal{F}_t = \mathcal{F}_{t-1} \cup \{(I_t, \mu_t, c_t)\}, \quad \mathcal{F}_0 = \varnothing. \tag{4.2}$$

Here θ_t is reached only by editing θ_{t-1} for request t; it is *never* obtained by retraining from θ_0. The retain set $\mathcal{R} = \mathcal{D} \setminus \bigcup_t \{(I_t, \mu_t, c_t)\}$ is everything the model should continue to predict as the original model does.

The ideal target (exact unlearning). Let $\mathcal{A}$ denote the training algorithm. The gold reference is the model retrained from scratch on the data with the forget instances removed,

$$\theta_t^\star \;=\; \mathcal{A}\big(\mathcal{D} \setminus \mathcal{F}_t\big). \tag{4.3}$$

Equation (4.3) defines *exact* unlearning: $f_{\theta_t^\star}$ is, by construction, a model that never saw the forget instances. It is intractable here—it requires the full dataset and a retraining run per request—and is ruled out by constraint C2. PAUSE instead pursues *approximate, empirical* unlearning: it seeks parameters whose *behaviour* matches the two defining properties of $\theta_t^\star$ without computing it.

Forgetting and retention as measurable conditions. The two properties are made precise as follows. *Forgetting* requires that, for every removed instance, no segment of its class survives in the rendered output:

$$\forall\,(I,\mu,c) \in \mathcal{F}_t : \qquad \nexists \text{ segment } s \in \Pi\big(f_{\theta_t}(I)\big) \text{ with } \mathrm{class}(s) = c \text{ and } s \cap \mu \neq \varnothing. \tag{4.4}$$

This condition is stated on the output of Π (objective O1, constraint C4), not on internal query scores, because the two can disagree. *Retention* requires behavioural agreement with the teacher on the retain set, measured by a divergence $d(\cdot,\cdot)$ between predictions:

$$\mathcal{L}_{\mathcal{R}}(\theta) \;=\; \mathbb{E}_{I\sim\mathcal{R}}\, d\big(f_\theta(I),\, f_{\theta_0}(I)\big) \;\leq\; \epsilon. \tag{4.5}$$

In (4.5), the comparison target is the *original* model f_{θ_0}: retention is defined as not departing from the model we started with on data unrelated to the request. This is precisely what makes f_{θ_0} a natural *teacher* and the retention term a distillation loss (subsection 4.3.1); $\epsilon \geq 0$ is the tolerated deviation.

The constrained unlearning problem. Writing $\mathcal{L}_{\mathcal{F}}(\theta)$ for a differentiable surrogate of the forgetting condition (4.4) (instantiated in subsection 4.3.2), PAUSE solves, for each request,

$$\boxed{\theta_t \;=\; \arg\min_\theta \;\mathcal{L}_{\mathcal{F}}(\theta) \quad \text{subject to} \quad \mathcal{L}_{\mathcal{R}}(\theta) \leq \epsilon.} \tag{4.6}$$

Equation (4.6) is the formal statement of instance-level unlearning in panoptic segmentation that the rest of the chapter implements: *forget the instance, subject to staying close to the original model everywhere else.*

Why a retain set is necessary. We now justify, rather than assume, the constraint in (4.6). Consider the unconstrained forget-only problem $\min_\theta \mathcal{L}_{\mathcal{F}}(\theta)$ used by a pure gradient-ascent baseline. Its set of minimisers,

$$\Theta_{\mathcal{F}} \;=\; \{\theta : \text{condition (4.4) holds}\}, \tag{4.7}$$

is a high-dimensional manifold: many parameter settings forget the instance, and $\mathcal{L}_{\mathcal{F}}$ assigns them all the same (minimal) value, expressing *no* preference among them for utility. The choice among members of $\Theta_{\mathcal{F}}$ is therefore underdetermined by forgetting alone. Worse, under the shared-representation constraint C1 the forget

gradient is not utility-neutral. Let $u(\theta) = f_\theta(I_R)$ be the model's prediction on a retain input I_R. Because the forget queries also serve retained instances,

$$\left\langle \nabla_\theta\, u(\theta),\ g_{\mathcal{F}}(\theta) \right\rangle \neq 0 \quad \text{in general,} \qquad g_{\mathcal{F}}(\theta) = \nabla_\theta \mathcal{L}_{\mathcal{F}}(\theta), \tag{4.8}$$

so a descent step $-\eta\, g_{\mathcal{F}}$ moves the retained prediction u by $-\eta \langle \nabla_\theta u, g_{\mathcal{F}} \rangle \neq 0$ with *no restoring force*—utility degrades freely. The retain term (4.5) supplies that restoring force: its gradient $g_{\mathcal{R}} = \nabla_\theta \mathcal{L}_{\mathcal{R}}$ penalises exactly such departures from f_{θ_0}. The constraint thus converts the ill-posed problem (4.7) into the well-posed problem (4.6), selecting from $\Theta_{\mathcal{F}}$ the forgetting solutions that are closest to the original model. The retain set is therefore not optional but *constitutive* of the method; its empirical necessity is confirmed by the collapse of the retain-free baseline in chapter 5.

Penalty form and the link to PAUSE. Relaxing the constraint with a multiplier $\lambda \geq 0$ gives the equivalent penalty problem

$$\min_\theta\ \mathcal{L}_{\mathcal{F}}(\theta) + \lambda\, \mathcal{L}_{\mathcal{R}}(\theta), \tag{4.9}$$

whose stationary points satisfy $g_{\mathcal{F}} + \lambda\, g_{\mathcal{R}} = 0$. A single fixed λ is inadequate here for two reasons that the remainder of the chapter addresses directly: the natural magnitudes of $g_{\mathcal{F}}$ and $g_{\mathcal{R}}$ differ and drift by orders of magnitude during the schedule, so no constant λ holds the intended balance (*dynamic weighting*, subsection 4.3.3); and when the two gradients point in opposing directions, simply summing them lets retention partially undo forgetting, whereas the constraint in (4.6) says forgetting must take priority (*forget-priority projection*, subsection 4.3.3). Finally, because the constraint must hold over the *cumulative* set $\mathcal{F}_t$ in (4.2), forgetting from earlier requests must be re-imposed during later ones (*replay*, subsection 4.3.4).

4.2 Target Set Creation

The request stream of Equation 4.2 is not arbitrary: each request must concern an instance the model genuinely predicts, the set must be unbiased across classes, and the whole construction must be reproducible. This subsection describes how the candidate instances are indexed, filtered for validity and difficulty, and sampled into the target set.

Candidate index. We perform a single teacher pass over the Cityscapes training split and, for every ground-truth thing instance, record its best-query IoU (4.15), the matched-query class probability $p^{\mathcal{T}}_{q^\star}(c)$, and the resulting validity score below. The result is a fixed index of candidate instances, computed once and reused so that every method and ablation draws from identical data.

Validity and difficulty. A request must concern an instance the model actually represents; asking it to forget something it never predicted is vacuous. Using the teacher's response to the ground-truth mask, we define the **predicted-instance validity**

$$\mathrm{PIV} \ = \ \Big(\max_q \mathrm{IoU}(\hat{m}^{\mathcal{T}}_q, \mu)\Big) \ \times \ p^{\mathcal{T}}_{q^\star}(c) \ \in [0,1], \tag{4.10}$$

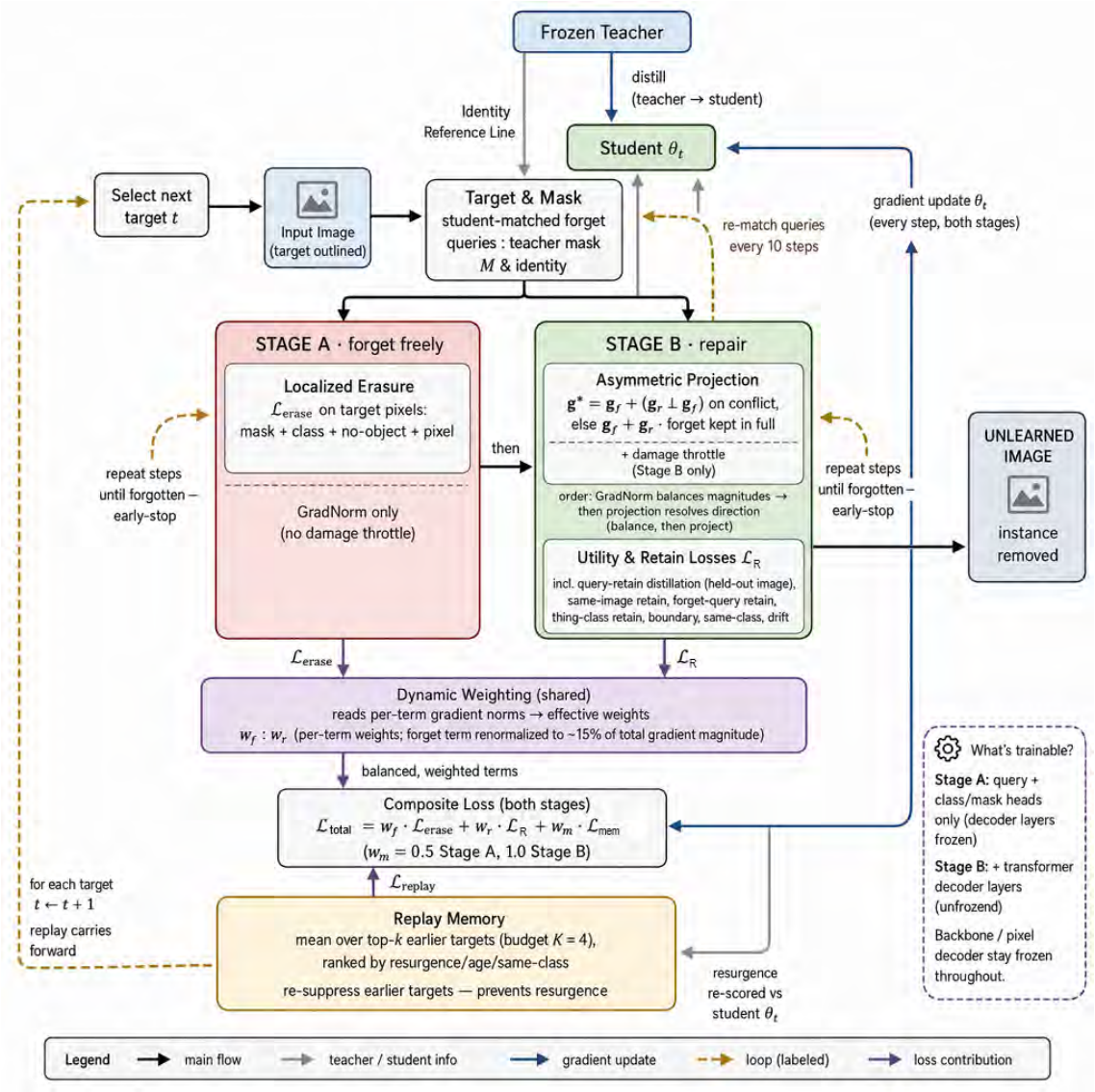


Figure 4.1: PAUSE methodology diagram for sequential targets unlearning.

where $\max_q \mathrm{IoU}(\hat{m}_q^{\mathcal{T}}, \mu)$ is the overlap of the best-matching teacher query with the instance mask μ (4.14), $q^\star$ is that best query (4.15), $p_{q^\star}^{\mathcal{T}}(c)$ is the teacher's probability that $q^\star$ has the instance's class c, and the product is high only when the teacher both *localises* the instance well and *classifies* it confidently. A candidate is deemed *valid* when its best-query IoU exceeds 0.3, its matched class probability exceeds 0.4, and its mask covers at least 1024 pixels (filtering out instances that are barely predicted, mislabelled, or too small to matter). Difficulty is graded by the best-query IoU as *easy* (≥ 0.6), *medium* (≥ 0.3), or *hard* (≥ 0.2): a lower starting overlap means the instance is more diffusely represented and harder to isolate cleanly.

Random balanced selection. From the valid candidates we keep those whose PIV falls in a chosen band $[\mathrm{PIV}_{\mathrm{lo}}, \mathrm{PIV}_{\mathrm{hi}}]$ (e.g. $[0.5, 1.0]$ for the well-predicted headline set), then sample the target set *at random*, balanced across the $K = 8$ thing classes, under a fixed seed. Concretely, the surviving candidates are grouped by class; for a

desired total of N targets the per-class quota is

$$n_k = \left\lfloor \tfrac{N}{K} \right\rfloor + \mathbf{1}[\, k \le (N \bmod K) \,], \qquad \textstyle\sum_k n_k = N, \tag{4.11}$$

so the N mod K leftover slots are spread one per class. A seeded pseudo-random generator (seed 42) shuffles each class pool and draws the first n_k instances uniformly without replacement; if a class is short of its quota the deficit is topped up from the shuffled remainder, and the chosen set is shuffled once more to randomise request order. In (4.11), N is the target-set size, K the number of thing classes, n_k the count drawn from class k, and $\mathbf{1}[\cdot]$ the indicator function. Random sampling avoids hand-picking favourable instances, the per-class balance prevents any class from dominating the per-class utility measurements of chapter 5, and the fixed seed makes the entire selection reproducible. Difficulty-specific sets (easy, hard) and the retain and validation splits below are drawn the same way with distinct seed offsets so that the splits do not overlap.

Retain pool and validation manifest. Two further splits are built from the same index. A held-out *retain* pool of training images is sampled to be disjoint from every selected target; it is the empirical realisation of the retain set $\mathcal{R}$ used by the retain distillation (4.5). A *validation manifest* fixes the 500-image official Cityscapes validation split over which whole-model panoptic quality is measured. Neither overlaps the target images.

Headline stream. The headline evaluation uses a curated stream of 10 targets (7 cars, 3 persons) drawn by the above procedure with PIV ≥ 0.5, of mid-range size, and isolated where possible, so that the instance to be removed is unambiguous.[1]

4.3 Design of the PAUSE Unlearning Framework

4.3.1 Teacher–Student Model and Instance Identification

The base model is a HuggingFace Mask2Former with a Swin-B backbone [29], [32], pretrained for panoptic segmentation on Cityscapes [5], which provides 19 evaluation classes of which 8 are countable *thing* classes (person, rider, car, truck, bus, train, motorcycle, bicycle) and the remainder are *stuff* classes.

Teacher–Student Isolation

PAUSE maintains two copies of the model. The **teacher** $\mathcal{T} = f_{\theta_0}$ is a frozen copy of the original model; its parameters are never updated and receive no gradient. The **student** $\mathcal{S} = f_\theta$ is initialised from the teacher, $\theta \leftarrow \theta_0$, and is the only model edited. By *isolation* we mean that the teacher is wholly decoupled from the optimisation: it is not differentiated through, not tracked by a moving average, and not updated between requests.

[1] The annotation files index classes by Cityscapes *label ids* (person=24, car=26); the model uses contiguous *train ids* (person=11, car=13). A fixed remapping is applied so that masks, class indices, and the protected thing-class set are expressed in train-id space.

Table 4.2: Module training policy. Backbone and pixel decoder are frozen throughout; the trainable set grows from Stage A to Stage B.

Module group	**Stage A**	**Stage B**
Swin backbone	frozen	frozen
Pixel decoder	frozen	frozen
Object queries / class predictor / mask embed.	trainable	trainable
Transformer decoder layers	frozen	trainable

This isolation is what makes the retain objective well-defined. Recall from (4.5) that retention is defined against the teacher's predictions, and consider its gradient,

$$g_{\mathcal{R}}(\theta) = \nabla_\theta \, \mathbb{E}_{I\sim\mathcal{R}} \, d\big(f_\theta(I),\, f_{\theta_0}(I)\big) = \mathbb{E}_{I\sim\mathcal{R}} \, \partial_1 d\big(f_\theta(I), f_{\theta_0}(I)\big) \, \nabla_\theta f_\theta(I), \tag{4.12}$$

where $\partial_1 d$ denotes the derivative of d in its first argument. The second argument $f_{\theta_0}(I)$ is a *constant* in θ, so it contributes no term to (4.12); the reference target does not move. Consequently

$$\mathcal{L}_{\mathcal{R}}(\theta) = 0 \iff f_\theta \equiv f_{\theta_0} \text{ on } \mathcal{R}, \tag{4.13}$$

i.e. the retain objective is zero exactly when the student reproduces the original behaviour. Had the reference instead been the student's own (moving) output or an exponential moving average of it—a self-distillation target—then the second argument of d would drift together with the student. The very degradation we seek to prevent would be written into the target, and $\mathcal{L}_{\mathcal{R}}$ could be driven small while utility collapsed, because the moving target follows the model down. Freezing the teacher prevents this *contamination of the retained knowledge*: it pins an uncontaminated, unmoving anchor for every distillation term.

The teacher plays a second, distinct reference role: it fixes instance *identity*. The target mask μ_t, the validity/difficulty assessment, and the responsible-query identification at request start are all computed against the frozen teacher (section 4.3.1), so "which instance is being removed" never drifts as the student is edited, even though "which student query currently carries it" does. Both roles—a fixed behavioural anchor and a fixed identity reference—require a frozen teacher, and together they yield *stable* forgetting (the retain anchor cannot drift) and *targeted* forgetting (the instance region is established once against an unchanging reference).

Frozen and Trainable Modules

The Swin backbone and the pixel decoder produce dense, scene-level features and are not where a single instance's identity is decided; we freeze them for the entire run (constraint C3) and confine all editing to the transformer decoder head. Freezing the pixel pathway has a direct utility benefit: stuff classes are read out almost entirely from it and are thereby structurally protected. The editable parameters grow between the two stages, as summarised in Table 4.2.

Instance Identification by IoU Matching

To act on an instance we must resolve the queries responsible for it. Let $\mu \in \{0,1\}^{H\times W}$ be a binary target mask and let $\hat{m}_q = \mathbf{1}[m_q > \tau_m]$ be a thresholded query

mask ($\tau_m = 0.5$). With the binary intersection-over-union

$$\mathrm{IoU}(\hat{m}_q, \mu) \;=\; \frac{|\hat{m}_q \cap \mu|}{|\hat{m}_q \cup \mu|}, \tag{4.14}$$

the best-matching query and the forget-query set are

$$q^\star = \arg\max_q \; \mathrm{IoU}(\hat{m}_q, \mu), \tag{4.15}$$

$$\mathcal{Q}_\mathcal{F} = \big\{\, q : \mathrm{IoU}(\hat{m}_q, \mu) > \tau_{\mathrm{IoU}} \,\big\} \;\cup\; \{q^\star\}, \qquad \tau_{\mathrm{IoU}} = 0.2. \tag{4.16}$$

In (4.14)–(4.16), $|\cdot|$ counts foreground pixels; τ_m binarises the soft masks; τ_{IoU} is the overlap above which a query is deemed responsible for the instance; and the union with $q^\star$ guarantees $\mathcal{Q}_\mathcal{F} \neq \varnothing$ even when no query clears τ_{IoU}.
A key sequential subtlety is *which model* (4.16) is run against. Instance *identity* (μ, validity, difficulty) is always taken from the teacher so that it cannot drift (section 4.3.1). The *query to suppress*, however, is matched against the student's *current* output: earlier edits push the student away from the teacher, so the query carrying the instance in the live student can differ from the teacher's stale query. Suppressing the teacher's query while the instance has re-routed to another student query is a wrong-direction update that leaves the instance untouched; matching on the student is what makes suppression act on the instance as the model currently represents it.

4.3.2 Localised Erasure and Retention

This subsection instantiates the surrogate forget objective $\mathcal{L}_\mathcal{F}$ and the retain objective $\mathcal{L}_\mathcal{R}$ of (4.6). Each optimisation step uses the weighted total

$$\mathcal{L}_{\mathrm{total}}(\theta) \;=\; w_\mathcal{F}\, \mathcal{L}_{\mathrm{erase}}(\theta) \;+\; \underbrace{\sum_{j \in \mathcal{R}\text{-terms}} w_j\, \mathcal{L}_j(\theta)}_{\mathcal{L}_\mathcal{R}(\theta)}, \tag{4.17}$$

where $\mathcal{L}_{\mathrm{erase}}$ is the single forget term, the sum is the retain objective composed of eight terms (Table 4.3), and the weights $w_\mathcal{F}, w_j$ are *not* fixed scalars but are set per step by the dynamic weighter of subsection 4.3.3; Table 4.4 lists their *base* ratios.

The Forget Objective

The forget term is a weighted sum of four sub-objectives,

$$\mathcal{L}_{\mathrm{erase}} = \lambda_{\mathrm{mask}}\, \mathcal{L}_{\mathrm{mask}} + \lambda_{\mathrm{cls}}\, \mathcal{L}_{\mathrm{cls}} + \lambda_{\mathrm{null}}\, \mathcal{L}_{\mathrm{null}} + \lambda_{\mathrm{pix}}\, \mathcal{L}_{\mathrm{pix}}, \tag{4.18}$$

with $(\lambda_{\mathrm{mask}}, \lambda_{\mathrm{cls}}, \lambda_{\mathrm{null}}, \lambda_{\mathrm{pix}}) = (2.0, 1.0, 1.5, 1.0)$. Each sub-objective is defined and explained in turn; m_q, p_q denote the *student*'s mask and class distribution, $\mathcal{Q}_\mathcal{F}$ the forget-query set (4.16), and μ the target region.

Query-mask erasure.

$$\mathcal{L}_{\mathrm{mask}} = -\,\frac{1}{|\mathcal{Q}_\mathcal{F}|\,|\mu|} \sum_{q \in \mathcal{Q}_\mathcal{F}} \sum_{(x,y) \in \mu} \log\big(1 - m_q(x,y)\big). \tag{4.19}$$

Table 4.3: The nine terms of $\mathcal{L}_{\text{total}}$, by side. The forget side is the composite erase term (4.18); the eight retain terms instantiate $\mathcal{L}_{\mathcal{R}}$ (4.5).

Term	**Side**	**Role**
erase $\mathcal{L}_{\text{erase}}$	forget	Suppress the target instance (composite; (4.18)).
retain $\mathcal{L}_{\text{ret}}$	retain	Distil the teacher on held-out retain images.
same-image retain $\mathcal{L}_{\text{si}}$	retain	Distil the non-forget queries on the target image.
forget-query retain $\mathcal{L}_{\text{fq}}$	retain	Preserve the forget queries on *non*-target pixels.
thing-class retain $\mathcal{L}_{\text{tc}}$	retain	Distil class presence for *all* thing classes off-target.
boundary $\mathcal{L}_{\text{bd}}$	retain	Preserve masks in a band around the target boundary.
same-class $\mathcal{L}_{\text{sc}}$	retain	Protect the target class's presence off-target.
replay $\mathcal{L}_{\text{mem}}$	retain	Re-suppress previously forgotten targets (subsection 4.3.4).
drift $\mathcal{L}_{\text{dr}}$	retain	Anchor the student to an EMA of its past weights.

Table 4.4: Base loss weights per stage. These set the *intended ratios* on which the dynamic weighter (subsection 4.3.3) operates; it rescales them per step by measured gradient magnitude ((4.38)–(4.39)). The damage controller modulating the erase weight is active in Stage B only.

Term	**Stage A**	**Stage B**
erase $\mathcal{L}_{\text{erase}}$	1.0	1.0
retain $\mathcal{L}_{\text{ret}}$	0.5	1.0
same-image retain $\mathcal{L}_{\text{si}}$	1.0	1.0
forget-query retain $\mathcal{L}_{\text{fq}}$	1.0	1.0
thing-class retain $\mathcal{L}_{\text{tc}}$	1.0	1.5
boundary $\mathcal{L}_{\text{bd}}$	0.5	1.0
same-class $\mathcal{L}_{\text{sc}}$	1.0	1.5
replay $\mathcal{L}_{\text{mem}}$	0.5	1.0
drift $\mathcal{L}_{\text{dr}}$	0.01	0.05

This drives each forget query's mask probability towards 0 on the target pixels. The sum is restricted to $(x, y) \in \mu$ (the target region) and to $q \in \mathcal{Q}_{\mathcal{F}}$ (the responsible queries); the $\log(1 - \cdot)$ form grows without bound as $m_q \to 1$, applying strong pressure where the mask is still confident. Because only target pixels are indexed, the gradient on a query's mask *outside* μ is exactly zero, so other instances the query serves are untouched here.

Localised class erasure. Define the per-query, per-pixel class-presence score

$$s_q(x, y) \;=\; p_q(c)\, m_q(x, y) \;\in [0, 1], \tag{4.20}$$

the product of query q's class-c probability and its mask value at (x, y). The localised class-erase term suppresses this score only where the query mask and the target overlap:

$$\mathcal{L}_{\text{cls}} = -\,\frac{1}{|\mathcal{Q}_{\mathcal{F}}|\,|\mu|} \sum_{q \in \mathcal{Q}_{\mathcal{F}}} \sum_{(x,y) \in \mu} \log\big(1 - s_q(x, y)\big). \tag{4.21}$$

This is the key design choice that controls collateral. A forget query is typically a *general* class detector serving many same-class instances; collapsing its *global* class-c

logit would degrade detection of all cars or persons. By tying the signal to s_q on μ only, the erase pressure is attached to the target *instance* rather than the class, and the gradient vanishes off-target.

No-object promotion.

$$\mathcal{L}_{\text{null}} = -\frac{1}{|\mathcal{Q}_{\mathcal{F}}|} \sum_{q \in \mathcal{Q}_{\mathcal{F}}} \log p_q(\varnothing), \tag{4.22}$$

where $p_q(\varnothing)$ is the probability mass on the "no object" class (whose index is derived automatically from the model configuration). Maximising $p_q(\varnothing)$ turns suppression into genuine *reassignment* of the forget queries to "nothing here", rather than a mere reduction of the class-c score that could be re-interpreted as another class.

Pixel-class erasure. Aggregate the per-query scores (4.20) over queries with a noisy-OR,

$$P_c(x,y) \;=\; 1 - \prod_q \big(1 - s_q(x,y)\big) \;\in [0,1], \tag{4.23}$$

the probability that *at least one* query asserts class c at pixel (x, y) under an independence assumption, and suppress it on the target:

$$\mathcal{L}_{\text{pix}} = -\frac{1}{|\mu|} \sum_{(x,y)\in\mu} \log\big(1 - P_c(x,y)\big). \tag{4.24}$$

Unlike a softmax-normalised sum, the noisy-OR in (4.23) stays in $[0, 1]$ and neither saturates nor dilutes with the number of queries, so it keeps a usable gradient regardless of how many queries assert the class.

The Retain Objective

All retain terms are temperature-scaled distillations of the student towards the frozen teacher [3], which (by section 4.3.1) is a fixed reference. We define two reusable operators. For a query set $\mathcal{A}$, the *class distillation* is

$$\mathcal{K}_\tau(\mathcal{A}) = \frac{\tau^2}{|\mathcal{A}|} \sum_{q\in\mathcal{A}} \mathrm{KL}\Big(\sigma_\tau\big(z_q^{\mathcal{T}}\big) \,\Big\|\, \sigma_\tau\big(z_q^{\mathcal{S}}\big)\Big), \qquad \sigma_\tau(z)_k = \frac{e^{z_k/\tau}}{\sum_{k'} e^{z_{k'}/\tau}}, \tag{4.25}$$

and, for a spatial region Ω and query set $\mathcal{A}$, the *masked mask-discrepancy* is

$$\mathcal{M}_\Omega(\mathcal{A}) = \frac{1}{|\mathcal{A}|\,|\Omega|} \sum_{q\in\mathcal{A}} \sum_{(x,y)\in\Omega} \big(m_q^{\mathcal{S}}(x,y) - m_q^{\mathcal{T}}(x,y)\big)^2. \tag{4.26}$$

In (4.25), $z_q^{\mathcal{T}}, z_q^{\mathcal{S}}$ are teacher/student class logits, τ is the distillation temperature, σ_τ the temperature-softened softmax, $\mathrm{KL}(\cdot\|\cdot)$ the Kullback–Leibler divergence from the teacher distribution to the student's, and the τ^2 factor restores the gradient scale [3]. In (4.26), $m_q^{\mathcal{S}}, m_q^{\mathcal{T}}$ are student/teacher soft masks and Ω selects the pixels over which agreement is enforced. Let $\mathcal{Q} = \{1, \ldots, Q\}$ be the full query set, $\mathcal{Q}_{\mathcal{F}}$ the forget-query set (4.16), $\bar{\mu}$ the complement of the target region (all pixels outside μ),

and w_m the weight balancing the class- and mask-distillation parts. The first three retain terms apply these operators on three disjoint domains.
The *held-out retain* term distils the teacher over *all* queries on a retain image I_R drawn from $\mathcal{R}$:

$$\mathcal{L}_{\text{ret}} = \mathcal{K}_\tau(\mathcal{Q}) + w_m \, \mathcal{M}_{\Omega_{\text{all}}}(\mathcal{Q}), \qquad I_R \sim \mathcal{R}, \tag{4.27}$$

where Ω_{all} is the whole image. It guards against generic drift on scenes unrelated to the request and is the empirical instance of the retain objective (4.5).
The *same-image retain* term distils the *non-forget* queries ($\mathcal{Q} \setminus \mathcal{Q}_\mathcal{F}$) on the target image itself, restricted to the pixels *outside* the target ($\bar{\mu}$):

$$\mathcal{L}_{\text{si}} = \mathcal{K}_\tau(\mathcal{Q} \setminus \mathcal{Q}_\mathcal{F}) + w_m \, \mathcal{M}_{\bar{\mu}}(\mathcal{Q} \setminus \mathcal{Q}_\mathcal{F}), \tag{4.28}$$

where $\mathcal{Q} \setminus \mathcal{Q}_\mathcal{F}$ excludes the queries being suppressed and $\bar{\mu}$ excludes the target region. It protects the other instances visible in the target's own frame from in-frame collateral.
The *forget-query retain* term distils the forget queries $\mathcal{Q}_\mathcal{F}$ on the non-target pixels $\bar{\mu}$:

$$\mathcal{L}_{\text{fq}} = \mathcal{K}_\tau(\mathcal{Q}_\mathcal{F}) + w_m \, \mathcal{M}_{\bar{\mu}}(\mathcal{Q}_\mathcal{F}). \tag{4.29}$$

Here the *same* queries that are suppressed on μ by the erase terms (4.19)–(4.21) are *preserved* on $\bar{\mu}$: because a forget query is a general detector serving many instances, (4.29) retains the other instances it serves and is the exact off-target complement of the localised erasure.
The remaining terms protect class-presence and geometry. Using the pixel class-presence map (4.23) for the student ($P_k^\mathcal{S}$) and teacher ($P_k^\mathcal{T}$), and the off-target hard/soft regions $\Omega_h = \{(x,y) \notin \mu : P_c^\mathcal{T} > 0.3\}$ and $\Omega_s = \{(x,y) \notin \mu : 0.15 < P_c^\mathcal{T} \leq 0.3\}$, the *same-class* term is

$$\mathcal{L}_{\text{sc}} = \frac{1}{|\Omega_h|} \sum_{(x,y)\in\Omega_h} \left(P_c^\mathcal{S} - P_c^\mathcal{T}\right)^2 + \frac{1}{2\,|\Omega_s|} \sum_{(x,y)\in\Omega_s} \left(P_c^\mathcal{S} - P_c^\mathcal{T}\right)^2, \tag{4.30}$$

which penalises the student for diverging from the teacher's class-c presence *away from* the target, weighting confidently-class regions (Ω_h) fully and ambiguous regions (Ω_s) at half. Extending this to every thing class $k \in \mathcal{C}_{\text{thing}}$ (with region $\Omega_k = \{(x,y) \notin \mu : P_k^\mathcal{T} > 0.15\}$) gives the *thing-class retain* term

$$\mathcal{L}_{\text{tc}} = \frac{1}{|\mathcal{C}_{\text{thing}}|} \sum_{k\in\mathcal{C}_{\text{thing}}} \frac{1}{|\Omega_k|} \sum_{(x,y)\in\Omega_k} \left(P_k^\mathcal{S} - P_k^\mathcal{T}\right)^2, \tag{4.31}$$

which was added because suppressing a car or person otherwise leaks onto neighbouring thing classes (rider, motorcycle, bicycle). With B the radius-r band around the target boundary, the *boundary* term is $\mathcal{L}_{\text{bd}} = \mathcal{M}_B(\mathcal{Q})$ ($r = 3$), and the *drift* term anchors the trainable weights to a slow reference θ^{ref},

$$\mathcal{L}_{\text{dr}} = \left\|\theta - \theta^{\text{ref}}\right\|_2^2, \tag{4.32}$$

where θ^{ref} is initialised at θ_0 and advanced after each request by an exponential moving average towards the student ($\theta^{\text{ref}} \leftarrow (1-\alpha)\theta^{\text{ref}} + \alpha\theta$, $\alpha = 0.1$), limiting cumulative movement across the stream. The replay term $\mathcal{L}_{\text{mem}}$, the last term of (4.17), is defined next; its selection and eviction policy is deferred to subsection 4.3.4.

Replay (memory) loss. Because the forget constraint must hold over the cumulative set $\mathcal{F}_t$ (4.2), editing the shared decoder for a later request can re-awaken an instance forgotten earlier. The replay term re-imposes the forgetting condition on previously processed targets. At each step we draw a budget of b past targets from the replay memory—selected, and evicted at capacity, by the priority defined in subsection 4.3.4—and, for each, measure how strongly it has *resurged* in the student's *current* output. For a replay item j with stored image I_j, target mask μ_j and class c_j, let $m_q^{(j)}$ denote the student's soft masks and $P_{c_j}^{\mathcal{S}}$ its noisy-OR class-presence (4.23) on I_j. Resurgence is scored by a differentiable class residue and a soft mask residue. The *class residue* is the mean class-c_j presence the student still places on the erased region,

$$r_{\text{cls}}^{(j)} = \frac{1}{|\mu_j|} \sum_{(x,y)\in\mu_j} P_{c_j}^{\mathcal{S}}(x,y), \tag{4.33}$$

and the *soft mask residue* replaces the thresholded overlap of (4.14) with a differentiable soft IoU between any student query mask and the target,

$$\text{IoU}_{\text{soft}}\big(m_q^{(j)}, \mu_j\big) = \frac{\sum_{(x,y)} m_q^{(j)}(x,y)\,\mu_j(x,y)}{\sum_{(x,y)} m_q^{(j)}(x,y) + \sum_{(x,y)} \mu_j(x,y) - \sum_{(x,y)} m_q^{(j)}(x,y)\,\mu_j(x,y) + \varepsilon}. \tag{4.34}$$

The per-item resurgence score combines the two,

$$\rho^{(j)} = r_{\text{cls}}^{(j)} + \lambda_{\text{IoU}} \max_q \ \text{IoU}_{\text{soft}}\big(m_q^{(j)}, \mu_j\big), \qquad \lambda_{\text{IoU}} = 1, \tag{4.35}$$

a soft, differentiable counterpart of the forgetting condition (4.4) evaluated on I_j; λ_{IoU} balances the class and mask residues. The replay loss is the mean resurgence score over the usable subset $\mathcal{B}$ of the drawn budget (items whose stored mask and class are valid on the current student),

$$\mathcal{L}_{\text{mem}} = \frac{1}{|\mathcal{B}|} \sum_{j\in\mathcal{B}} \rho^{(j)}. \tag{4.36}$$

Minimising $\mathcal{L}_{\text{mem}}$ drives both residues back towards zero, re-suppressing exactly the past targets most at risk of returning and so enforcing the cumulative constraint that single-request erasure cannot. The soft form (4.34) is what is *optimised*; the hard, thresholded overlap used to *report* resurgence in chapter 5 is its non-differentiable diagnostic counterpart, and the two need not agree exactly.

4.3.3 Asymmetric Gradient Balancing and Forget-Priority Projection

The penalty form (4.9) hides two difficulties that a single fixed λ cannot resolve: the forget and retain gradients have different and *drifting* magnitudes, and they often point in *opposing* directions. PAUSE resolves these in order—first *magnitudes* (section 4.3.3), then *directions* (section 4.3.3). The guiding principle is *balance, then project.*

Dynamic Gradient Weighting

The problem with fixed weights. A term's influence on the update is its *contribution* $c_i = w_i \, \|g_i\|$, the product of its weight and its gradient magnitude. Because the raw magnitudes $\|g_i\|$ differ across terms and change by orders of magnitude over the schedule, fixing w_i leaves c_i uncontrolled: a retain term whose gradient happens to be large can swamp the forget term even at equal nominal weight, and the realised forget/retain balance wanders. We instead set w_i adaptively so that the *contributions* hit an intended ratio.

Magnitude equalisation. Following GradNorm [9], we first equalise each term's gradient magnitude to a common reference and then re-apply the base ratios. Let $\tilde{g}_i$ be an exponential moving average of $\|g_i\|$,

$$\tilde{g}_i \leftarrow \beta \, \tilde{g}_i + (1 - \beta) \, \|g_i\|, \qquad \beta = 0.9, \tag{4.37}$$

which smooths step-to-step noise (the EMA is reset at the start of each stage, since a new instance is a fresh optimisation). With the reference $\bar{g} = \frac{1}{|\mathcal{R}\text{-terms}|} \sum_j \tilde{g}_j$ taken as the mean retain-term norm, the equalised weight of term i with base weight w_i^{base} (Table 4.4) is

$$w_i^{\text{eq}} = w_i^{\text{base}} \, \frac{\bar{g}}{\tilde{g}_i + \varepsilon}. \tag{4.38}$$

Each term now contributes magnitude $\approx \bar{g}$ scaled by its base ratio, so the base ratios—not the accidental raw scales—determine the balance.

Forget-fraction renormalisation. We then rescale the forget weight so that the forget term holds a fixed share $\rho = 0.15$ of the total contribution. With $C_{\mathcal{F}} = w_{\mathcal{F}}^{\text{eq}} \tilde{g}_{\mathcal{F}}$ and $C_{\mathcal{R}} = \sum_j w_j^{\text{eq}} \tilde{g}_j$, the requirement $\frac{s \, C_{\mathcal{F}}}{s \, C_{\mathcal{F}} + C_{\mathcal{R}}} = \rho$ is met by

$$w_{\mathcal{F}} \leftarrow w_{\mathcal{F}}^{\text{eq}} \cdot s, \qquad s = \frac{\rho \, C_{\mathcal{R}}}{(1 - \rho) \, C_{\mathcal{F}}}. \tag{4.39}$$

Holding ρ small but non-zero realises the constraint reading of (4.6): forgetting is given a controlled, bounded slice of the update, and the majority of the gradient budget is reserved for retention.

Damage-aware modulation. In the repair stage only (subsection 4.3.4), a proportional controller modulates the forget weight to hold the measured same-class/boundary damage D near a budget $D^\star = 0.05$:

$$w_{\mathcal{F}} \leftarrow \kappa \, w_{\mathcal{F}}, \qquad \kappa = \text{clip}\left(\left(\frac{D^\star}{D} \right)^{\gamma}, \, 0.1, \, 2.0 \right), \qquad \gamma = 2.0. \tag{4.40}$$

When damage is below budget ($D < D^\star$) the multiplier $\kappa > 1$ gently boosts forgetting; when damage overshoots ($D > D^\star$) the squared exponent makes κ shrink sharply, throttling forgetting; the clip bounds the intervention. The controller is disabled in the "forget freely" stage, where unconstrained suppression is intended.

Stability guards and why dynamic beats fixed. Three guards bound the adaptation. A hard clamp keeps each effective weight within a factor of 8 of its base ($w_i \in [w_i^{\text{base}}/8,\, 8\, w_i^{\text{base}}]$), preventing the explosion that (4.38) would cause when some $\tilde{g}_i \to 0$. A *retain floor* rescales the retain side up if its aggregate share would fall below 0.6, preventing retain starvation. A *forget floor* keeps $w_{\mathcal{F}}$ at $\geq 10\%$ of its equalised value when the retain magnitude is momentarily ≈ 0 (at a target's first step, where the student still equals the teacher and (4.39) would otherwise zero the forget weight). The net effect is that the *realised* contribution ratio tracks ρ throughout training regardless of the raw magnitudes—something no constant λ in (4.9) can do across a 160-step schedule and a T-request stream. Dynamic weighting is thus the adaptive Lagrange multiplier of the constrained problem: it improves *stability* (clamp and EMA prevent thrashing and explosions), *forgetting effectiveness* (the forget floor guarantees initial progress), and *utility* (the retain floor and damage controller protect the retain side).

Forget-Priority Projection

After balancing, the forget and retain gradients can still *conflict* in direction. Let

$$g_{\mathcal{F}} = \nabla_\theta\big(w_{\mathcal{F}}\mathcal{L}_{\text{erase}}\big), \qquad g_{\mathcal{R}} = \nabla_\theta\, \mathcal{L}_{\mathcal{R}} = \sum_j \nabla_\theta\big(w_j \mathcal{L}_j\big), \tag{4.41}$$

be the (weighted) forget and retain gradients. A conflict is the event

$$\langle g_{\mathcal{R}}, g_{\mathcal{F}} \rangle < 0, \tag{4.42}$$

i.e. a negative inner product: a descent step on retain would, in part, undo forgetting, and vice versa.

Conflict resolution. Unlearning is *asymmetric*: by (4.6) forgetting is the objective and retention the constraint, so the two should not be treated even-handedly. Symmetric gradient surgery such as PCGrad [26] would, under (4.42), project *both* gradients onto each other's normal plane, weakening the forget signal. PAUSE instead delivers the forget gradient *in full* and projects only the retain gradient onto the orthogonal complement of $g_{\mathcal{F}}$:

$$g_{\mathcal{R}}^{\perp} = \begin{cases} g_{\mathcal{R}} - \dfrac{\langle g_{\mathcal{R}}, g_{\mathcal{F}} \rangle}{\|g_{\mathcal{F}}\|^2}\, g_{\mathcal{F}}, & \langle g_{\mathcal{R}}, g_{\mathcal{F}} \rangle < 0, \\ g_{\mathcal{R}}, & \text{otherwise}, \end{cases} \qquad g = g_{\mathcal{F}} + w_r\, g_{\mathcal{R}}^{\perp}, \tag{4.43}$$

where g is the update gradient applied by the optimiser and w_r $(= 1)$ scales the orthogonalised retain contribution. Geometrically, $g_{\mathcal{R}}^{\perp}$ is the projection of $g_{\mathcal{R}}$ onto the hyperplane normal to $g_{\mathcal{F}}$ (Figure 4.2): the component of retention that fights forgetting is removed, and only its non-conflicting part survives.

Optimisation behaviour: a guaranteed forget-descent direction. The asymmetry yields a property symmetric surgery does not. Since $g_{\mathcal{R}}^{\perp} \perp g_{\mathcal{F}}$ in the conflicting case (and $\langle g_{\mathcal{R}}, g_{\mathcal{F}} \rangle \geq 0$ otherwise), the inner product of the update with the forget gradient is

$$\langle g, g_{\mathcal{F}} \rangle = \|g_{\mathcal{F}}\|^2 + w_r\, \langle g_{\mathcal{R}}^{\perp}, g_{\mathcal{F}} \rangle = \|g_{\mathcal{F}}\|^2 \;\geq 0. \tag{4.44}$$

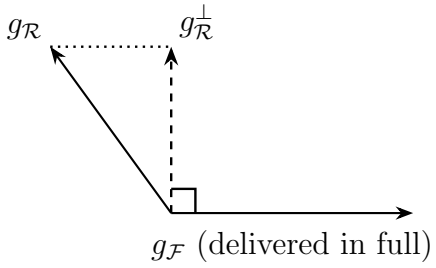


Figure 4.2: Asymmetric forget-priority projection (4.43). When the retain gradient $g_{\mathcal{R}}$ conflicts with the forget gradient $g_{\mathcal{F}}$, only the component $g_{\mathcal{R}}^{\perp}$ orthogonal to $g_{\mathcal{F}}$ is kept; the forget gradient is never altered. The combined update therefore satisfies $\langle g, g_{\mathcal{F}} \rangle = \|g_{\mathcal{F}}\|^2 \geq 0$.

Hence the gradient step $-\eta g$ is always a descent direction for the forget loss,

$$\frac{\mathrm{d}}{\mathrm{d}\eta} \mathcal{L}_{\text{erase}}\big(\theta - \eta g\big)\Big|_{\eta=0} = -\langle g_{\mathcal{F}}, g \rangle = -\|g_{\mathcal{F}}\|^2 \ \leq 0, \tag{4.45}$$

so forgetting makes first-order progress at *every* step, no matter what retention asks for. Meanwhile retention contributes its full non-conflicting descent: its first-order effect is $-\langle g_{\mathcal{R}}, g \rangle = -\langle g_{\mathcal{R}}, g_{\mathcal{F}} \rangle - w_r \|g_{\mathcal{R}}^{\perp}\|^2$, in which the projection has stripped out exactly the part of $g_{\mathcal{R}}$ that pulled against $g_{\mathcal{F}}$. This is the precise sense in which the method "prioritises forgetting while preserving utility": (4.45) guarantees the objective of (4.6) always advances, while the constraint is honoured as far as it can be without reversing that advance.

Multiple retain terms. With the eight retain terms of (4.17), each gradient $g_{\mathcal{R}_j} = \nabla_\theta(w_j \mathcal{L}_j)$ is projected independently against the *same* fixed $g_{\mathcal{F}}$ and the results summed,

$$g = g_{\mathcal{F}} + \sum_j w_r \, g_{\mathcal{R}_j}^{\perp}, \qquad g_{\mathcal{R}_j}^{\perp} \perp g_{\mathcal{F}} \ \text{ when conflicting.} \tag{4.46}$$

Because every component is projected against one fixed direction rather than against one another, (4.44) still holds ($\langle g, g_{\mathcal{F}} \rangle = \|g_{\mathcal{F}}\|^2$), and no projection can undo another's effect—the sequential-reprojection conflict of symmetric multi-task surgery cannot arise. The symmetric (PCGrad) variant is kept only as an ablation comparator. A null-space alternative—constraining updates to the null space of the retained-feature covariance, after Adam-NSCL [36]—was prototyped and found unsuitable for this near-full-rank decoder; that negative result, which further motivates the forget-priority choice, is reported in chapter 5.

4.3.4 The Sequential Repair Loop

Each request is processed in two stages with separate optimisers (Table 4.5). The two-stage design is a core contribution: it *separates achieving forgetting from recovering utility*, turning the coupled problem (4.6) into two better-conditioned subproblems. Stage A drives the instance out using a small, high-leverage set of parameters and no projection; Stage B then repairs the retained behaviour under the forget-priority projection while holding the instance down.

Table 4.5: The two-stage schedule applied to every request. Step counts are upper bounds; early stopping (subsection 4.3.5) ends a stage sooner once its criteria are met (subject to the minimum).

	Stage A — "forget freely"	**Stage B — "repair"**
Trainable	queries, class pred., mask embed.	+ transformer decoder layers
Projection	none	asymmetric (forget-priority)
Damage controller	off	on
Learning rate	1×10^{-5}	5×10^{-6}
Steps (max/min)	40 / 15	120 / 40
Distil. temp. τ	2.0	1.2

Stage A: Forget Freely

Inputs. The current student θ_{t-1} (already edited for requests $1, \ldots, t-1$), the request (I_t, μ_t, c_t), the forget-query set $\mathcal{Q}_\mathcal{F}$ matched on the *student* (4.16), a held-out retain batch, and replay items. The trainable parameters are the shallow head only—object queries, class predictor, mask embedding (Table 4.2)—and a fresh AdamW optimiser [13] is created with learning rate 10^{-5}.

Objective and optimisation. Stage A minimises $\mathcal{L}_{\text{total}}$ (4.17) with projection *disabled* and the damage controller *off*; the dynamic weighter still balances magnitudes (section 4.3.3). The retain terms are present but down-weighted (Table 4.4), and the update is the plain weighted sum $-\eta(g_\mathcal{F} + g_\mathcal{R})$. "Freely" refers to the absence of the projection/damage constraints, letting forgetting dominate the early steps where $g_\mathcal{F}$ is strongest.

Output and contribution. The stage returns a student θ_t^{A} in which the instance is largely suppressed at the head level: the forget queries are pushed towards "no object" (4.22) and their target-region mask and class scores are driven down (4.19)–(4.21). Because only the shallow, high-leverage parameters move, this is a low-dimensional edit that erases the instance quickly without yet disturbing the deeper decoder layers.

Why it is necessary. Editing the head first means the deeper repair in Stage B begins from an *already-forgotten* state and only has to *hold* the forget while restoring utility. Attempting both at once from step 0—under projection, with the student still equal to the teacher so $g_\mathcal{R} = 0$ and (4.39) ill-conditioned—would slow initial forgetting and entangle the two objectives. Separating them makes each sub-problem better-conditioned.

Stage B: Repair

Transition from Stage A. Stage B is *initialised* from θ_t^{A}; the forget-query set is inherited (and may have grown via re-matching during Stage A, section 4.3.4); and a fresh AdamW optimiser (learning rate 5×10^{-6}) and a reset weighter EMA are created. The key handoff is that Stage B inherits an instance that is *already suppressed*, so its role is constrained repair, not initial erasure.

Inputs. θ_t^{A}, the same request and (inherited) forget set, retain and replay batches. The trainable set now *additionally* includes the transformer decoder layers (Table 4.2), giving the capacity to re-route retained instances around the edited queries.

Objective and optimisation. Stage B minimises $\mathcal{L}_{\text{total}}$ with the asymmetric pro-

jection *on* (section 4.3.3) and the damage controller *on* ((4.40)), at higher retain/same-class/thing-class weights and a sharper distillation temperature $\tau = 1.2$. The update is the feasible-direction step $-\eta\, g$ of (4.43)/(4.46): by (4.44) the forget loss keeps descending while retention contributes only its non-conflicting part.

Output and contribution. The stage returns θ_t, the result of processing request t: the instance remains removed (the in-full forget gradient plus replay hold it down) while the shared representation perturbed by Stage A is repaired, recovering per-class quality and reducing collateral. After the stage the EMA anchor θ^{ref} is advanced (4.32) and the target is added to replay memory.

Why it is necessary. Stage A alone leaves the whole class degraded (constraint C1); without Stage B utility is unacceptable. The deeper decoder layers must move to repair, but they can be moved safely only once forgetting is established and the projection guards retention from re-admitting the instance.

Replay Memory

Because the forget constraint must hold over the cumulative set $\mathcal{F}_t$ ((4.2)), editing the shared decoder for request t can re-awaken an instance forgotten earlier. A replay memory stores each completed request (its image reference, mask, class, and matched queries) up to capacity 50. At every step a budget of $b = 4$ past targets is drawn and re-suppressed through a mask-based replay loss $\mathcal{L}_{\text{mem}}$ (the "memory" term of (4.17)). Selection is by a priority

$$\pi(i) = \alpha\, r_i + \beta\, a_i + \gamma\, \mathbf{1}[\, c_i = c_{\text{cur}}\,], \qquad (\alpha, \beta, \gamma) = (1.0,\, 0.1,\, 0.5), \tag{4.47}$$

where r_i is item i's most recently measured resurgence, a_i its age in the stream, and the indicator favours items of the current request's class c_{cur}; the same priority governs eviction at capacity. Replay thus re-imposes (4.4) on the past targets most at risk of returning.

Student Re-matching

Two re-matching mechanisms keep $\mathcal{Q}_{\mathcal{F}}$ aligned with the live student under drift (section 4.3.1). *Across* requests, the forget set is matched against the student's current output. *Within* a stage, every 10 steps the target mask is re-matched against the student's current masks and any newly responsible queries are *unioned* into $\mathcal{Q}_{\mathcal{F}}$; the teacher-matched queries are re-unioned at each re-match and so can never be dropped, only added to. This handles the case where the instance re-routes mid-stage to a query outside the original forget set that the retain terms would otherwise protect—a direct cause of partial forgetting.

The end-to-end procedure. algorithm 1 ties the components together. Across the stream the teacher and frozen modules are held fixed; only the decoder head moves, the EMA anchor is advanced after each request, and the replay memory accumulates the edits so far.

4.3.5 Early Stopping and Stability Safeguards

Because the stream is long and the forget/retain coupling tight, several criteria decide *when* to stop a stage and surface collateral before it compounds.

Convergence early-stop. A stage halts once the instance is suppressed and held suppressed: over a patience window of the last 10 steps, the target's residual class response stays below 0.15 and its residual mask IoU below 0.20, subject to a per-stage minimum (15 steps for A, 40 for B). The window prevents stopping on a single lucky step.

Emergency early-stop. If forgetting is incomplete but the measured boundary or same-class damage exceeds twice its budget (2×0.05), the stage stops immediately: further steps would only deepen the collateral without completing the forget, so "stop" is the safe action under a runaway-damage trajectory.

Adaptive erase throttle. Complementing the per-step controller (4.40), a slower safeguard halves the (already renormalised) forget weight whenever the three-step average same-class or boundary damage exceeds 0.05. The two controls differ in timescale—one per-step and proportional, one averaged and hard—and the throttle is a no-op on the healthy trajectory where damage stays within budget.

Sequential collateral circuit breaker. A diagnostic guard watches for the method's principal failure mode. If a request the teacher recognised at selection (PIV ≥ 0.5) arrives *already* unrecognised by the student before its own step (pre-edit PIV < 0.2), this signals that same-class collateral from earlier requests has bled forward and degraded the class. The run records the event and, optionally, halts if it recurs on more than two consecutive requests. We surface this rather than suppress it: it is direct evidence of the forget/retain tension characterised in chapter 5.

4.3.6 Baselines and Ablation Design

Baselines. We re-implement three unlearning baselines that share PAUSE's stage freezing, step budget, matching, and metrics through a common runner, so only the per-step loss differs. *NegGrad* performs pure gradient ascent on the forget loss (the retain-free $\min_\theta \mathcal{L}_\mathcal{F}$ of subsection 4.1.2), with no retention, projection, or replay. *NegGrad+* adds a teacher-distilled retain term. *SCRUB* [42] alternates maximising the student–teacher divergence on the forget set with distilling on the retain set. To keep the comparison fair, the baselines use the published teacher-only matching of their original formulations; the student re-matching of section 4.3.4 is part of PAUSE's contribution and is deliberately *not* retrofitted into them. Table 4.6 summarises the mechanisms each uses.

Ablation conditions. Each contribution is isolated by a single-component ablation, all else held constant (Table 4.7); the corresponding results appear in chapter 5.

4.3.7 Evaluation Protocol and Metrics

We fix metric definitions here and apply them consistently in chapter 5.

Forgetting. PIV (5.1) is a per-query measure of whether the responsible query still predicts the instance. The honest headline forgetting metric is the **panoptic-removal rate**: we run the same full panoptic post-processing Π used for visualisation and test condition (4.4)— whether any segment of the target's class still overlaps μ. This matches what renders (objective O1, constraint C4) and can disagree with PIV, so we report both and treat the panoptic-removal rate as primary.

Utility. Utility is the change in **panoptic quality** (PQ, with components SQ and RQ) [16], reported both globally on the 500-image validation split and *per class*, since a near-unchanged global PQ can hide a concentrated cost on the unlearned classes; we therefore always report the per-class breakdown alongside the global figure.

Collateral, resurgence, membership. Collateral is quantified by same-class and boundary damage; late resurgence by the change in PIV from a request's own completion to the end of the stream (positive values indicate a returned instance). Where a membership-inference test is run, an AUC near 0.5 indicates a forgotten instance is indistinguishable from one never seen.

4.3.8 Implementation and Reproducibility Details

All experiments run on a single consumer GPU (an RTX 4090 with 24 GB), under mixed-precision (AMP) training and inference. Optimisation uses AdamW [13] with weight decay 0.01 and no gradient accumulation; a fresh optimiser and a reset weighter EMA are created at the start of each stage, reflecting that each request–stage is a distinct optimisation. A fixed random seed makes target selection, matching, and the training loop reproducible. For efficiency in the multi-retain projection (4.46), each term's gradient is computed once per step and reused for both the dynamic-weight magnitudes and the projection, avoiding a second backward pass. The operative hyperparameters are collected in Table 4.8; of these, the two parameters of the balance—the forget fraction ρ (4.39) and the damage budget $D^\star$ (4.40)—are dataset-dependent and would require re-tuning on a different dataset or architecture, as examined in chapter 5.

Algorithm 1: PAUSE: per-request sequential instance unlearning.

Input: teacher $\mathcal{T}$, student $\mathcal{S} * \theta$ with $\theta \leftarrow \theta_0$, request stream $(I_t, \mu_t, c_t) * t = 1^T$, retain pool $\mathcal{D}_{\mathcal{R}}$, replay memory $\mathcal{M} \leftarrow \varnothing$, EMA anchor $\theta^{\text{ref}} \leftarrow \theta_0$

freeze $\mathcal{T}$ and the student backbone/pixel encoder;

for $t = 1 \dots T$ **do**

- obtain teacher prediction on I_t and establish target identity $(\mu_t, \text{PIV}, \text{difficulty})$ using (5.1);
- ‘‘‘ match teacher queries $\mathcal{Q}_{\mathcal{T}}$ and student queries $\mathcal{Q}_{\mathcal{S}}$ to μ_t using (4.16)
- **if** *target is invalid* **then**
  - **continue**
- $\mathcal{Q}_{\mathcal{F}} \leftarrow \mathcal{Q}_{\mathcal{S}}$ if the student match is valid, otherwise $\mathcal{Q}_{\mathcal{T}}$
- $\mathcal{Q}_{\text{seed}} \leftarrow \mathcal{Q}_{\mathcal{T}}$
- `// Stage A: free forgetting`
- set head/query parameters trainable; projection=none; damage-throttle=off; lr $= 10^{-5}$; fresh AdamW; reset weighter
- **for** step$= 1 \dots 40$ **do**
  - every 10 steps: re-match on the student and update $\mathcal{Q}_{\mathcal{F}} \leftarrow \mathcal{Q}_{\mathcal{F}} \cup \mathcal{Q}_{\text{seed}} \cup \text{Match}(\mathcal{S}_\theta(I_t), \mu_t)$
  - compute forget, retain, replay, and drift losses using (4.17); sample replay items from $\mathcal{M}$ using (4.47)
  - balance loss magnitudes using (4.37)–(4.39); take an AdamW step on the weighted objective
  - **if** *early stop holds and step*≥ 15 **then**
    - **break**
- `// Stage B: repair`
- add decoder layers to trainable parameters; projection=asymmetric; damage-throttle=on; lr $= 5 \times 10^{-6}$; fresh AdamW; reset weighter
- **for** step$= 1 \dots 120$ **do**
  - every 10 steps: re-match and update $\mathcal{Q}_{\mathcal{F}}$ with the teacher seed and current student match
  - compute forget, retain, replay, and drift losses; balance weights using (4.37)–(4.40)
  - compute $g_{\mathcal{F}}$ and retain-side gradients $\{g_{\mathcal{R},k}\}$; project retain gradients onto $g_{\mathcal{F}}^{\perp}$ using (4.43)–(4.46)
  - take an AdamW step using $g_{\mathcal{F}} + \sum_k \widetilde{g}_{\mathcal{R},k}$
  - **if** *early stop holds and step*≥ 40 **then**
    - **break**
- update θ^{ref} using (4.32); add request t to $\mathcal{M}$; age replay items and update resurgence scores
- ‘‘‘

Table 4.6: Baselines compared against PAUSE. All share the same freezing, step budget, and metrics; only the per-step loss and mechanisms differ. (✓ present, – absent.)

Method	**Retain term**	**Projection**	**Replay**	**Student re-match**
NegGrad [42]	–	–	–	–
NegGrad+ [42]	✓	–	–	–
SCRUB [42]	✓	–	–	–
PAUSE	✓	✓	✓	✓

Table 4.7: Ablation conditions. Each removes or replaces one component to test the design claim it supports.

Condition	**Design claim tested**
no replay	Replay is needed to prevent cross-request resurgence.
no projection	Direction conflict between forget and retain harms the trade-off.
symmetric vs. asymmetric	Forget-priority projection beats even-handed (PCGrad-style) surgery.
no same-class / global erase	Localised erasure and class protection reduce class-level collateral.
no Stage A (optional)	The unconstrained "forget freely" stage is necessary before repair.

Table 4.8: Operative hyperparameters of PAUSE.

Group	**Parameter**	**Value**
Matching	mask thr. τ_m / IoU thr. τ_{IoU}	0.5 / 0.2
	within-stage re-match interval	10 steps
Erase weights	$(\lambda_{\text{mask}}, \lambda_{\text{cls}}, \lambda_{\text{null}}, \lambda_{\text{pix}})$	$(2.0, 1.0, 1.5, 1.0)$
Dynamic weights	forget fraction ρ	0.15
	min. retain fraction	0.60
	EMA β / max weight ratio	0.9 / 8.0
Damage control	budget $D^\star$ / gain γ / clip	0.05 / 2.0 / $[0.1, 2.0]$
Early stop	class / mask residue thr. / patience	0.15 / 0.20 / 10
	emergency damage factor	2.0
Projection	mode (Stage A / Stage B)	none / asymmetric
Replay	capacity / per-step budget b	50 / 4
	priority (α, β, γ)	$(1.0, 0.1, 0.5)$
Distillation	temperature τ (Stage A / Stage B)	2.0 / 1.2
Boundary	band radius r	3 px
Drift	EMA anchor α	0.1
Optimiser	AdamW weight decay / precision	0.01 / AMP

Chapter 5

Result Analysis

This chapter evaluates PAUSE on the task it was designed for: sequential, instance-level unlearning in a panoptic segmentation model. We first set out the evaluation criteria and protocol, then report the method's utility preservation, its forgetting quality, and its behaviour across the removal sequence. We analyse the design through qualitative output and a set of component ablations, apply statistical tests to the per-target measurements, and finally place PAUSE against three unlearning baselines. Throughout, the central finding is reported honestly: PAUSE attains near-complete utility preservation while removing the majority of targeted instances, and the small number of instances that resist removal expose a forgetting–retention tension that we characterise rather than conceal.

5.1 Performance Evaluation

5.1.1 Evaluation Criteria and Testing Protocol

All experiments use the HuggingFace Mask2Former model with a Swin-B backbone [29], pre-trained on Cityscapes panoptic [5]. The teacher is frozen and a student is initialised from its weights; only the transformer decoder head is trainable, while the Swin backbone and pixel decoder remain frozen. Unlearning is applied sequentially to ten curated targets (seven cars, three persons), each well-predicted by the teacher (initial PIV ≥ 0.5 where possible), mid-sized and isolated. Global utility is measured on the official 500-image Cityscapes validation split before and after the full sequence.

Two families of criteria are used. *Utility* is measured by panoptic quality (PQ) and its segmentation- and recognition-quality components (SQ, RQ), globally and per class, following the standard panoptic protocol [16]; the headline utility statistic is the change in global PQ, ΔPQ. *Forgetting* is measured at the level of the individual target. We define the predicted-instance validity,

$$\mathrm{PIV} \;=\; \max_{q} \mathrm{IoU}(q,\, t) \;\times\; p_{\mathrm{class}}(q), \tag{5.1}$$

a per-query quantity equal to the best query–target mask overlap weighted by that query's matched class probability; class- and mask-residue are the residual class and mask responses on the target region after unlearning. The honest removal criterion is the **panoptic-removal rate**: the target's image is passed through the same full panoptic post-processing used for visualisation, and the target is counted

as removed only if no segment of its class overlaps the target region. This is the quantity that matches what actually renders, and it is the primary forgetting metric in this chapter. A per-query target suppression success (TSS) is reported as a secondary measure; as Section 5.2.3 shows, the two can disagree.

5.1.2 Global Utility Preservation

PAUSE leaves global utility almost unchanged. Table 5.1 reports global PQ, SQ and RQ together with the underlying true-positive (TP), false-positive (FP) and false-negative (FN) counts before and after the full ten-target sequence. Global PQ moves from 0.6453 to 0.6427, a change of -0.0026 (about -0.26 percentage points), i.e. 99.6% of the teacher's panoptic quality is retained. SQ and RQ are likewise essentially flat (-0.003 and -0.0005).

Table 5.1: Global panoptic utility on the 500-image Cityscapes validation split, before and after the full sequential unlearning of all ten targets (PAUSE). PQ/SQ/RQ are essentially preserved; the detection counts show fewer detections with a slight precision gain.

Metric	Before	After	Δ
PQ	0.6453	0.6427	-0.0026
SQ	0.8200	0.8170	-0.0030
RQ	0.7773	0.7768	-0.0005
True positives	10646	10039	-607
False positives	3987	3169	-818
False negatives	4025	4632	$+607$

The detection-count changes are consistent with targeted removal rather than indiscriminate degradation. True positives fall by 607 and false negatives rise by the same amount, as expected when instances are withdrawn from the output; notably, false positives fall further still (-818), so the model becomes marginally more precise overall. Figure 5.1 visualises these shifts. The pattern is what one would hope to see from a method that suppresses specific instances while protecting the surrounding prediction.

5.1.3 Per-Class Utility Analysis

The small global cost is concentrated, as designed, on the unlearned and closely related thing-classes; stuff-classes are untouched. Table 5.2 lists per-class PQ before and after, and Figure 5.2 shows the change as a diverging bar chart.
The two unlearned classes lose only 1.47 pp (car) and 1.65 pp (person) of PQ. The largest single cost is on motorcycle (-3.25 pp); this is a low-frequency class with high evaluation variance and is not itself targeted, so the change is better read as collateral sensitivity than as targeted damage. Crucially, every stuff-class moves within roughly ± 2 pp of its starting value and several *increase* (wall $+1.81$, terrain $+1.50$, sky $+1.17$); because the pixel decoder is frozen, these fluctuations are evaluation noise rather than a signal. The localisation of cost to a handful of thing-classes is the intended behaviour and is the direct consequence of the localised erasure and thing-class retention objectives, which we ablate in Section 5.3.

Table 5.2: Per-class PQ before and after unlearning (PAUSE), grouped into thing- and stuff-classes. The two unlearned classes are **car** and **person**. Δ is in percentage points (pp). The largest cost falls on *motorcycle*, a low-frequency, high-variance class; stuff-classes vary only within noise, consistent with the frozen pixel decoder.

Class	PQ before	PQ after	Δ (pp)
Thing-classes (instance)			
person †	0.5241	0.5076	−1.65
rider	0.5098	0.4939	−1.59
car †	0.6880	0.6733	−1.47
truck	0.5450	0.5257	−1.93
bus	0.7251	0.7259	+0.08
train	0.7036	0.7041	+0.05
motorcycle	0.4567	0.4242	−3.25
bicycle	0.4512	0.4446	−0.66
Stuff-classes (frozen pixel decoder)			
road	0.9802	0.9753	−0.49
sidewalk	0.7895	0.7911	+0.16
building	0.8982	0.8952	−0.30
wall	0.4502	0.4683	+1.81
fence	0.4527	0.4676	+1.49
pole	0.5827	0.5827	+0.00
traffic light	0.5671	0.5743	+0.72
traffic sign	0.7405	0.7370	−0.35
vegetation	0.9054	0.9027	−0.27
terrain	0.3963	0.4113	+1.50
sky	0.8944	0.9061	+1.17

† Unlearned class.

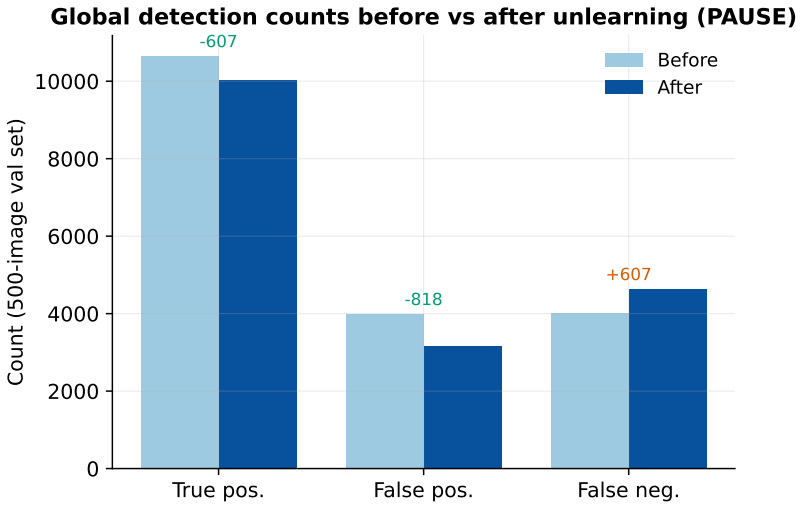


Figure 5.1: Global true-/false-positive and false-negative counts before and after unlearning (PAUSE). Fewer detections overall, with false positives reduced more than true positives.

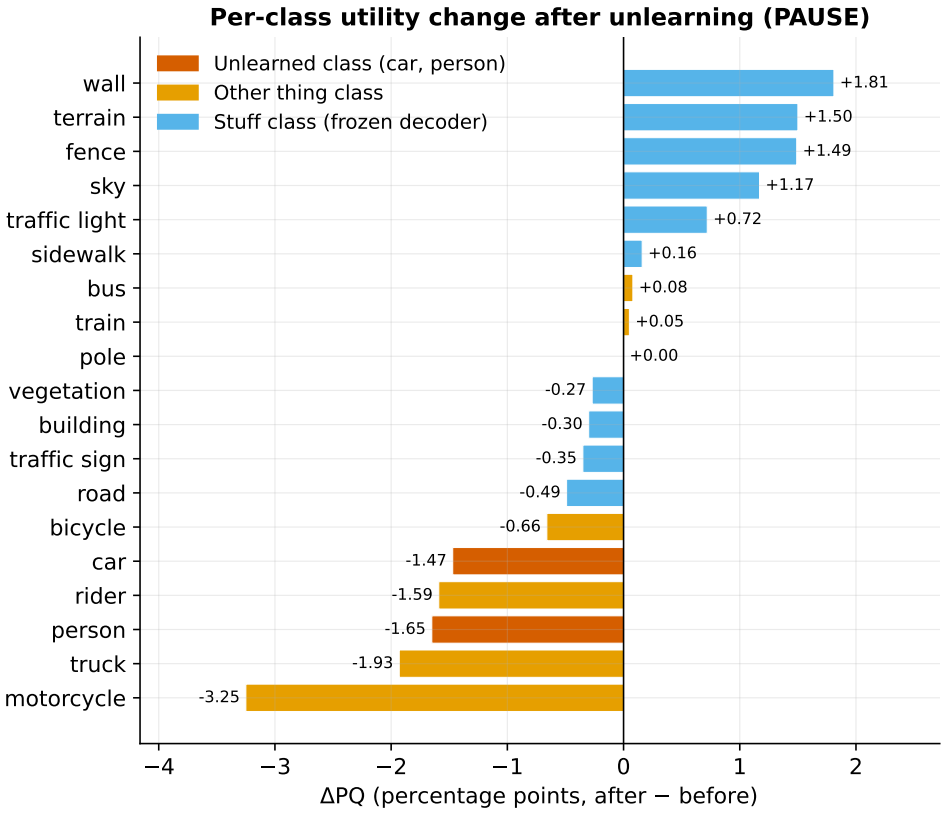


Figure 5.2: Per-class ΔPQ after unlearning (PAUSE), sorted. The unlearned classes (red) lose 1.5–1.7 pp; related thing-classes (orange) a little more in the low-frequency cases; stuff-classes (blue) fluctuate within noise around zero, several rising, as expected from the frozen pixel decoder.

5.1.4 Instance Forgetting Quality

PAUSE removes the majority of targeted instances. Across the ten targets the panoptic-removal rate is **8/10**, the per-query final TSS is 9/10, and the after-phase per-query TSS is 7/10 (Table 5.3). The mean per-target PIV falls from 0.708 before unlearning to 0.077 immediately after each target's own step, and to 0.074 at the end of the sequence; mean class residue falls from 0.673 to 0.137 and mean mask residue from 0.740 to 0.099. The per-target breakdown is given in Table 5.4 and visualised in Figure 5.3.

Table 5.3: Aggregate forgetting metrics for PAUSE across the ten targets. The three success rates measure forgetting differently and need not agree; the panoptic-removal rate is the primary (rendered-output) metric.

Quantity	Before	After / Final
Mean PIV (after own step)	0.7076	0.0767
Mean PIV (final)	0.7076	0.0736
Mean class residue	0.6729	0.1367
Mean mask residue	0.7397	0.0994
After-phase TSS (per-query)	7/10	
Final TSS (per-query)	9/10	
Panoptic-removal rate	**8/10**	
Mean same-class damage	0.1535	
Mean boundary damage	0.0073	

Table 5.4: Per-target forgetting outcomes (PAUSE). PIV is reported before unlearning, immediately after the target's own step, and at the end of the sequence. "aTSS"/"fTSS" are the per-query after-/final suppression successes; "Rem." is panoptic removal from the rendered output; "Res." is resurgence (final − after PIV). T9 is the hard failure and T10 the soft failure discussed in Section 5.2.1.

ID	Class	PIV_{bef}	PIV_{aft}	PIV_{fin}	aTSS	fTSS	Rem.	Res.
T1	car	0.9547	0.0013	0.0000	✓	✓	✓	−0.001
T2	car	0.8648	0.0031	0.0007	✓	✓	✓	−0.002
T3	person	0.8313	0.0376	0.0079	✓	✓	✓	−0.030
T4	car	0.9030	0.4008	0.0008	×	✓	✓	−0.400
T5	person	0.6275	0.0431	0.0041	×	✓	✓	−0.039
T6	car	0.8440	0.1492	0.0000	×	✓	✓	−0.149
T7	car	0.9369	0.0014	0.0001	✓	✓	✓	−0.001
T8	car	0.8793	0.0560	0.0630	✓	✓	✓	+0.007
T9	car	0.0001	0.0110	0.5901	✓	×	×	+0.579
T10	person	0.2345	0.0633	0.0692	✓	✓	×	+0.006

Eight of the ten targets are driven to near-zero PIV and remain suppressed through the rest of the sequence. The two exceptions are informative rather than random. T9 begins with PIV ≈ 0.0001 – it was already collaterally suppressed before its own unlearning step – and is the single hard failure: it resurges to PIV 0.59 and reappears in the output. T10 reaches low PIV (0.069) and passes the per-query criterion, yet

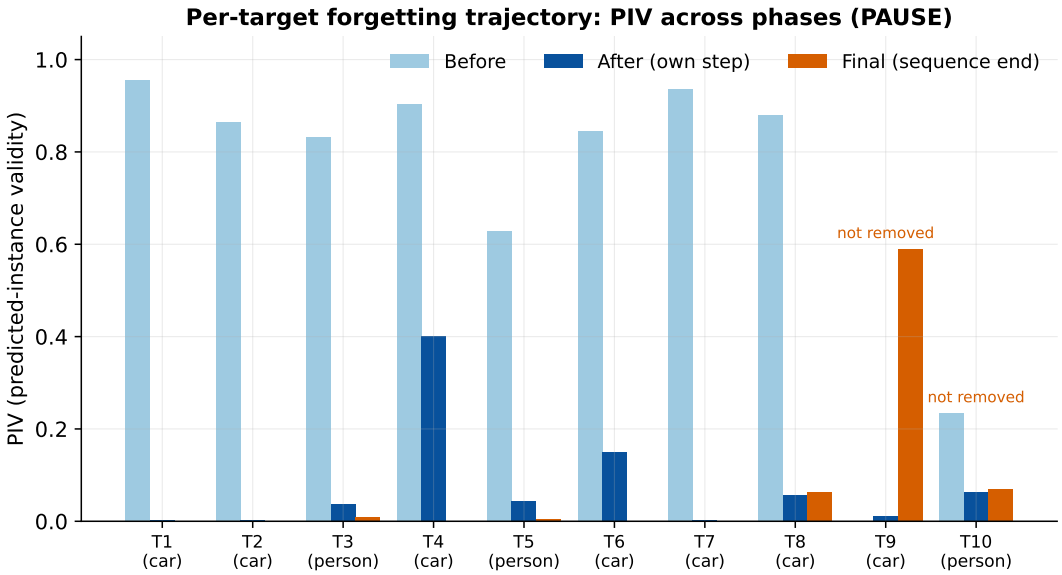


Figure 5.3: Per-target PIV before unlearning, after the target's own step, and at the sequence end (PAUSE). Eight targets are driven to near-zero PIV and stay there. T9 (already near-zero before its own step) resurges to PIV 0.59 by the end and is not removed; T10 stays at low PIV yet remains visible in the rendered output.

a person segment still renders over its region, so it fails the panoptic-removal test. Both cases are analysed in Sections 5.2.1 and 5.2.3.

5.1.5 Sequential Stability and Resurgence

Because targets are unlearned in sequence, an instance suppressed early can be disturbed by later updates. We quantify this with resurgence, the change in PIV from immediately after a target's own step to the end of the sequence (Figure 5.4). For eight targets resurgence is zero or negative – they are further suppressed, not revived – which indicates that the replay objective is actively re-suppressing earlier targets as the sequence proceeds (quantified in Section 5.3). The only material positive resurgence is T9 (+0.58); T8 and T10 show negligible positive drift (+0.007, +0.006). Mean resurgence across the ten targets is −0.003, i.e. no net revival on average.

5.2 Analysis of Design Solutions

5.2.1 Qualitative Behaviour

The quantitative picture is borne out by the rendered panoptic output. Each panel in Figures 5.5–5.9 shows the input image, the teacher's prediction before unlearning, and the student's prediction after the full sequence.

Figure 5.5 (T1, a foreground van) is a representative clean success: the target's segment, outlined in the "before" panel, is absent from the "after" panel, while the neighbouring cars and the tram are preserved with their original labels. This is the intended behaviour – the instance is removed and its surroundings are left intact. Figure 5.7 (T3) makes the same point for the person class and under a stricter test:

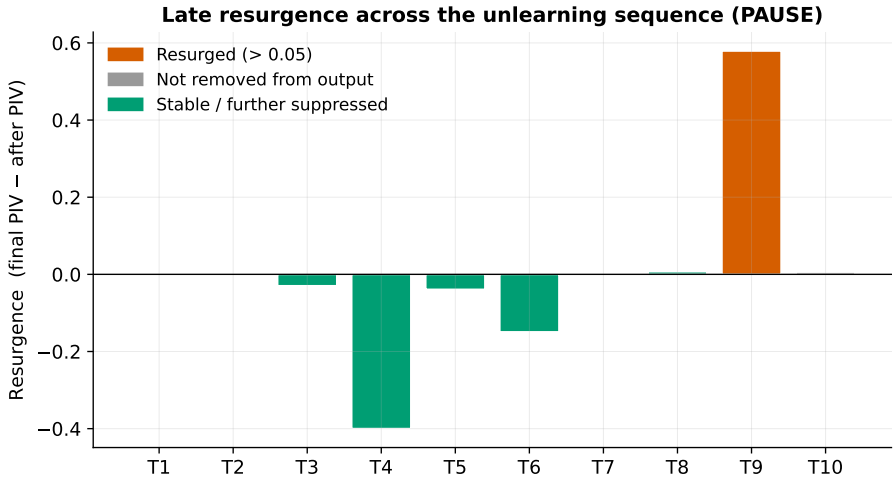


Figure 5.4: Resurgence (final PIV − after PIV) per target (PAUSE). Most targets are further suppressed over the sequence; only T9 shows substantial revival, and T10 remains visible despite low PIV.

the targeted pedestrian is removed while several other pedestrians in the same image are retained, confirming that removal is instance-level rather than class-level.

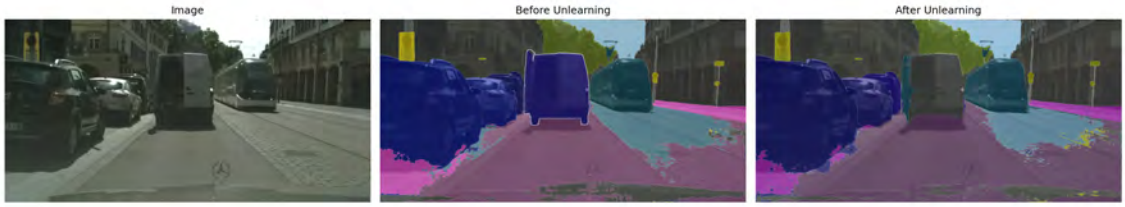


Figure 5.5: Clean removal (T1). Left: input. Centre: teacher prediction with the target outlined. Right: student prediction after the full sequence – the target van is gone while surrounding instances are preserved.

The same scene also illustrates the *sequential* character of the method, because it contains a second curated target. The image of Figure 5.5 additionally holds T4, a light-coloured car parked among the vehicles on the left, which is unlearned at a later step in the sequence than the van. Figure 5.6 shows the outcome at the end of the full sequence: the car outlined in the "before" panel is absent from the "after" panel, while the neighbouring vehicles, the tram and the surrounding stuff-classes remain intact. Two distinct instances drawn from a single image are thus each removed, and neither removal is revived as the sequence proceeds – the van (T1, processed earlier) and the car (T4, processed later) both finish the sequence further suppressed than immediately after their own steps (resurgence -0.001 and -0.400 respectively; see Table 5.4). This is direct visual evidence that suppression is instance-level and that earlier removals are preserved while subsequent targets are processed.

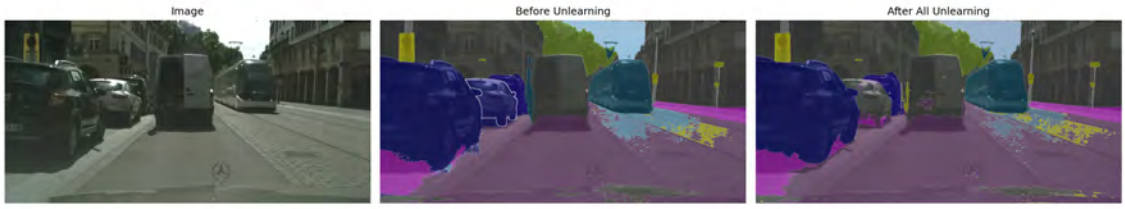


Figure 5.6: Sequential instance-level removal within a single image (T4). This is the scene of Figure 5.5; the second target, a light-coloured car (outlined, centre), is removed after the full sequence (right) at a step later than the van of Figure 5.5, while the other vehicles, the tram and the surrounding classes are preserved. The two targets from this image (T1, the van; T4, this car) are both removed and neither is revived by later steps, illustrating that sequential unlearning accumulates removals without disturbing earlier ones.

Figure 5.7: Clean instance-level removal on the person class (T3). The targeted pedestrian (outlined, centre) is removed after the full sequence (right) while the other pedestrians of the same class and the surrounding vehicles are preserved – evidence that suppression is instance-level, not class-level.

Figure 5.8 (T9) is the hard failure, and its cause is best understood at the level of individual decoder queries. In Mask2Former the mapping between object queries and instances is not fixed: the same instance can be carried by different queries in the teacher and in the drifting student. For T9 the teacher anchors the target to query 77, whereas by the time of T9's own step the student has already re-routed it to query 72; PAUSE's union-seeded re-matching detects this and suppresses both queries ($\{72, 77\}$), driving the after-step PIV to 0.011 with the segment correctly removed. The instance nonetheless resurges to PIV 0.59 by the end of the sequence. The most consistent explanation is a query-reassignment effect. T9 was already collaterally suppressed before its own step (initial PIV $\approx$ 0.0001), so the forget objective had almost no active instance on which to anchor a durable suppression; once optimisation moves past T9, the broad class-preservation objective – which protects the car class everywhere in order to retain utility – supplies a gradient that reinstates a car prediction in this confidently predicted region, and the matcher

assigns that prediction to a query *outside* the frozen forget set $\{72, 77\}$. The instance therefore reappears through a different query from the one that was suppressed, which T9's forget terms no longer cover. This is the forgetting–retention tension in its sharpest form: the same class protection that holds global PQ at 99.6% is what reintroduces an already-forgotten instance, because to a class-level objective that instance is indistinguishable from the cars that must be kept.
Figure 5.9 (T10) is the soft failure: the pedestrian's PIV is low, so the per-query criterion records success, but a person segment still renders, so panoptic-removal records failure. Together these two cases motivate using the rendered-output metric as the honest headline (Section 5.2.3).

Figure 5.8: Hard failure (T9). The target car is already absent before its own step (centre) but reappears as a confident car instance after the full sequence (right). The teacher anchors this instance to query 77 and the student to query 72 (both suppressed at T9's step); the resurgence emerges on a query outside that suppressed set, driven by the class-preservation objective – a query-reassignment effect.

Figure 5.9: Soft failure (T10). The target pedestrian reaches low PIV yet a person segment still renders (right), so it passes the per-query criterion but fails panoptic-removal.

5.2.2 The Forgetting–Retention Tension

The two failures are not independent defects; they are two faces of the same tension. Removing an instance pushes the decoder to withdraw class evidence on the target region, while preserving utility requires the decoder to keep predicting that class

wherever it legitimately appears. When a target lies in a region the teacher predicts confidently, these objectives pull in opposite directions, and the retention side can dominate – producing either revival (T9) or persistence (T10). PAUSE manages this tension well in the common case (eight of ten targets are removed with negligible utility cost), but it does not dissolve it. We regard the explicit characterisation of this trade-off, rather than a claim of perfect forgetting, as a contribution in its own right.

5.2.3 Why Panoptic-Removal is the Honest Metric

The three forgetting rates in Table 5.3 measure different things and can disagree (Figure 5.10). The after-phase and final per-query TSS examine whether a particular query still predicts the instance; the panoptic-removal rate examines whether *any* segment of the class still renders over the target after full panoptic post-processing. T10 illustrates the gap directly: its query is suppressed (per-query success) yet a segment renders (panoptic failure). Because what matters for an unlearning claim is what the deployed model actually outputs, we adopt the panoptic-removal rate (8/10) as the primary metric and report the per-query rates as secondary diagnostics. Reporting only the more favourable per-query rate would overstate the method.

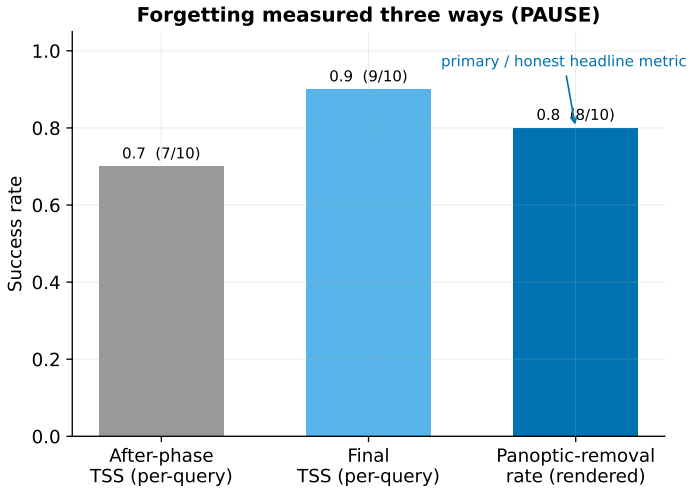


Figure 5.10: Forgetting measured three ways (PAUSE). The per-query rates (7/10, 9/10) are more generous than the rendered-output panoptic-removal rate (8/10), which is adopted as the honest headline.

5.3 Final Design Adjustments

This section justifies the final configuration of PAUSE through ablations. Each variant removes exactly one component and holds everything else constant – the same model, freezing, step budget, matching and metrics – so that any change in behaviour is attributable to that component. Table 5.5 summarises the four runs and Figures 5.11–5.12 visualise them.

Table 5.5: Ablation study. Each variant removes one component from the full method. "Rem." is the panoptic-removal rate, "fTSS" the final per-query rate, "after-PIV" the mean PIV immediately after each target's step (lower is better), "SCD" the mean same-class damage.

Variant	Rem.	fTSS	Mean after-PIV	SCD	ΔPQ
No projection	7/10	8/10	0.8718	0.0517	−0.0078
No replay	1/10	1/10	0.0787	0.0941	−0.0044
No same-class	8/10	8/10	0.1078	0.1628	−0.0050
PAUSE (full)	**8/10**	**9/10**	**0.0767**	0.1535	**−0.0026**

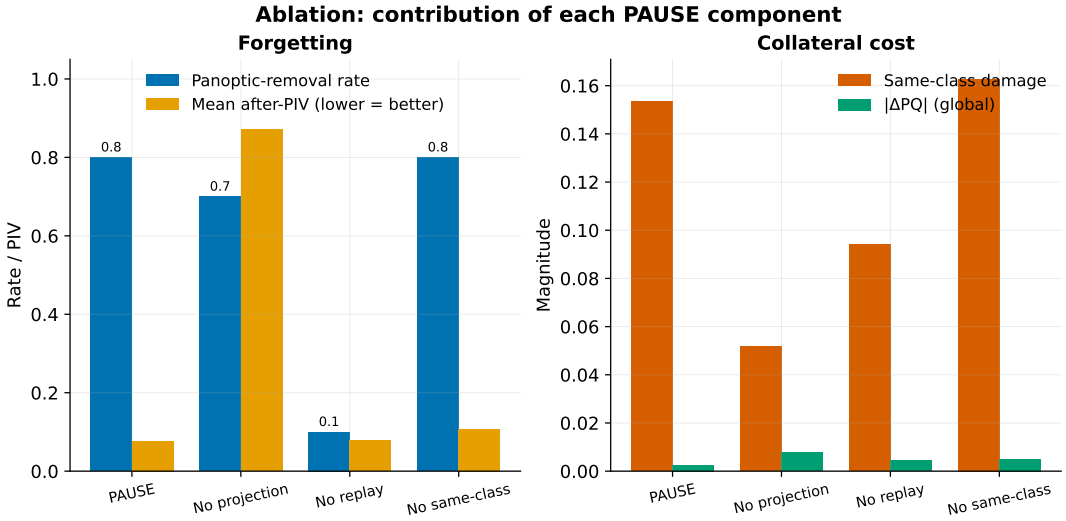


Figure 5.11: Contribution of each component. Left: forgetting (panoptic-removal rate and mean after-PIV). Right: collateral cost (same-class damage and |ΔPQ|). Removing replay collapses forgetting; removing projection collapses per-step suppression.

Replay (sequential retention). Removing replay is the most damaging single change: the panoptic-removal rate collapses from 8/10 to **1/10** and the final per-query rate to 1/10, even though immediate after-step suppression is essentially unchanged (after-PIV 0.079, close to the full method's 0.077). The interpretation is clear: without replay the model forgets each target at its own step but later updates revive almost all of them. Replay is therefore the component responsible for sequential stability, and its absence turns the method into a single-shot procedure that does not survive a stream of requests.

Asymmetric projection. Removing the asymmetric forget-priority projection leaves the final removal rate broadly intact (7/10) but destroys clean per-step forgetting: mean after-PIV rises from 0.077 to **0.872**, meaning that at the moment each target is processed it is barely suppressed. That the final rate nonetheless recovers to 7/10 indicates that downstream updates and replay eventually compensate, but the method becomes far less predictable per step and the global utility cost roughly triples (ΔPQ −0.0078 vs −0.0026). Projection is what allows the forget gradient to act decisively at each step without the retain gradient cancelling it.

Same-class / thing-class retention. Removing same-class retention keeps the global numbers superficially similar (removal 8/10, ΔPQ −0.0050), which is exactly why a global figure can mislead: the cost reappears at the class level. Figure 5.12 shows that without this term the person class loses 4.1 pp of PQ (against 1.65 pp for the full method), while car is largely unaffected – consistent with the term's role in protecting the unlearned classes from broad suppression. The same-class objective trades a little additional same-class damage (SCD 0.163 vs 0.154) for substantially lower class-level utility loss on the more diffuse class.

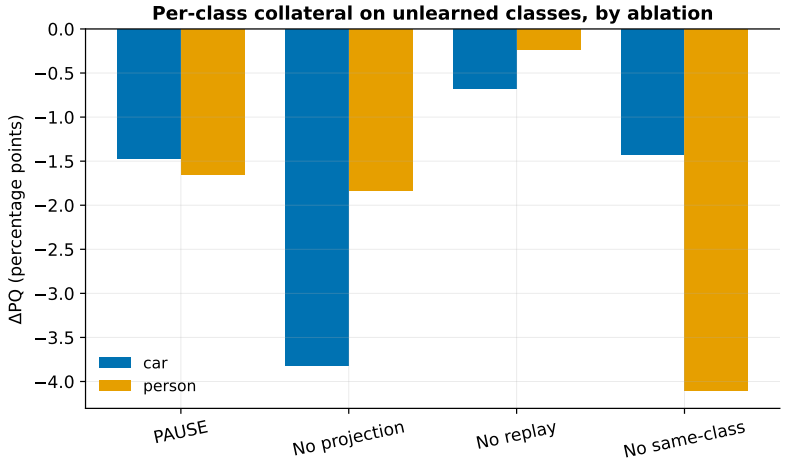


Figure 5.12: Per-class ΔPQ on the unlearned classes across ablations. Removing same-class retention roughly doubles the person-class cost, a difference the global ΔPQ hides.

Final configuration. The ablations support retaining all four analysed components. Replay is indispensable for sequential stability; asymmetric projection is required for decisive per-step forgetting and lower utility cost; same-class retention protects the unlearned classes at the class level; localised erasure (whose removal produced large class-level damage in earlier development and is reflected in the localisation seen in Section 5.1.3) keeps the cost confined. The final PAUSE configuration is therefore the full method as evaluated in Section 5.1.

5.4 Statistical Analysis

We apply non-parametric tests appropriate to the ten paired per-target measurements; with $n = 10$ these are indicative rather than definitive, and we report them as such. Table 5.6 collects the results.

5.4.1 Significance of Instance Suppression

The reduction in PIV from before to after unlearning is statistically significant under a Wilcoxon signed-rank test ($W = 1.0$, $p = 0.0039$), with a large effect size (Cliff's $\delta = 0.78$) and a median reduction of 0.74. The reduction persists to the end of the sequence ($W = 2.0$, $p = 0.0059$, median reduction 0.83). In descriptive terms, mean PIV falls from 0.708 ± 0.328 to 0.077 ± 0.122 after each step and to 0.074 ± 0.184

Table 5.6: Statistical analysis of the per-target measurements ($n = 10$, PAUSE). Wilcoxon signed-rank tests assess PIV reduction; the correlation tests assess whether the initial PIV predicts later resurgence.

Test	Statistic	p	Interpretation
Wilcoxon, PIV before vs after	$W = 1.0$	0.0039	significant reduction
Wilcoxon, PIV before vs final	$W = 2.0$	0.0059	significant reduction
Cliff's δ (before/after)	0.78	–	large effect
Pearson r (init. PIV, resurg.)	-0.72	0.019	suggestive‡
Spearman ρ (init. PIV, resurg.)	-0.21	0.567	not significant

at the end. The suppression effect is thus both large and consistent across targets, the elevated final standard deviation being attributable almost entirely to T9.

5.4.2 Predictors of Resurgence

We tested whether a target's initial PIV predicts its later resurgence (Figure 5.13). The Pearson correlation is moderate-to-strong and significant ($r = -0.72$, $p = 0.019$), suggesting that targets with lower initial PIV resurge more, which matches the T9 mechanism. However, the rank-based Spearman correlation is weak and not significant ($\rho = -0.21$, $p = 0.567$). The discrepancy is itself informative: the Pearson value is driven largely by the single extreme case (T9, initial PIV ≈ 0, resurgence $+0.58$), so we report the association as *suggestive but not robust* at this sample size. The honest reading is that the already-suppressed-then-resurged pattern is a plausible and observed failure mechanism, not a statistically established law; confirming it would require more targets spanning the low-initial-PIV regime.

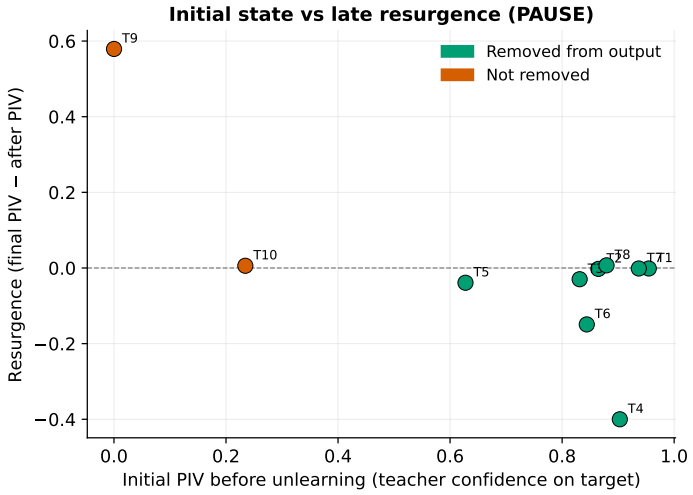


Figure 5.13: Initial PIV against resurgence (PAUSE), coloured by whether the target was removed. The single low-initial-PIV target (T9) is the one large resurgence, illustrating the failure mechanism while also dominating the parametric correlation.

5.4.3 Utility–Forgetting Pareto Position

Finally, we summarise the methods on the two axes that matter jointly – forgetting and retained utility – in Section 5.5, where Figure 5.14 places PAUSE at the favourable corner of the trade-off. We use relative PQ retention there to neutralise the evaluator difference described below.

5.5 Comparisons and Relationships

5.5.1 Comparison Protocol and Fairness

PAUSE is compared with three unlearning baselines implemented faithfully and run through the same harness: NegGrad (gradient ascent on the forget loss), NegGrad+ (ascent on forget plus a distilled retain term), and SCRUB [42], a distillation-based MAX/MIN method. All baselines share PAUSE's freezing, early-stopping, step budget, matching and metrics; only the per-step loss differs. Baselines use teacher-only matching, their published form, while student re-matching is part of PAUSE's contribution and is not retrofitted into them – stated here so the comparison is not silently biased in PAUSE's favour.
All experiments, including PAUSE, the ablations, and the three baselines, were evaluated using the same official panopticapi evaluator. The teacher model obtains a PQ of 0.6453 under this evaluator. Therefore, both absolute post-unlearning PQ and relative retention metrics are directly comparable across methods. We report ΔPQ and relative PQ retention (after / before) to quantify the extent of performance preservation after unlearning, while using the consistently evaluated absolute PQ values for direct method-level comparison.

5.5.2 Unified Cross-Method Comparison

Table 5.7 reports all four methods. The headline relationship is a clear trade-off: the three baselines all achieve 10/10 panoptic-removal, but they do so by degrading the model, whereas PAUSE removes 8/10 while retaining 99.6% of utility.

Table 5.7: PAUSE vs. baselines on the ten targets. "Rem." is panoptic-removal; "aTSS"/"fTSS" are after-/final per-query rates (from the per-target logs); "Ret." is relative PQ retention (after/before); "SCD" is mean same-class damage. Absolute PQ is not directly comparable across evaluators (see Section 5.5.1); compare ΔPQ and Ret.

Method	PQ_{bef}	PQ_{aft}	ΔPQ	Ret. (%)	Rem.	aTSS	fTSS	SCD
NegGrad [42]	0.6453	0.0029	−0.6424	0.45	10/10	9/10	10/10	0.582
NegGrad+ [42]	0.6453	0.3171	−0.3282	49.14	10/10	9/10	10/10	0.572
SCRUB [42]	0.6453	0.4703	−0.1750	72.88	10/10	9/10	10/10	0.446
PAUSE	0.6453	0.6427	**−0.0026**	**99.6**	8/10	7/10	9/10	**0.154**

Comparison with NegGrad. Pure gradient ascent removes every target (10/10) but is catastrophic for utility: global PQ falls from 0.6453 to 0.0029 (ΔPQ −0.6424; 0.45% retained), and both unlearned classes are reduced to zero PQ (Table 5.8).

NegGrad "forgets" by destroying the model; it is an upper bound on aggression and a lower bound on usefulness.

Comparison with NegGrad+. Adding a distilled retain term rescues some utility – PQ after is 0.3171 (49.14% retained) – but the damage remains severe, and the unlearned classes retain only 40.95% (car) and 22.15% (person) of their PQ. The retain term helps but cannot, on its own, localise the forgetting.

Comparison with SCRUB. SCRUB [42] is the strongest baseline: its MAX/MIN distillation retains 72.88% of PQ while removing all ten targets. Even so, its same-class damage (0.446) is roughly three times PAUSE's (0.154), and its unlearned classes retain under half their PQ (car 43.89%, person 44.11%). SCRUB forgets thoroughly but pays a class-level utility price an order of magnitude larger than PAUSE.

Table 5.8: Retention of PQ on the two unlearned classes per method (after/before). PAUSE preserves the unlearned classes almost entirely; the baselines remove the target instances by degrading the whole class.

	car		person	
Method	Δ (pp)	Ret. (%)	Δ (pp)	Ret. (%)
NegGrad [42]	−64.88	0.0	−52.41	0.0
NegGrad+ [42]	−40.62	40.95	−40.80	22.15
SCRUB [42]	−38.60	43.89	−29.29	44.11
PAUSE	**−1.47**	**97.9**	**−1.65**	**96.9**

The trade-off, visualised. Figures 5.14–5.16 make the relationship explicit. On the utility–forgetting plane (Figure 5.14) PAUSE sits alone in the desirable corner: high removal *and* near-full retention. The baselines occupy the high-removal/low-retention edge, NegGrad at the extreme. Figure 5.15 shows the order-of-magnitude gap in ΔPQ, and Figure 5.16 shows that the baselines achieve removal by gutting the unlearned class, whereas PAUSE leaves it almost intact. The conclusion is not that PAUSE forgets more – by the raw removal count the baselines match or exceed it – but that PAUSE is the only method that forgets substantially *without* sacrificing the utility the model exists to provide.

5.6 Discussions

5.6.1 Summary of Results

PAUSE removes the majority of targeted instances (8/10 by the rendered-output metric) while retaining 99.6% of global panoptic quality and 97–98% of PQ on the unlearned classes themselves. The utility cost is small, localised to a few thing-classes, and absent from the frozen stuff-classes. The ablations show that each analysed component earns its place: replay provides sequential stability, projection provides decisive per-step forgetting, same-class retention protects the unlearned classes at the class level, and localised erasure confines the cost. Against three

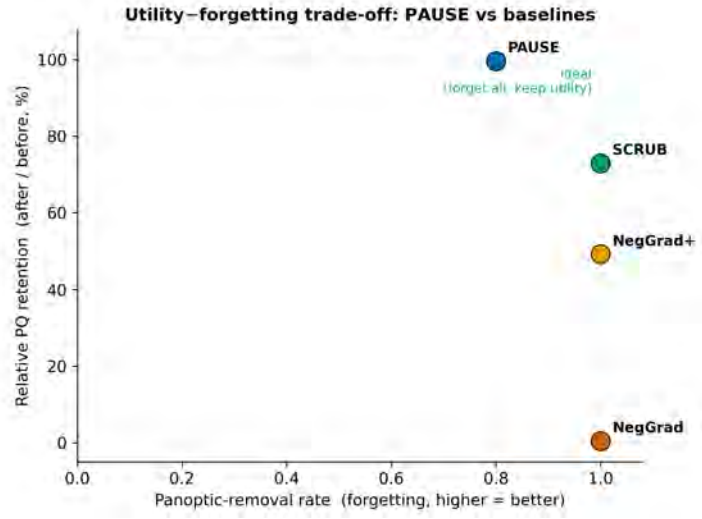


Figure 5.14: Utility–forgetting trade-off. The vertical axis is relative PQ retention (after/before), chosen so the official-vs-internal evaluator offset cancels; the horizontal axis is panoptic-removal. PAUSE occupies the favourable top-right region.

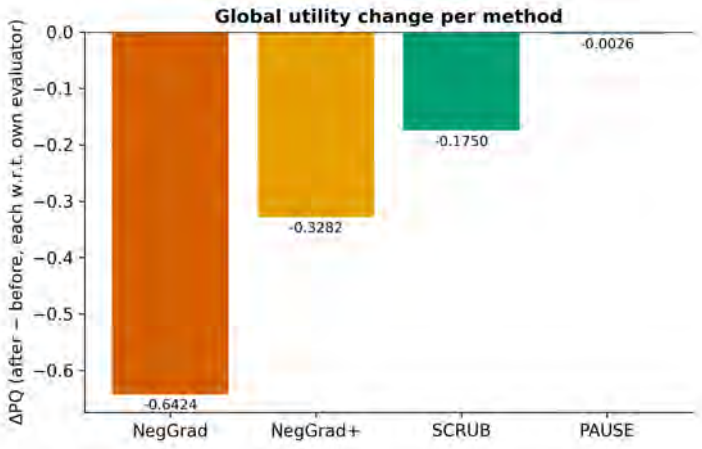


Figure 5.15: Global ΔPQ per method. PAUSE's utility cost is two to three orders of magnitude smaller than the baselines'. Compare the magnitude of the drop, not absolute PQ, because of the evaluator difference.

baselines, PAUSE is the only method that combines substantial forgetting with preserved utility.

5.6.2 Interpretation and Implications

The comparison clarifies what is hard about this problem. Forgetting an instance is easy if one is willing to break the model – NegGrad demonstrates this emphatically. The difficulty lies in forgetting an instance in a shared-representation dense predictor *without* collateral damage, and it is on this axis that PAUSE's design choices pay off. The asymmetry of the unlearning objective – forgetting must win locally while retention must hold globally – is what motivates an asymmetric, forget-priority projection rather than a symmetric multi-task scheme, and the ablation of projection supports that reasoning. More broadly, the results suggest that instance unlearning

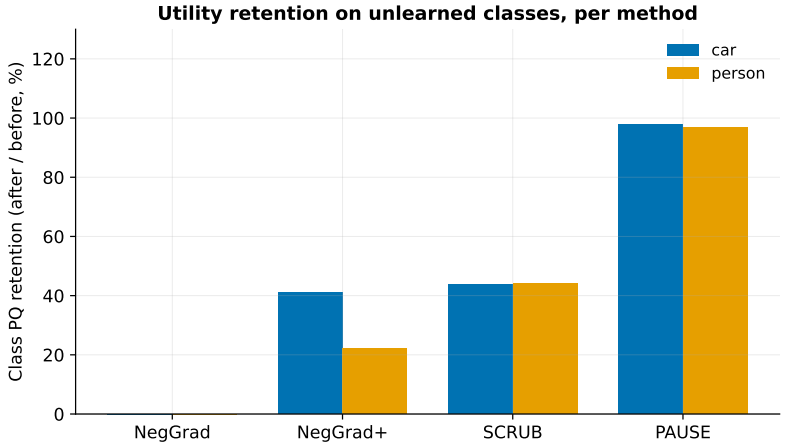


Figure 5.16: PQ retention on the unlearned classes (car, person) per method. PAUSE retains 97–98%; the baselines retain half or less, and NegGrad none.

in panoptic models is best framed not as a binary capability but as a position on a utility–forgetting frontier, and that the right question for a method is where on that frontier it sits.

5.6.3 Limitations

The results carry honest limitations. First, forgetting is not complete: two of ten targets resist removal, and the hard failure (T9) shows that a target which is already collaterally suppressed and lies in a confidently predicted region can be reintroduced by the class-preservation objective – the forgetting–retention tension is real and unresolved in such cases. Second, forgetting leans on class suppression even with no-object promotion, and the per-query and rendered-output metrics can disagree (T10), which is why we report the stricter panoptic-removal rate. Third, some targets are not cleanly forgotten at their own step and only settle by the final evaluation, relying in part on downstream reinforcement; the no-projection ablation shows how pronounced this can become. Fourth, generality is unvalidated beyond Cityscapes and Mask2Former-Swin-B: the dynamic weighting's two hyperparameters are dataset-dependent and would need re-tuning elsewhere, and an evaluated null-space projection was rejected because the panoptic decoder is near full-rank, which is what motivated the forget-priority projection in the first place. Finally, the statistical analysis rests on ten targets; the resurgence predictor is suggestive rather than established, and the cross-method utility comparison is qualified by the evaluator difference noted in Section 5.5.1.

5.6.4 Relation to Prior Work

The baselines situate PAUSE within the unlearning literature. Gradient-ascent methods (NegGrad, NegGrad+) and distillation-based methods (SCRUB [42]) were developed largely for classification, where forgetting a sample need not damage a densely shared representation; transplanted to panoptic segmentation they forget effectively but at a utility cost that the present results quantify. PAUSE's projection draws on gradient-surgery ideas from multi-task learning – PCGrad [26] and

gradient-magnitude balancing in the spirit of GradNorm [9] – but adapts them to the asymmetric structure of unlearning rather than the symmetric structure of multi-task optimisation. The honest-removal metric connects to the membership and auditing perspective in the unlearning literature [27] by insisting that a forgetting claim be evaluated on what the model actually outputs. In short, the contribution is not a new optimiser but a problem formulation and a component design that hold the utility–forgetting trade-off in a regime the prior methods do not reach, together with an honest account of where it still fails.

Chapter 6

Conclusion

6.1 Summary of Findings

In this thesis, we introduced PAUSE, a post-hoc sequential instance unlearning framework for panoptic segmentation models. The framework was designed to selectively erase individual object instances without full retraining while preserving overall segmentation utility. To achieve this, we combined localized erase objectives, asymmetric gradient projection, dynamic loss balancing, and replay-guided sequential repair within a teacher–student setting. Experiments on Mask2Former using the Cityscapes dataset showed that PAUSE can effectively suppress targeted instances under sequential deletion while maintaining stable global performance. Compared to existing baselines, the framework achieved stronger forgetting quality with lower collateral degradation. The results also highlighted the inherent tension between precise instance removal and same-class retention in dense prediction systems.

6.2 Contributions to the Field

This work contributes to the field by formally studying sequential instance-level unlearning in panoptic segmentation, a largely unexplored dense prediction setting. We proposed PAUSE, a practical framework tailored for query-based architectures, and introduced an asymmetric forget-priority gradient projection strategy for resolving forget–retain conflicts during optimization. In addition, we incorporated replay-guided repair and dynamic loss balancing to improve sequential stability and reduce resurgence. Finally, we established an evaluation framework combining forgetting quality, utility preservation, and per-class collateral analysis for dense prediction unlearning systems.

6.3 Recommendations for Future Work

Future work should evaluate PAUSE on larger datasets and alternative segmentation architectures to examine its generalizability. Extending the framework toward stuff-region unlearning, video segmentation, and multimodal dense prediction systems also remains an important direction. Further research is needed to improve query-tracking stability and reduce resurgence during long sequential deletion streams.

Finally, integrating stronger privacy verification mechanisms and more computationally efficient update strategies may help make dense-prediction unlearning more practical for real-world deployment.

Bibliography

[1] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440, 2014. [Online]. Available: https://www.cv-foundation.org/openaccess/content_cvpr_2015/papers/Long_Fully_Convolutional_Networks_2015_CVPR_paper.pdf

[2] Y. Cao and J. Yang, "Towards making systems forget with machine unlearning," *2015 IEEE Symposium on Security and Privacy*, pp. 463–480, 2015. [Online]. Available: https://ieeexplore.ieee.org/document/7163042

[3] G. E. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *ArXiv*, vol. abs/1503.02531, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:7200347

[4] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *ArXiv*, vol. abs/1505.04597, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:3719281

[5] M. Cordts et al., "The cityscapes dataset for semantic urban scene understanding," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3213–3223, 2016. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2016/papers/Cordts_The_Cityscapes_Dataset_CVPR_2016_paper.pdf

[6] J. Kirkpatrick et al., "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences*, vol. 114, pp. 3521–3526, 2016. [Online]. Available: https://www.pnas.org/doi/10.1073/pnas.1611835114

[7] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, pp. 2935–2947, 2016. [Online]. Available: https://ieeexplore.ieee.org/document/8107520

[8] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 3–18, 2016. [Online]. Available: https://ieeexplore.ieee.org/document/7958568

[9] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, "Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks," *ArXiv*, vol. abs/1711.02257, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:4703661

[10] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask r-cnn," 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:54465873

[11] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7482–7491, 2017. [Online]. Available: https://ieeexplore.ieee.org/document/8578879

[12] P. W. Koh and P. Liang, "Understanding black-box predictions via influence functions," in *International Conference on Machine Learning*, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:13193974

[13] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2017. [Online]. Available: https://openreview.net/pdf?id=Bkg6RiCqY7

[14] P. Voigt and A. von dem Bussche, "The eu general data protection regulation (gdpr)," 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:169453414

[15] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *European Conference on Computer Vision*, 2018. [Online]. Available: https://openaccess.thecvf.com/content_ECCV_2018/papers/Liang-Chieh_Chen_Encoder-Decoder_with_Atrous_ECCV_2018_paper.pdf

[16] A. Kirillov, K. He, R. B. Girshick, C. Rother, and P. Dollár, "Panoptic segmentation," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9396–9405, 2018. [Online]. Available: https://ieeexplore.ieee.org/document/8953237

[17] G. Zeng, Y. Chen, B. Cui, and S. Yu, "Continual learning of context-dependent processing in neural networks," *Nature Machine Intelligence*, vol. 1, pp. 364–372, 2018. [Online]. Available: https://www.nature.com/articles/s42256-019-0080-x

[18] L. Bourtoule et al., "Machine unlearning," *2021 IEEE Symposium on Security and Privacy (SP)*, pp. 141–159, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/9519428

[19] A. Golatkar, A. Achille, and S. Soatto, "Eternal sunshine of the spotless net: Selective forgetting in deep networks," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9301–9309, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/9157084

[20] C. Guo, T. Goldstein, A. Y. Hannun, and L. van der Maaten, "Certified data removal from machine learning models," in *International Conference on Machine Learning*, 2019. [Online]. Available: https://proceedings.mlr.press/v119/guo20c/guo20c.pdf

[21] Y. He, S. Rahimian, B. Schiele, and M. Fritz, "Segmentations-leak: Membership inference attacks and defenses in semantic image segmentation," in *European Conference on Computer Vision*, 2019. [Online]. Available: https://www.ecva.net/papers/eccv_2020/papers_ECCV/papers/123680511.pdf

[22] R. Schwartz, J. Dodge, N. A. Smith, and O. Etzioni, "Green ai," 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:198229505

[23] E. Strubell, A. Ganesh, and A. McCallum, "Energy and policy considerations for deep learning in nlp," *ArXiv*, vol. abs/1906.02243, 2019. [Online]. Available: https://doi.org/10.1609/aaai.v34i09.7123

[24] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," *ArXiv*, vol. abs/2005.12872, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:218889832

[25] F. Cermelli, M. Mancini, S. R. Bulò, E. Ricci, and B. Caputo, "Modeling the background for incremental learning in semantic segmentation," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9230–9239, 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9157089

[26] T. Yu, S. Kumar, A. Gupta, S. Levine, K. Hausman, and C. Finn, "Gradient surgery for multi-task learning," *ArXiv*, vol. abs/2001.06782, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:210839011

[27] N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramèr, "Membership inference attacks from first principles," *2022 IEEE Symposium on Security and Privacy (SP)*, pp. 1897–1914, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9833649

[28] M. Chen, Z. Zhang, T. Wang, M. Backes, M. Humbert, and Y. Zhang, "Graph unlearning," *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, 2021. [Online]. Available: https://dl.acm.org/doi/10.1145/3548606.3559352

[29] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1280–1289, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9878483

[30] B. Cheng, A. G. Schwing, and A. Kirillov, "Per-pixel classification is not all you need for semantic segmentation," in *Neural Information Processing Systems*, 2021. [Online]. Available: https://papers.neurips.cc/paper_files/paper/2021/file/950a4152c2b4aa3ad78bdd6b366cc179-Paper.pdf

[31] B. Liu, X. Liu, X. Jin, P. Stone, and Q. Liu, "Conflict-averse gradient descent for multi-task learning," *ArXiv*, vol. abs/2110.14048, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:239998731

[32] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9992–10 002, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9710580

[33] G. Saha, I. Garg, and K. Roy, "Gradient projection memory for continual learning," *ArXiv*, vol. abs/2103.09762, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:232257725

[34] A. K. Tarun, V. S. Chundawat, M. Mandal, and M. Kankanhalli, "Fast yet effective machine unlearning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, pp. 13 046–13 055, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/10113700

[35] A. Thudi, H. Jia, I. Shumailov, and N. Papernot, "On the necessity of auditable algorithmic definitions for machine unlearning," in *USENIX Security Symposium*, 2021. [Online]. Available: https://www.usenix.org/system/files/sec22-thudi.pdf

[36] S. Wang, X. Li, J. Sun, and Z. Xu, "Training networks in null space of feature covariance for continual learning," *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 184–193, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9578171

[37] V. S. Chundawat, A. K. Tarun, M. Mandal, and M. Kankanhalli, "Can bad teaching induce forgetting? unlearning in deep networks using an incompetent teacher," *ArXiv*, vol. abs/2205.08096, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:248834527

[38] J. Jain, J. Li, M. C. Chiu, A. Hassani, N. Orlov, and H. Shi, "Oneformer: One transformer to rule universal image segmentation," *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2989–2998, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/10203147

[39] T. T. Nguyen, T. T. Huynh, P.-L. Nguyen, A. W.-C. Liew, H. Yin, and Q. V. H. Nguyen, "A survey of machine unlearning," *ACM Transactions on Intelligent Systems and Technology*, vol. 16, pp. 1–46, 2022. [Online]. Available: https://dl.acm.org/doi/10.1145/3749987

[40] C. Fan, J. Liu, Y. Zhang, D. Wei, E. Wong, and S. Liu, "Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation," *ArXiv*, vol. abs/2310.12508, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:264305818

[41] J. Jia et al., "Model sparsity can simplify machine unlearning," *Advances in Neural Information Processing Systems 36*, 2023. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2023/file/a204aa68ab4e970e1ceccfb5b5cdc5e4-Paper-Conference.pdf

[42] M. Kurmanji, P. Triantafillou, and E. Triantafillou, "Towards unbounded machine unlearning," *ArXiv*, vol. abs/2302.09880, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:257038445

[43] Z. Cai, Y. Tan, and M. S. Asif, "Targeted unlearning with single layer unlearning gradient," in *International Conference on Machine Learning*, 2024. [Online]. Available: https://openreview.net/pdf?id=6Ofb0cGXb5

[44] Y. Zhou, D. Zheng, Q. Mo, R. Lu, K.-Y. Lin, and W.-S. Zheng, "Decoupled distillation to erase: A general unlearning method for any class-centric tasks," *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20 350–20 359, 2025. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2025/papers/Zhou_Decoupled_Distillation_to_Erase_A_General_Unlearning_Method_for_Any_CVPR_2025_paper.pdf