Rethinking Post-Hoc Calibration in Semantic Segmentation

Anonymous authors
Paper under double-blind review

Abstract

Reliable confidence estimates are essential in semantic segmentation, especially in safety-critical settings where overconfident errors can mislead downstream decisions. Yet modern segmentation models often remain miscalibrated. Post-hoc calibration offers a practical way to correct confidence estimates without retraining the segmentation model, but its use in dense prediction raises structural issues that are often overlooked. We study two such issues. First, adding a constant to all logits leaves the softmax probabilities unchanged, but several standard calibrators can still depend on this arbitrary offset. As a result, two logit representations encoding the same predictive distribution may yield different calibrated probabilities. We define translation-invariant (TI) calibrators as those whose outputs are unchanged under such shifts, characterize which common calibrators satisfy this property, and construct TI counterparts of shift-sensitive calibrators to isolate the effect of removing representation dependence. Second, post-hoc calibration is typically fitted by minimizing a likelihood-based objective, whereas segmentation models are trained with task-specific metrics such as Dice. This mismatch can cause calibration to alter class orderings and degrade the deployed segmentation map. We study decision-preserving calibration under argmax- and order-preservation constraints. Since enforcing these constraints collapses affine softmax calibrators to temperature scaling, we introduce class-conditional affine calibrators that can be made argmax- or order-preserving while retaining greater expressivity, allowing us to quantify the calibration–segmentation trade-off induced by decision preservation. Across natural-image and medical segmentation benchmarks, and under corruption-based covariate shift, matched comparisons show that TI variants generally improve calibration metrics, while decision-preserving variants prevent segmentation degradation and retain strong calibration performance. These results provide practical design principles for well-defined post-hoc calibration pipelines in semantic segmentation.

1 Introduction

Semantic segmentation models can achieve strong task performance while still producing unreliable confidence estimates. In dense prediction settings, calibration is particularly important because voxel- or pixel-wise confidence scores are used for downstream decisions, quality control, uncertainty-aware post-processing, and failure detection (Mehrtash et al., 2020; Kahl et al., 2024; Zenk et al., 2025). However, high segmentation accuracy does not guarantee reliable predictive probabilities: segmentation models may be miscalibrated, exhibiting over- or underconfidence, especially near object boundaries, ambiguous structures, or distribution shifts (Rahaman & Thiery, 2021; Zeevi et al., 2025). Such errors can undermine the practical utility of confidence estimates even when the predicted segmentation masks are accurate.

Post-hoc calibration offers a practical way to correct confidence estimates after training, without retraining or modifying the underlying segmentation model. Train-time calibration methods, such as label smoothing (Müller et al., 2019), differentiable calibration losses (Kumar et al., 2018), and focal-loss-based calibration (Mukhoti et al., 2020), are complementary. We focus on post-hoc calibration because it can be applied to already deployed segmentation models without retraining, can be updated under changing deployment conditions, including distribution shifts, and allows confidence correction to be studied separately from segmentation-model training. Standard post-hoc methods typically learn a global transformation in

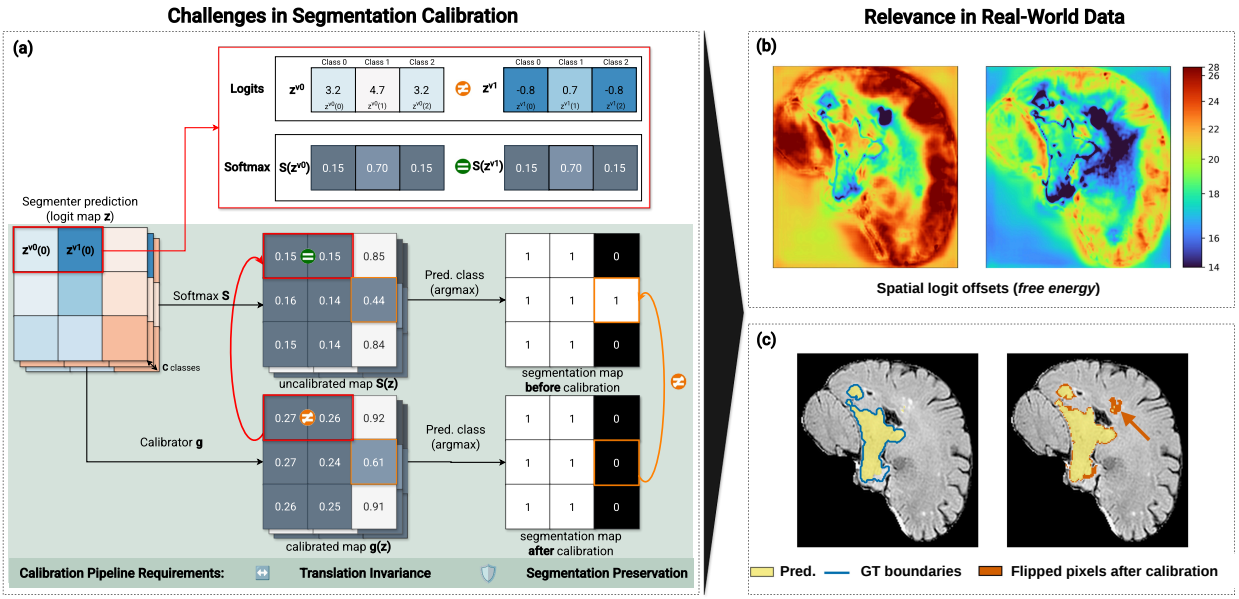

Figure 1: **Challenges in Segmentation Calibration: Evidence from Real-World Data. (1) Translation invariance.** As illustrated in panel (a), logit embeddings that differ only by a constant offset induce identical class probabilities under softmax, yet a calibrator may produce different outputs for these equivalent representations. Panel (b) shows, for two segmenters, spatial maps of the logit free energy, highlighting that this additive logit component varies across locations in real segmentation outputs. This motivates calibration pipelines that are invariant to such offsets. **(2) Segmentation preservation.** As shown in panel (a), calibration may change the argmax, flipping labels and altering the segmentation. Panel (c) shows a real example where such flips perturb brain tumor boundaries. A desirable post-hoc calibrator therefore preserves the deployed segmentation while improving confidence estimates.

logit or probability space on a held-out calibration set, often by minimizing cross-entropy (Guo et al., 2017; Kull et al., 2019; Zhang et al., 2020). These methods are largely inherited from image-level classification, where each sample is associated with a single prediction. In semantic segmentation, however, calibration is applied densely across many spatial locations, and the interaction between logit-space transformations, spatially structured predictions, and task-specific segmentation metrics has not been fully characterized.

A fundamental but often overlooked point is that logits are non-identifiable: adding a constant to all class logits at a given spatial location leaves the softmax probabilities unchanged. Thus, a predictive distribution corresponds not to a unique logit vector, but to an equivalence class of logits that differ by additive shifts, sometimes interpreted as a "gauge" degree of freedom. Consequently, two logit vectors that induce the same predictive distribution can yield different calibrated outputs if the calibration function operates directly in logit space and is sensitive to such shifts. This dependence on an arbitrary logit representative introduces representation sensitivity, potentially leading to inconsistent or unstable calibration behavior.

While logit non-identifiability also holds in image-level classification, its practical impact may be amplified in semantic segmentation. Calibration is applied across many voxels or pixels, and additive offsets may vary across spatial locations, effectively introducing a field of local gauge degrees of freedom. This spatial structure increases the dimensionality of the ambiguity and may make its practical consequences more pronounced for dense prediction tasks. Fig. 1(a, b) illustrates this issue through shifted-logit examples and empirical free-energy maps, showing that such offsets occur in practice and vary spatially.

A second structural issue arises from the mismatch between training and calibration objectives. Segmentation models are typically trained with task-specific losses such as the Dice loss or its variants (Sudre et al., 2017; Isensee et al., 2021), whereas post-hoc calibration usually minimizes cross-entropy (Guo et al., 2017; Kull et al., 2019). Because optimizing cross-entropy can alter class orderings, calibration may change the

predicted segmentation mask and potentially reduce the DSC. In our experiments, such calibration-induced decision changes degrade segmentation performance (Fig. 1(c)), motivating calibrators that correct confidence estimates without altering the original decision map.

Despite the widespread use of post-hoc calibration in segmentation, the interplay between logit non-identifiability, spatially dense calibration, and calibration-induced decision changes has not been systematically analyzed. We address this gap through two complementary structural constraints on post-hoc calibration:

- We formalize *translation invariance* as a condition for representation-consistent calibration, characterize which common calibrators satisfy it, and construct matched translation-invariant (TI) counterparts of shift-sensitive calibrators to isolate the effect of removing representation dependence.
- We study decision-preserving calibration under argmax- and order-preservation constraints. Since enforcing these constraints collapses affine softmax calibrators to temperature scaling, we introduce class-conditional affine calibrators that can be made argmax- or order-preserving while retaining greater expressivity, allowing us to quantify the calibration–segmentation trade-off induced by decision preservation.
- Across natural-image and medical segmentation benchmarks, and under corruption-based covariate shift, paired comparisons show that TI variants improve calibration metrics in most matched comparisons. Decision-preserving variants prevent calibration-induced DSC drops by construction, while retaining strong calibration performance.

Overall, our goal is not to identify a universally optimal calibration pipeline, since performance also depends on calibrator expressivity and dataset-specific characteristics. Instead, we clarify which structural properties make post-hoc calibration well-defined for dense prediction. Translation invariance ensures that calibration does not depend on an arbitrary logit representative, while argmax or order preservation prevents calibration from altering the deployed segmentation map.

2 Related Work

Post-hoc calibration of neural networks. Post-hoc calibration aims to align predicted confidence with empirical accuracy without modifying the trained model. Pioneering work focused on binary or small-scale classification, including Platt scaling (Platt, 1999), histogram binning (Zadrozny & Elkan, 2001), isotonic regression (Zadrozny & Elkan, 2002), and Bayesian binning into quantiles (Pakdaman Naeini et al., 2015). For modern multiclass neural networks, commonly used methods include temperature scaling (TS), vector scaling (VS), matrix scaling (MS), and Dirichlet calibration (DC) (Guo et al., 2017; Kull et al., 2019), as well as order-preserving and Gaussian process-based extensions (Rahimi et al., 2020; Milios et al., 2018; Wenger et al., 2020). A fundamental but often overlooked issue in calibration is that logits are not uniquely defined: adding a constant to all logits leaves softmax probabilities unchanged. The implications of this invariance for post-hoc calibration have received little attention. In practice, calibration is often performed directly in logit space using affine transformations such as vector or matrix scaling, which may depend on the particular logit representative chosen. As a result, calibrated predictions may vary across equivalent logit embeddings. Although one could argue that logit shifts may encode useful information in certain contexts, for example in energy-based formulations for out-of-distribution detection (Liu et al., 2020; Kim & Kwon, 2024), post-hoc calibration of predictive probabilities should not depend on arbitrary representational degrees of freedom unless explicitly modeled. In contrast to prior work, we formalize translation invariance as a necessary condition for representation-consistent calibration and characterize which calibration pipelines satisfy it, or how this property can be enforced.

Post-hoc calibration in semantic segmentation. In semantic segmentation, reliable voxel- or pixel-wise confidence estimates are important for downstream decisions, quality control, failure detection, and uncertainty-aware post-processing (Mehrtash et al., 2020; Kahl et al., 2024; Zenk et al., 2025). Calibration methods developed for classification are commonly transferred to dense prediction by treating pixels or voxels

as independent calibration samples and applying a global mapping (Rousseau et al., 2025). This global approach assumes uniform calibration across the image, although calibration errors often vary spatially, especially near object boundaries and ambiguous structures (Zeevi et al., 2025; Liang et al., 2025). To address spatial heterogeneity, adaptive methods such as Local Temperature Scaling (LTS) (Ding et al., 2021) learn spatially varying temperature maps, where the temperature at each location is predicted from the network's outputs, such as logits. Selective scaling mechanisms have also been proposed to enable more flexible calibration in semantic segmentation by focusing adjustment on misclassified pixels (Wang et al., 2023). A second challenge is the mismatch between calibration and task-specific segmentation objectives. Segmentation models are often trained or evaluated with overlap-based criteria such as Dice (Sudre et al., 2017; Isensee et al., 2021), whereas post-hoc calibration is typically fitted by minimizing cross-entropy. Consequently, a calibrator can improve the calibration loss while changing class orderings, thereby modifying the predicted segmentation map and causing measurable DSC drops (Rousseau et al., 2025). This motivates decision-preserving calibration, in which calibrated probabilities preserve either the original argmax or the complete class ordering. In segmentation, methods with this property commonly rely on temperature-scaling variants (Ding et al., 2021; Wang et al., 2023), which are naturally order-preserving. Decision and order preservation have also been considered in classification. Mix-n-Match (Zhang et al., 2020) promotes accuracy-preserving calibration and introduces ETS, a convex mixture of temperature scaling, the identity map, and the uniform distribution, to increase expressivity while preserving accuracy. Rahimi et al. (2020) go further by learning intra order-preserving calibration maps with a constrained neural architecture that preserves top-k predictions. In dense prediction, this issue has received comparatively less attention, although the loss mismatch discussed above makes it particularly important. We propose to construct matched decision-preserving counterparts of expressive unconstrained calibrators, in order to isolate the effect of this constraint and quantify the calibration–segmentation trade-off. The main difficulty is to impose argmax or order preservation without reducing the calibrator to temperature scaling. To this end, we introduce class-conditional affine calibrators that can be made to preserve either the original argmax or the complete class ordering while retaining greater expressivity.

3 Methods

3.1 Translation-Invariant Post-hoc Calibration

For a C-class segmentation task, each location v is associated with a ground-truth label $y^v \in \{1, \dots, C\}$ and a logit vector $\mathbf{z}^v \in \mathbb{R}^C$ produced by a segmentation model. Class probabilities are obtained through the softmax transformation $\mathbf{p}^v = S(\mathbf{z}^v)$. Post-hoc calibration learns a mapping g such that the calibrated probabilities $\mathbf{q}^v = g(\mathbf{z}^v)$ better reflect empirical accuracies. In the case of top-label calibration, this can be expressed as

$$\mathbb{P}\big(y^v = \hat{y} \mid \mathbf{q}^v_{\hat{y}}\big) = \mathbf{q}^v_{\hat{y}}, \tag{1}$$

where $\hat{y} = \arg\max_c \mathbf{q}^v_c$. In practice, g is fitted by minimizing a proper scoring rule, typically the negative log-likelihood (NLL), on a calibration set.

Since the softmax is invariant to additive shifts, two logit vectors $\mathbf{z}^{v_1}$ and $\mathbf{z}^{v_2}$ differing by a common additive constant yield identical probability distributions ($S(\mathbf{z}^{v_1}) = S(\mathbf{z}^{v_2})$), but applying a calibrator in logit space can break this equivalence, producing different calibrated predictions: $g(\mathbf{z}^{v_1}) \neq g(\mathbf{z}^{v_2})$. To preserve this equivalence, we define a calibrator as translation-invariant (TI) if it satisfies

$$g(\mathbf{z}) = g(\mathbf{z} + c\mathbf{1}), \quad \forall \mathbf{z} \in \mathbb{R}^C,\ \forall c \in \mathbb{R}. \tag{2}$$

Under this property, any two logit vectors yielding the same softmax probabilities necessarily produce identical calibrated predictions. Equivalently, the output of a TI calibrator depends only on the predictive distribution induced by the softmax and not on the particular logit representation. Translation invariance is a natural consistency requirement for post-hoc calibration of predictive probabilities. Otherwise, the calibrator would depend on a degree of freedom that carries no predictive information.

Standard post-hoc calibrators. We consider standard multiclass post-hoc calibrators applied voxel-wise, including temperature scaling (TS) and its affine extensions, vector scaling (VS) and matrix scal-

ing (MS) (Guo et al., 2017). We further study two TS-based variants: ensemble temperature scaling (ETS) (Zhang et al., 2020), which mixes the temperature-scaled distribution with the original softmax output and a uniform prior, and local temperature scaling (LTS) (Ding et al., 2021), where a temperature function τ_θ (e.g., a CNN) predicts a spatially varying temperature field from the full logit map $\mathbf{Z}$ and optional features ξ:

$$g_T^{\mathrm{TS}}(\mathbf{z}) = S(\mathbf{z}/T), \qquad T > 0, \tag{3}$$

$$g_{\mathbf{a},\mathbf{b}}^{\mathrm{VS}}(\mathbf{z}) = S(\mathrm{diag}(\mathbf{a})\,\mathbf{z} + \mathbf{b}), \qquad \mathbf{a}, \mathbf{b} \in \mathbb{R}^C, \tag{4}$$

$$g_{\mathbf{W},\mathbf{b}}^{\mathrm{MS}}(\mathbf{z}) = S(\mathbf{W}\mathbf{z} + \mathbf{b}), \qquad \mathbf{W} \in \mathbb{R}^{C\times C},\ \mathbf{b} \in \mathbb{R}^C, \tag{5}$$

$$g_{\lambda,T}^{\mathrm{ETS}}(\mathbf{z}) = \lambda_1 S(\mathbf{z}/T) + \lambda_2 S(\mathbf{z}) + \lambda_3 \mathbf{u}, \qquad \lambda_k \geq 0,\ \sum_{k=1}^{3} \lambda_k = 1,\ \mathbf{u} = \frac{1}{C}\mathbf{1}, \tag{6}$$

$$g_\theta^{\mathrm{LTS}}(\mathbf{z}^v) = S(\mathbf{z}^v/\tau_\theta(v;\mathbf{Z},\xi))\,, \qquad \tau_\theta(v;\mathbf{Z},\xi) > 0. \tag{7}$$

The following properties follow directly from the definitions. TS and ETS are translation-invariant (TI). VS is TI if and only if all components of $\mathbf{a}$ are equal, in which case it reduces to temperature scaling with a class-wise additive bias. MS is TI if and only if $\mathbf{W}\mathbf{1} = \alpha\mathbf{1}$ for some $\alpha \in \mathbb{R}$, which is equivalent to all rows of $\mathbf{W}$ summing to the same value. Assuming non-constant logits, LTS is TI if and only if the temperature predictor τ_θ is invariant to additive logit shifts (proof in Appendix A.1). However, this property is not guaranteed in Ding et al. (2021) since τ_θ takes the raw logit field $\mathbf{Z}$ as input.

Violating translation invariance can lead to undesirable behavior. As an example, consider MS: $g(\mathbf{z} + c\mathbf{1}) = S(\mathbf{W}\mathbf{z} + \mathbf{b} + c\mathbf{W}\mathbf{1})$. If $\mathbf{W}\mathbf{1} \notin \mathrm{span}\{\mathbf{1}\}$, then g is not TI; in the extreme case $c \to +\infty$, the output distribution converges to a one-hot distribution concentrated on $k^\star = \arg\max_k (\mathbf{W}\mathbf{1})_k$, independently of $S(\mathbf{z})$. Thus, an arbitrary shift of the logits, which leaves the predictive distribution unchanged, can fully determine the calibrated prediction.

Making calibrators translation-invariant (TI). We consider two mechanisms to enforce translation invariance. The first mechanism constrains the calibrator such that additive logit shifts only induce a constant shift in the pre-softmax outputs. For matrix scaling (MS), this can be achieved by enforcing that all rows of $\mathbf{W}$ have identical sums (this variant is denoted MS_c). This constraint can be implemented via an unconstrained parameterization. Specifically, we learn a free matrix $\mathbf{W}_{\mathrm{free}} \in \mathbb{R}^{C\times(C-1)}$ and a scalar $s \in \mathbb{R}$. The full matrix $\mathbf{W} \in \mathbb{R}^{C\times C}$ is obtained by appending a final column to $\mathbf{W}_{\mathrm{free}}$, defined such that each row sums to s.

The second mechanism consists of feeding the calibrator with canonical logits instead of their raw representatives. In particular, subtracting the free energy $F(\mathbf{z})$ (with a sign convention opposite to some energy-based formulations) from $\mathbf{z}$ yields a canonical representative of the equivalence class, removing the arbitrary global offset while preserving the predictive distribution:

$$\log S(\mathbf{z}) = \mathbf{z} - F(\mathbf{z})\,\mathbf{1}, \tag{8}$$

$$F(\mathbf{z}) = \log \sum_{c=1}^{C} e^{z_c}. \tag{9}$$

Equation 8 shows that feeding a calibrator with $\log S(\mathbf{z})$ is equivalent to subtracting the free energy $F(\mathbf{z})$ from $\mathbf{z}$, both ensuring translation invariance.

This viewpoint recovers Dirichlet calibration (DC) (Kull et al., 2019) as MS applied to canonical logits:

$$g_{W,b}^{\mathrm{DC}}(\mathbf{z}) = S\big(\mathbf{W}\log S(\mathbf{z}) + \mathbf{b}\big), \tag{10}$$

and therefore DC is TI by construction. The same idea applies to LTS. The temperature predictor τ_θ can be made invariant to additive logit shifts by using log-probabilities as input instead of raw logits:

$$g_\theta^{\mathrm{LTS}}(\mathbf{z}^v) = S\big(\mathbf{z}^v/\tau_\theta\big(v;\log S(\mathbf{Z}),\xi\big)\big)\,. \tag{11}$$

3.2 Argmax/Order Preservation in Calibration

Since calibration and segmentation objectives are not fully aligned, improvements in calibration may come at the expense of segmentation accuracy. It remains an open question whether enforcing argmax preservation during calibration limits the achievable calibration gains. To investigate this question, we construct argmax- and order-preserving counterparts of standard calibrators to quantify the resulting calibration–segmentation trade-off. Among the standard multiclass post-hoc calibrators presented in Eqs. 3–7, only TS, ETS, and LTS are inherently argmax-preserving and order-preserving. When argmax- or order-preserving constraints are imposed, VS and MS collapse to TS (Appendix A.2). Consequently, comparisons between VS/MS and TS do not isolate the effect of enforcing argmax preservation, as they simultaneously involve a severe reduction in model expressiveness.

To obtain meaningful argmax- and order-preserving counterparts of standard calibrators, one must ensure that the imposed constraints do not collapse the model to overly restrictive forms. This requires introducing additional flexibility beyond global transformations. We introduce a conditional calibration framework in which each class is equipped with a dedicated calibrator, selected at inference time according to the argmax of the input logits. Class-conditioning alone does not guarantee argmax or order preservation, since a selected calibrator may still alter the ranking of the logits. We therefore enforce argmax or order preservation within each class-specific calibrator, which ensures that the resulting global model inherits the same property.

3.2.1 Enforcing Argmax and Order Preservation for Conditional Class-wise Calibration

General case. Let Ψ_1 and Ψ_2 denote the argmax and order cones, respectively:

$$\Psi_1 = \{\mathbf{z} \in \mathbb{R}^C \mid z_1 \geq z_k,\ \forall k \neq 1\},$$
$$\Psi_2 = \{\mathbf{z} \in \mathbb{R}^C \mid z_1 \geq z_2 \geq \cdots \geq z_C\}.$$

For any permutation π, we denote by $\mathbf{v}[\pi]$ the vector obtained by reordering $\mathbf{v}$ according to π:

$$\mathbf{v}[\pi] = \big(v_{\pi(1)}, \ldots, v_{\pi(C)}\big).$$

We introduce a permutation $\pi_{\mathbf{z}}$ associated with each input $\mathbf{z}$, which induces the canonical reordering $\mathbf{z}^{\text{can}} = \mathbf{z}[\pi_{\mathbf{z}}]$, where $\pi_{\mathbf{z}}$ is chosen such that $\mathbf{z}^{\text{can}} \in \Psi_1$ or $\mathbf{z}^{\text{can}} \in \Psi_2$, depending on whether argmax or order preservation is enforced. In the argmax-preserving case (Ψ_1), $\pi_{\mathbf{z}}$ exchanges the first index with the index of $\arg\max_k z_k$, while leaving all other indices unchanged. In the order-preserving case (Ψ_2), $\pi_{\mathbf{z}}$ is defined such that the components of $\mathbf{z}^{\text{can}}$ are sorted in non-increasing order.

Let $i(\mathbf{z}) = \pi_{\mathbf{z}}(1)$ denote the class occupying the first position in the canonical ordering. The corresponding class-wise calibrator $g_{i(\mathbf{z})}$ is then applied to $\mathbf{z}^{\text{can}}$, producing calibrated probabilities in canonical order. The final calibrated probabilities are obtained by applying the inverse permutation:

$$g(\mathbf{z}) = \big(g_{i(\mathbf{z})}\,(\mathbf{z}[\pi_{\mathbf{z}}])\big)\,[\pi_{\mathbf{z}}^{-1}], \qquad i(\mathbf{z}) = \pi_{\mathbf{z}}(1).$$

Let $h_i : \mathbb{R}^C \to \mathbb{R}^C$ denote the pre-softmax transformation associated with the i-th class-wise calibrator, so that $g_i = S \circ h_i$. The conditional class-wise calibration is argmax-preserving (resp. order-preserving) if Ψ_1 (resp. Ψ_2) is invariant under h_i, i.e., $h_i(\mathbf{z}) \in \Psi_1 \quad \forall \mathbf{z} \in \Psi_1$ (resp. Ψ_2).

The key advantage of the conditional formulation is that preservation constraints are enforced in a fixed canonical space, independent of the original permutation of logits. This is less restrictive than enforcing argmax or order preservation directly in the original logit space, where the location of the maximum and the ordering pattern may vary across samples. Note that this leads to a difference in the information available to the class-wise calibrators. In the argmax-preserving setting, g_i operates on inputs expressed in a fixed canonical order, where the mapping between positions and class identities remains consistent across samples. In contrast, in the order-preserving setting, only the first position is consistently associated with the selected class i, whereas the remaining positions are associated with ranks rather than fixed class identities. In principle, one could consider extending the order-preserving formulation by learning separate calibrators

for different ordering patterns, thereby explicitly modeling distinct rank configurations. In practice, however, this would lead to a combinatorial explosion in the number of possible ordering patterns, making it intractable.

Argmax preservation for conditional class-wise calibration under MS. We derive the constraint for a single class-wise affine map and omit the class index i for readability. We first enforce set invariance of Ψ_1 under MS in the absence of bias: $h(\mathbf{z}) = \mathbf{W}\mathbf{z}$. We introduce the transformed coordinates $\mathbf{x} = \mathbf{G}\mathbf{z}$, where

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & -1 & 0 & \cdots & 0 \\ 1 & 0 & -1 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 1 & 0 & \cdots & 0 & -1 \end{pmatrix} \in \mathbb{R}^{C \times C}. \tag{12}$$

In this coordinate system, the argmax cone Ψ_1 is mapped to

$$\Psi_3 = \{\mathbf{x} \in \mathbb{R}^C \mid x_i \geq 0, i \geq 2\}.$$

Invariance of Ψ_1 under $\mathbf{W}$ is equivalent to invariance of Ψ_3 under $\mathbf{W}' = \mathbf{G}\mathbf{W}\mathbf{G}^{-1}$, which holds if and only if

$$\mathbf{W}' = \begin{pmatrix} a & \mathbf{r}^\top \\ \mathbf{0} & \mathbf{A} \end{pmatrix}, \quad a \in \mathbb{R},\ \mathbf{r} \in \mathbb{R}^{C-1},\ \mathbf{0} \in \mathbb{R}^{C-1},\ \mathbf{A} \in \mathbb{R}_+^{(C-1)\times(C-1)}.$$

We further extend the framework to the biased case ($\mathbf{b} \neq \mathbf{0}$). Since the bias term directly shifts the logits, preserving the argmax-cone structure requires the bias vector itself to lie in the same cone, i.e., $\mathbf{b} \in \Psi_1$.

An important property of the resulting calibration method is its translation invariance, which can be verified by checking that $\mathbf{W}\mathbf{1}_C = a\mathbf{1}_C$. Moreover, we derive from the previous results a fully differentiable and unconstrained parameterization. We parameterize a free matrix $\tilde{\mathbf{A}} \in \mathbb{R}^{(C-1)\times(C-1)}$ and enforce non-negativity via a softplus transformation: $\mathbf{A}_{ij} = \mathrm{softplus}(\tilde{\mathbf{A}}_{ij})$. We then construct $\mathbf{W}' \in \mathbb{R}^{C\times C}$ by embedding $\mathbf{A}$ into the lower-right block, setting the first column (except for $\mathbf{W}'_{11}$) to zero, and parameterizing the first row as an unconstrained vector in $\mathbb{R}^C$. Then, $\mathbf{W}$ is set to $\mathbf{G}^{-1}\mathbf{W}'\mathbf{G}$. For the bias, we parameterize $\tilde{\mathbf{b}} \in \mathbb{R}^C$ and enforce ordering via $\mathbf{b}_1 = \tilde{\mathbf{b}}_1$ and $\mathbf{b}_i = \mathbf{b}_1 - \mathrm{softplus}(\tilde{\mathbf{b}}_i)$ if $i \geq 2$.

Order preservation for conditional class-wise calibration under MS. Enforcing invariance of Ψ_2 under MS follows exactly the same construction as for Ψ_1 with:

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & -1 & 0 & \cdots & 0 \\ 0 & 1 & -1 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & -1 \end{pmatrix} \in \mathbb{R}^{C \times C}.$$

The bias vector must now satisfy $\mathbf{b} \in \Psi_2$. To enforce this constraint, we parameterize a free vector $\tilde{\mathbf{b}} \in \mathbb{R}^C$ and define $\mathbf{b}_1 = \tilde{\mathbf{b}}_1$, $\mathbf{b}_i = \mathbf{b}_{i-1} - \mathrm{softplus}(\tilde{\mathbf{b}}_i)$ if $i \geq 2$, which guarantees $\mathbf{b}_1 \geq \mathbf{b}_2 \geq \cdots \geq \mathbf{b}_C$.

3.3 Instantiation of Segmentation/Calibration Pipelines

Logits provided to the calibrator are produced by a segmentation model. In this work, we consider deep ensembles (Lakshminarayanan et al., 2017), which are widely adopted in semantic segmentation to improve performance and predictive robustness. They typically yield higher Dice scores (DSC) and improved reliability compared to single networks (Lakshminarayanan et al., 2017; Fort et al., 2020; Ovadia et al., 2019).

However, improved segmentation performance does not necessarily translate into well-calibrated confidence estimates, and ensemble predictions are often still miscalibrated (Rahaman & Thiery, 2021).

An ensemble consists of M independently trained segmentation models. At voxel v, model m produces logits $\mathbf{z}_m(v) \in \mathbb{R}^C$. Ensemble predictions are aggregated by a pooling operator acting either on logits $\{\mathbf{z}_m(v)\}_{m=1}^M$ or on probabilities $\{\mathbf{p}_m(v)\}_{m=1}^M$, yielding pooled logits $\mathbf{z}(v)$ or pooled probabilities $\mathbf{p}(v)$ that are passed to the calibrator. When pooling is performed in probability space, we provide $\log \mathbf{p}(v)$ to the calibrator, ensuring consistency with the logit-based formulation.

We consider two standard pooling strategies: probability averaging, also known as mixture-of-experts fusion, and logit averaging, which induces a product-of-experts–type aggregation and penalizes inter-model disagreement more strongly. Pipelines based on probability averaging are denoted $g(\bar{\mathbf{p}})$, whereas those based on logit averaging are denoted $g(\bar{\mathbf{z}})$. Both pooling operators are invariant to member-wise additive logit shifts: adding $\delta_m \mathbf{1}$ to all logits of model m leaves the resulting pooled probabilities unchanged. Moreover, $g(\bar{\mathbf{p}})$ pipelines are TI by construction, since the calibrator operates on log-probabilities rather than logits, which guarantees translation invariance (Sec. 3.1).

We next introduce the pipelines considered in our experiments.

Uncalibrated baselines. To assess the intrinsic impact of calibration, we consider three uncalibrated baselines: a single segmentation model, denoted p_0, and a deep ensemble using either logit averaging ($\bar{\mathbf{z}}$) or probability averaging ($\bar{\mathbf{p}}$) for pooling.

Translation invariance. To investigate the effect of translation invariance, we compare the non-TI pipeline $\mathrm{MS}(\bar{\mathbf{z}})$ with two TI counterparts: $\mathrm{MS}_c(\bar{\mathbf{z}})$ and $\mathrm{DC}(\bar{\mathbf{z}})$. MS_c enforces translation invariance in logit space via a row-sum constraint on the weight matrix, while $\mathrm{DC}(\bar{\mathbf{z}})$ can be interpreted as applying MS to canonical log-probabilities $\log S(\bar{\mathbf{z}})$ (cf. Section 3.1). Note that MS becomes equivalent to DC when pooling is performed in probability space ($\bar{\mathbf{p}}$); therefore, it is not included in this comparison.

For LTS, translation invariance depends on the representation provided to the temperature network τ_θ, as discussed in Section 3.1. We consider two variants: the original logit-based formulation of Ding et al. (2021), which is not TI, and a probability-based variant that enforces it. The first is denoted $\mathrm{LTS}(\bar{\mathbf{z}})$, and the second $\mathrm{LTS}(\log S(\bar{\mathbf{z}}))$, highlighting that the temperature network operates on canonical log-probabilities of the pooled predictions.

Argmax/Order preservation. We denote $\mathrm{CDC}(\bar{\mathbf{p}})$ and $\mathrm{CDC}(\bar{\mathbf{z}})$ the conditional class-wise calibration schemes introduced in Section 3.2.1, where each class is equipped with a DC calibrator. Consequently, both pipelines are translation-invariant (TI) but neither order- nor argmax-preserving. We compare $\mathrm{CDC}(\bar{\mathbf{p}})$ and $\mathrm{CDC}(\bar{\mathbf{z}})$ with their respective order- and argmax-preserving counterparts, constructed using the algorithms presented at the end of Section 3.2.1. For probability pooling, $\mathrm{CDC}(\bar{\mathbf{p}})$ is compared with $\mathrm{CMS}_{\mathrm{op}}(\bar{\mathbf{p}})$ and $\mathrm{CMS}_{\mathrm{ap}}(\bar{\mathbf{p}})$, where $\mathrm{CMS}_{\mathrm{op}}$ enforces order preservation and $\mathrm{CMS}_{\mathrm{ap}}$ enforces argmax preservation. Similarly, for logit pooling, $\mathrm{CDC}(\bar{\mathbf{z}})$ is compared with $\mathrm{CMS}_{\mathrm{op}}(\bar{\mathbf{z}})$ and $\mathrm{CMS}_{\mathrm{ap}}(\bar{\mathbf{z}})$.

Note that $\mathrm{CMS}_{\mathrm{op}}$ and $\mathrm{CMS}_{\mathrm{ap}}$ are TI, whereas CMS is not. Therefore, comparing $\mathrm{CMS}_{\mathrm{op}}$ and $\mathrm{CMS}_{\mathrm{ap}}$ with CMS would conflate translation invariance with order or argmax constraints. In contrast, CDC is TI by construction and is thus used as the non–order-/non–argmax-preserving reference.

Standard pipelines. We also evaluate standard temperature-based calibrators (TS, ETS), which preserve class ordering and are TI. We further include Dirichlet calibration (DC), which is TI by construction. This allows us to assess both the impact of the pooling operator and the role of calibrator expressiveness on performance.

Dataset	Modality	C	Seg. train	$\lvert\mathcal{D}_{\text{cal}}\rvert$	$\lvert\mathcal{D}_{\text{val}}\rvert$	$\lvert\mathcal{D}_{\text{test}}\rvert$	Pool
BraTS GLI 2024 (Correia de Verdier et al., 2024)	3D MRI	5	1350	50	27	194	271
Cityscapes (Cordts et al., 2016)	2D RGB	20	2975	50	50	400	500
Massachusetts Roads (RoadSeg) (Mnih, 2013)	2D RGB	2	566	50	24	169	243

Table 1: **Dataset splits and class counts.** Segmentation models are trained on the official training split for BraTS (*TrainData*, 1350 cases) and Cityscapes (*train*, 2975 images). The remaining data form a held-out pool partitioned into $\mathcal{D}_{\text{cal}}/\mathcal{D}_{\text{val}}/\mathcal{D}_{\text{test}}$: for BraTS, we use the official additional training release (271 labeled cases); for Cityscapes we use the official *val* set (test-set labels are withheld); for Massachusetts Roads, we exclude incomplete/occluded images, yielding 809 image–mask pairs, of which 243 form the held-out pool.

Figure 2: Sample inputs with ground-truth segmentation overlays from Massachusetts Roads, Cityscapes, and BraTS2024 Glioma (left to right). For BraTS, we show a 2D slice from the 3D MRI volume.

4 Experiments and Results

4.1 Experimental Setup

Datasets. We evaluate the calibration pipelines from Section 3.3 on three segmentation benchmarks spanning medical and natural-image domains: BraTS GLI 2024 (Correia de Verdier et al., 2024) (3D brain MRI glioma segmentation), Cityscapes (Cordts et al., 2016) (2D urban scene parsing), and Massachusetts Roads (RoadSeg) (Mnih, 2013) (2D aerial road extraction). These datasets vary in modality (MRI vs. RGB), dimensionality (3D vs. 2D), and number of classes, thereby enabling evaluation across heterogeneous imaging and semantic settings. Dataset statistics, split sizes, and the number of classes are summarized in Table 1. As shown in the table, we use a calibration set of size $n = 50$ for all experiments; a sensitivity analysis (Appendix F) indicates that calibration metrics largely stabilize for $n \geq 25$. Representative input images and ground-truth overlays are shown in Figure 2.

Segmenter training. For each dataset, we train a deep ensemble of $M = 5$ nnU-Net (Isensee et al., 2021) v2.4.1 models initialized with different random seeds. The 2D or 3D configuration is selected according to the dataset dimensionality, both using the `ResEncM` preset and default preprocessing. Optimization follows the nnU-Net defaults (SGD with momentum, `polyLR` schedule, deep supervision, and automatic patch and batch-size configuration). All ensemble members share identical preprocessing, data augmentation, and optimization settings. The ensemble is trained once per dataset on the official training split and remains fixed for all subsequent calibration experiments.

Post-hoc calibration protocol. Calibration is performed post hoc on the fixed ensemble segmenter. For each dataset, we perform $R = 3$ independent repeats, each using a new random partition of the held-out images (Table 1) into calibration, validation, and test subsets. In each repeat r, calibrator parameters are fitted on $\mathcal{D}_{\text{cal}}^{(r)}$ by minimizing the voxel-wise negative log-likelihood (NLL) of the pooled ensemble predictions with an additional regularization term. When applicable, early stopping and hyperparameter selection use $\mathcal{D}_{\text{val}}^{(r)}$, and final results are reported on $\mathcal{D}_{\text{test}}^{(r)}$. Details on optimization, regularization, and hyperparameter search are provided in Appendix B.

Calibrator	TI	Decision preservation	Optimized	Identifiable	RoadSeg (C=2)	BraTS (C=5)	Cityscapes (C=20)
Global calibrators							
TS	$\surd$	Order	1	1	1/1	1/1	1/1
ETS	$\surd$	Order	4	3	4/3	4/3	4/3
MS	$\times$	$\times$	$C(C+1)$	C^2-1	6/3	30/24	420/399
$\mathrm{MS_c}$	$\surd$	$\times$	C^2+1	$C(C-1)$	5/2	26/20	401/380
DC	$\surd$	$\times$	$C(C+1)$	C^2-1	6/3	30/24	420/399
Spatially adaptive or class-conditional calibrators							
LTS	$\times^{\dagger}$	Order	$7(25C+1)$	$7(25C+1)$	357/357	882/882	3 507/3 507
CDC	$\surd$	$\times$	$C^2(C+1)$	$C(C^2-1)$	12/6	150/120	8 400/7 980
$\mathrm{CMS_{ap}}$	$\surd$	Argmax	$C(C^2+1)$	$C^2(C-1)$	10/4	130/100	8 020/7 600
$\mathrm{CMS_{op}}$	$\surd$	Order	$C(C^2+1)$	$C^2(C-1)$	10/4	130/100	8 020/7 600

Table 2: **Calibrator properties, optimized parameters, and identifiable degrees of freedom.** "TI" indicates a translation-invariant calibrator; "Decision preservation" indicates whether the calibrator preserves the predicted decision (Argmax), the full class ordering (Order), or neither ($\times$). "Optimized" counts fitted scalars; "Identifiable" removes continuous softmax-gauge directions. The last three columns report the optimized and identifiable counts for each calibrator and dataset. †LTS is TI only for our variant $\mathrm{LTS}(\log S(\bar{z}))$, in which the temperature network operates on canonical log-probabilities, not for the original formulation $\mathrm{LTS}(\bar{z})$.

Model parameterization and expressiveness A common way to compare the expressiveness of calibration methods is to count trainable parameters. However, this can be partially misleading since most used models are overparameterized due to the softmax's invariance to additive constants. It is therefore more meaningful to consider the number of identifiable parameters. For example, in the MS case, subtracting from each column of $\mathbf{W}$ its first entry and subtracting b_1 from $\mathbf{b}$ leaves the induced softmax distribution unchanged. Consequently, without loss of generality, one may fix the first row of $\mathbf{W}$ to zero and set $\mathbf{b}_1 = 0$.

Table 2 reports both the total number of parameters and the number of identifiable parameters for each calibrator. In particular, the identifiable parameter counts exploit the fact that any MS-based calibrator admits an equivalent parameterization with $\mathbf{b}_1 = 0$ and a zero first row in $\mathbf{W}$. For CMS_{ap} and CMS_{op}, the first row of the matrix $\mathbf{A}$ from which $\mathbf{W}$ is computed can also be fixed to zero without loss of generality.

In practice, we retain the overparameterized parameterizations during optimization and account for these redundancies only when comparing model expressiveness. Importantly, overparameterized models are necessary during optimization due to the regularization term used during training (cf. Appendix B). We use a Dirichlet-style regularizer, described in Kull et al. (2019), that penalizes off-diagonal weights and large biases via an ℓ_2 norm. Under such a regularization, the identity mapping is not penalized; however, in the reduced parameterization presented above, the identity is not representable, and its equivalent representation has a first column given by $(0, -1, -1, \dots)^\top$, which is strongly penalized. This leads to different regularization behavior depending on the chosen parameterization.

Evaluation. Segmentation accuracy is measured with the multiclass Dice score (DSC), which allows us to quantify whether calibration affects segmentation performance. For methods that may change the predicted class, we further report the class-flip rate, defined as the percentage of voxels whose predicted class changes after calibration. Calibration is assessed with negative log-likelihood (NLL), expected calibration error (ECE) (Guo et al., 2017), boundary-aware ECE (BA-ECE) (Zeevi et al., 2025), and Average Calibration Error (ACE) (Kahl et al., 2024). NLL is a strictly proper scoring rule and corresponds to the loss used to fit the calibration models. ECE is computed with $B = 50$ fixed-width confidence bins; a sensitivity analysis over $B \in \{20, 30, 40, 50\}$ shows that ECE values change only marginally and that the relative ranking across methods is stable (Appendix G). BA-ECE evaluates calibration as a function of distance to the ground-truth boundary; details on distance bands and weighting are provided in Appendix C. Since ECE weights bin-wise calibration gaps by bin occupancy, it can be dominated by highly populated confidence ranges. We therefore

Dataset	$n_{\textbf{cases}}$	$\mathbb{E}[\mathrm{Std}_v(F_{\mathrm{pool}}(v))]$
RoadSeg	243	1.08 [0.73, 1.06, 1.34]
Cityscapes	500	2.54 [2.30, 2.46, 2.71]
BraTS	271	8.77 [8.56, 8.69, 8.87]

Table 3: Spatial variability of the pooled-logit free energy $F_{\mathrm{pool}}(v) = \log \sum_{c=1}^{C} \exp(z_{\mathrm{pool},c}(v))$, where $z_{\mathrm{pool}}(v)$ denotes the pooled logits at voxel v. Statistics are computed from nnU-Net ensembles with $M = 5$. For each dataset, we report the mean of $\mathrm{Std}_v(F_{\mathrm{pool}}(v))$ across held-out cases, with interquartile range $[p_{25}, p_{50}, p_{75}]$.

Figure 3: Spatial maps of the pooled-logit free energy $F_{\mathrm{pool}}(v) = \log \sum_{c=1}^{C} \exp(z_{\mathrm{pool},c}(v))$ for representative test images from RoadSeg, Cityscapes, and BraTS.

also report ACE, which averages calibration gaps uniformly over non-empty bins; we use $B = 15$, with a binning sensitivity analysis also reported in Appendix G. All metrics, except ACE, are computed for each image in the held-out test set and then averaged across repeats. We report 95% confidence intervals obtained via a two-level hierarchical bootstrap (Efron & Tibshirani, 1994) with $N_{\mathrm{boot}} = 2000$ replicates, resampling repeats first and then cases within each selected repeat. For ACE, image-level estimates can be unstable because some confidence bins may contain very few voxels. We therefore compute ACE by pooling voxels within each repeat and report the resulting repeat-level values as mean [min, max].

4.2 Pooled Logit Offset Variability

As discussed in Sec. 3.1, segmenter logits can be shifted by an arbitrary constant without affecting the resulting softmax probabilities. To analyze whether such shifts occur in practice, we compute the free energy (cf. Equation 9). Since such offsets can influence non-TI calibrators, this provides a practical diagnostic for whether enforcing translation invariance can matter. It is not, by itself, a predictor of the empirical gain from enforcing invariance: a non-TI calibrator may still learn an approximately invariant solution if the calibration data expose the relevant offsets. For each held-out image, we compute the spatial standard deviation of the pooled-logit free energy, $\mathrm{Std}_v(F_{\mathrm{pool}}(v))$. Table 3 reports the mean across images for ensembles with M=5 models. Spatial variability is substantial for all datasets and is markedly larger for BraTS and Cityscapes than for RoadSeg. The same trend holds for smaller ensembles (see Appendix E).

Figure 3 shows representative spatial maps of $F_{\mathrm{pool}}(v)$. These maps illustrate the spatial variability of the pooled-logit free energy across datasets.

4.3 Comparison of Post-Hoc Calibration Pipelines

We evaluate the calibration pipelines (Sec. 3.3) across the three segmentation benchmarks. Table 4 summarizes the results in terms of calibration and segmentation metrics. For conciseness, only point estimates are reported; corresponding intervals are provided in Appendix I. Note that the trends observed in-distribution persist under corruption-based covariate shift on Cityscapes, and are often amplified under stronger perturbations (Appendix H):

Pipeline	**RoadSeg** Mnih (2013)						**BraTS** Correia de Verdier et al. (2024)						**Cityscapes** Cordts et al. (2016)					
	NLL↓	DSC↑	ECE↓	BA-ECE↓	ACE↓	Flip↓	NLL↓	DSC↑	ECE↓	BA-ECE↓	ACE↓	Flip↓	NLL↓	DSC↑	ECE↓	BA-ECE↓	ACE↓	Flip↓
(a) Uncalibrated baselines																		
p_0	0.091	76.3	0.017	0.094	0.119	—	0.524	75.2	0.340	0.049	0.112	—	0.485	71.7	0.063	0.182	0.125	—
$\bar{p}$	0.085	**76.7**	0.015	0.087	0.105	—	0.522	76.0	0.340	0.041	0.087	—	0.377	78.1	0.044	0.136	0.079	—
$\bar{z}$	0.088	**76.7**	0.016	0.091	0.117	—	0.523	**76.9**	0.340	0.046	0.096	—	0.434	**79.4**	0.058	0.170	0.122	—
(b) Translation invariance: MS vs. MS_c and DC																		
$\mathrm{MS}(\bar{z})^{\times}$	0.068	76.5	0.010	0.056	0.053	0.2	0.008	68.4	0.001	0.043	0.069	**0.1**	0.404	73.4	0.045	0.156	0.072	3.8
$\mathrm{DC}(\bar{z})$	0.067	**76.7**	0.008	0.047	0.040	**0.1**	0.008	67.6	0.001	0.040	0.065	**0.1**	0.324	70.4	0.035	0.096	0.040	**2.3**
$\mathrm{MS}_c(\bar{z})$	0.068	76.5	0.010	0.057	0.055	0.2	**0.007**	70.2	0.001	0.033	0.041	**0.1**	0.328	72.8	0.034	0.098	0.046	2.4
(c) Translation invariance: LTS with logits vs. log-probabilities																		
$\mathrm{LTS}(\bar{z})^{\times}$	0.067	**76.7**	0.004	0.047	0.024	—	0.018	**76.9**	0.012	0.134	0.237	—	0.344	**79.4**	0.016	0.075	0.039	—
$\mathrm{LTS}(\log S(\bar{z}))$	0.066	**76.7**	**0.002**	**0.040**	**0.007**	—	0.014	**76.9**	0.008	0.090	0.216	—	0.342	**79.4**	**0.010**	0.075	0.029	—
(d) Decision preservation: CDC vs CMS																		
$\mathrm{CDC}(\bar{p})$	**0.065**	75.7	0.006	0.043	0.013	0.5	0.008	72.0	0.001	0.034	0.046	**0.1**	0.335	69.3	0.032	0.108	0.052	5.1
$\mathrm{CMS}_{ap}(\bar{p})$	0.067	**76.7**	0.007	0.046	0.036	—	**0.007**	76.0	0.001	0.029	0.044	—	**0.314**	78.1	0.035	0.086	0.041	—
$\mathrm{CMS}_{op}(\bar{p})$	0.067	**76.7**	0.007	0.046	0.037	—	**0.007**	76.0	0.001	**0.028**	0.065	—	0.326	78.1	0.035	0.071	0.032	—
$\mathrm{CDC}(\bar{z})$	0.066	75.7	0.006	0.041	0.012	0.6	0.008	74.0	0.001	0.035	**0.033**	**0.1**	0.341	74.2	0.035	0.115	0.067	5.0
$\mathrm{CMS}_{ap}(\bar{z})$	0.067	**76.7**	0.008	0.047	0.043	—	**0.007**	**76.9**	0.001	0.031	0.041	—	0.327	**79.4**	0.038	0.091	0.081	—
$\mathrm{CMS}_{op}(\bar{z})$	0.067	**76.7**	0.007	0.047	0.043	—	**0.007**	**76.9**	0.001	0.030	0.055	—	0.339	**79.4**	0.035	0.079	0.041	—
(e) Standard temperature-based calibrators (order-preserving, TI): TS/ETS																		
$\mathrm{TS}(\bar{p})$	0.068	**76.7**	0.008	0.059	0.056	—	0.025	76.0	0.005	0.046	0.146	—	0.326	78.1	0.011	**0.069**	**0.025**	—
$\mathrm{TS}(\bar{z})$	0.068	**76.7**	0.009	0.061	0.065	—	0.029	**76.9**	0.007	0.048	0.167	—	0.338	**79.4**	0.012	0.079	0.039	—
$\mathrm{ETS}(\bar{p})$	0.068	**76.7**	0.008	0.059	0.055	—	0.010	76.0	**0.000**	0.047	0.166	—	0.326	78.1	**0.010**	0.071	0.026	—
$\mathrm{ETS}(\bar{z})$	0.068	**76.7**	0.009	0.061	0.065	—	0.010	**76.9**	**0.000**	0.048	0.187	—	0.338	**79.4**	0.013	0.079	0.039	—
(f) Standard affine calibrator (non order-preserving, TI): DC																		
$\mathrm{DC}(\bar{p})$	0.067	76.6	0.008	0.046	0.033	**0.1**	0.008	68.4	0.001	0.037	0.057	**0.1**	0.319	64.8	0.033	0.098	0.037	3.2
$\mathrm{DC}(\bar{z})$	0.067	**76.7**	0.008	0.047	0.040	**0.1**	0.008	67.6	0.001	0.040	0.065	**0.1**	0.324	70.4	0.035	0.096	0.040	**2.3**

Table 4: **Calibration results.** DSC and Flip are reported in %. — indicates not applicable; × denotes non–translation-invariant pipelines. **Bold** indicates the best value per dataset and metric. Light row colors identify the matched comparison family: blue for translation invariance, green for decision preservation, and beige for standard calibrators.

1. Ensembles improve segmentation but do not reliably fix calibration. (Table 4(a)) Ensemble pooling consistently improves segmentation accuracy over a single model, although the magnitude of the gain depends on the dataset. On Cityscapes, DSC increases from 71.7 (p_0) to 78.1 with probability pooling ($\bar{p}$) and 79.4 with logit pooling ($\bar{z}$). On BraTS, the improvement is more moderate (75.2 to 76.0–76.9), while on RoadSeg it is marginal (76.3 to 76.7). Logit pooling yields slightly higher DSC than probability pooling on BraTS and Cityscapes, with negligible differences on RoadSeg.

For calibration, ensemble pooling improves or leaves metrics essentially unchanged, but the magnitude of the gains depends on the dataset and pooling strategy. Improvements are marginal on RoadSeg, more pronounced on Cityscapes with probability pooling, and mixed across metrics on BraTS.

Overall, compared with a single model, ensembling improves or preserves segmentation and does not degrade calibration metrics in our experiments. However, the calibration gains remain modest compared with those obtained by dedicated post-hoc calibration methods, motivating the calibration pipelines evaluated next.

2. TI pipelines provide representation-consistent calibration. We next evaluate whether enforcing translation invariance (TI) improves calibration in matched pipeline comparisons. Calibration quality also depends on other factors, in particular calibrator expressivity. It is therefore not surprising that, outside matched pairs, a non-TI method may outperform a TI method.

We first compare matrix scaling $\mathrm{MS}(\bar{z})$ with its two TI counterparts $\mathrm{MS}_c(\bar{z})$ and $\mathrm{DC}(\bar{z})$ (Table 4(b)). Cityscapes gives the clearest effect: relative to $\mathrm{MS}(\bar{z})$, both TI variants reduce NLL (from 0.404 to 0.324–0.328), BA-ECE (from 0.156 to 0.096–0.098), and ACE (from 0.072 to 0.040–0.046). On BraTS, where ECE is dominated by the background class, class- and boundary-aware metrics show the same trend: MS_c improves BA-ECE from 0.043 to 0.033 and ACE from 0.069 to 0.041, while DC gives smaller gains. These gains are unlikely to be explained solely by differences in model complexity, as MS and DC have the same number of trainable and identifiable parameters, while MS_c has slightly fewer of both (Table 2). Rather, they further support the benefit of removing representation sensitivity. On RoadSeg, all affine variants are close, consistent with the lower pooled free-energy variability reported in Section 4.2. The binary setting also makes MS_c particularly restrictive: for $C = 2$, MS_c has only two identifiable parameters, compared with three for MS and DC (Table 2), which may explain why DC can be slightly preferable despite both variants being TI.

Taken together, these results support TI as a useful design principle for affine calibration: it makes calibration representation-consistent and improves or matches shift-sensitive counterparts. However, these affine calibrators are not constrained to preserve the argmax, so improving calibration can come at the cost of changing the predicted segmentation. Compared with the uncalibrated logit-pooled ensemble $\bar{z}$, the three calibration pipelines have negligible DSC changes on RoadSeg (76.7 to 76.5–76.7), but large drops on BraTS (76.9 to 67.6–70.2) and Cityscapes (79.4 to 70.4–73.4). Thus, the calibration gains obtained by these affine methods are accompanied by substantial losses in segmentation accuracy on two of the three datasets. A flip audit for $\mathrm{MS}(\bar{z})$ shows that harmful flips generally outnumber beneficial ones: 50.4% vs. 29.5% on Cityscapes and 74.8% vs. 18.6% on BraTS, with the remaining flips leaving voxel-level accuracy unchanged. Visual examples of the spatial distribution of these flips are provided in Appendix D. This motivates the decision-preserving variants evaluated next in Point 3.

For LTS, we compare the original logit-based formulation $\mathrm{LTS}(\bar{z})$ from Ding et al. (2021), which does not guarantee translation invariance, with our TI variant $\mathrm{LTS}(\log S(\bar{z}))$ (Table 4(c)). This is a matched comparison: both variants use the same local temperature network, have the same number of parameters (357 on RoadSeg, 882 on BraTS, and 3,507 on Cityscapes; Table 2), and preserve the order by construction, so DSC is unchanged from the input segmentation. Across the three datasets, the TI variant improves almost all calibration metrics, with BA-ECE on Cityscapes remaining unchanged. The gains are particularly clear on RoadSeg, where ECE decreases from 0.004 to 0.002 and ACE from 0.024 to 0.007, and on BraTS, where BA-ECE decreases from 0.134 to 0.090. $\mathrm{LTS}(\log S(\bar{z}))$ also improves ACE on BraTS (0.237 to 0.216), although ACE remains worse than for the uncalibrated baselines. We attribute this to the severe foreground–background imbalance: the voxel-wise NLL objective and ECE are dominated by easy background voxels, whereas BA-ECE and ACE emphasize sparse uncertain regions and tumor boundaries. The flexible local temperature map can therefore fit the dominant confidence regime without reliably correcting calibration on rare foreground structures. Overall, using canonical inputs makes LTS representation-consistent and improves calibration without increasing model capacity, highlighting the practical benefit of translation invariance.

3. Decision preservation exposes dataset-dependent calibration–segmentation effects (Table 4(d)). We next isolate the effect of decision preservation within expressive affine calibration by comparing the unconstrained CDC pipeline with its argmax- and order-preserving counterparts, $\mathrm{CMS}_{\mathrm{ap}}$ and $\mathrm{CMS}_{\mathrm{op}}$. Unlike CDC, the CMS variants preserve the original segmentation map by construction and therefore retain the pooled DSC. Across all datasets, unconstrained CDC induces decision changes. These changes are small on RoadSeg and BraTS, where CDC flips at most 0.6% and 0.1% of pixels, respectively, but become substantial on Cityscapes, with about 5% of pixels changing label. Consequently, CDC consistently reduces DSC relative to the decision-preserving CMS variants, with the largest drops on Cityscapes (69.3 vs. 78.1 for $\bar{p}$, and 74.2 vs. 79.4 for $\bar{z}$), followed by BraTS (72.0 vs. 76.0 and 74.0 vs. 76.9). On RoadSeg, the effect is more modest, with DSC decreasing from 76.7 to 75.7.

The effect on calibration is both dataset- and metric-dependent. On Cityscapes, decision preservation does not reveal a clear calibration cost. While ECE is slightly increased or remains unchanged by the preservation constraints, both NLL and BA-ECE improve relative to CDC. In particular, CMSap achieves

the lowest NLL with both pooling strategies, whereas CMSop yields the best BA-ECE. ACE also improves with order preservation, although the behavior of argmax preservation is less consistent across pooling strategies. Overall, on Cityscapes, preserving the decision map restores the pooled DSC while also improving most calibration metrics. On BraTS, decision-preserving variants achieve very close NLL and ECE values compared with CDC, while improving BA-ECE. ACE is more mixed and often slightly worse, especially for $\mathrm{CMS}_{\mathrm{op}}$. Thus, preserving the decision map does not induce a clear degradation of calibration performance. RoadSeg is the only dataset where a clear calibration–segmentation trade-off is observed. CDC achieves the best calibration across all reported metrics. This may be partly explained by the binary nature of the problem. When C=2, the identifiable degrees of freedom decrease from 6 for CDC to only 4 for the decision-preserving CMS variants (Table 2), making the preservation constraints comparatively more restrictive.

Overall, enforcing argmax or order preservation prevents unintended segmentation changes by construction, but its calibration cost is not universal. In our experiments, a clear trade-off is observed only in the low-class RoadSeg setting, whereas on BraTS and Cityscapes the decision-preserving variants achieve comparable or even better calibration performance. This suggests that, when the conditional affine calibrator remains sufficiently expressive, decision-preserving constraints do not inherently compromise calibration and may even act as a beneficial regularizer. Conversely, in binary or other low-class settings, the same constraints can reduce calibration flexibility and expose a more explicit calibration–segmentation trade-off.

4. Practical implications for choosing a calibration pipeline. When preservation of the deployed segmentation map is required by the downstream task, decision-preserving constraints are mandatory. When this requirement is relaxed, these constraints may still act as a beneficial regularizer for calibration, and constrained and unconstrained calibrators should therefore be systematically compared. In contrast, translation invariance is both a structural requirement for representation-consistent post-hoc calibration and, in our matched comparisons, an empirically beneficial constraint, which we therefore recommend by default.

Among TI pipelines, methods can be categorized along two dimensions: calibration flexibility and decision preservation. On both Cityscapes and BraTS, no single method consistently outperforms all others across all calibration metrics. Nevertheless, simple temperature-based methods (TS and ETS), which are also order-preserving, achieve excellent performance on Cityscapes. On RoadSeg, $\mathrm{LTS}(\log S(\bar{z}))$, which provides a more flexible TI and order-preserving alternative to TS and ETS, yields the strongest ECE, BA-ECE, and ACE. On BraTS, CDC and the CMS variants perform particularly well. Overall, no single calibrator is uniformly optimal across datasets and evaluation metrics. Reliability diagrams in Appendix J provide additional qualitative support for these dataset-specific observations.

5 Conclusion

We showed that post-hoc calibration for semantic segmentation can be gauge-dependent: softmax logits are only identifiable up to an additive shift, and common calibrators can produce different outputs for equivalent representations of the same predictive distribution. Motivated by this, we introduced representation-consistent calibration and derived a simple condition—translation invariance—that distinguishes well-defined calibrators (e.g., TS, DC) from representation-dependent ones (e.g., MS/VS and LTS). We also presented practical ways to enforce translation invariance, including canonicalizing inputs via log-softmax and introducing new TI variants, such as constrained matrix scaling (MS_c). We further investigated the mismatch between training and calibration objectives and found that likelihood-driven calibration can change argmax decisions and degrade segmentation quality. To analyze this effect, we introduced class-conditional affine calibrators and their argmax- and order-preserving counterparts, $\mathrm{CMS}_{\mathrm{ap}}$ and $\mathrm{CMS}_{\mathrm{op}}$, enabling matched comparisons that quantify the calibration–segmentation trade-off induced by decision preservation. Across natural-image and medical segmentation benchmarks, including under covariate shift, translation invariance emerges as both a structural requirement and an empirically beneficial constraint in our matched comparisons. Moreover, enforcing argmax or order preservation may induce an implicit regularization effect that can benefit calibration. Together, these results establish translation invariance and decision preservation as two fundamental design axes for post-hoc calibration in segmentation.

Broader Impact Statement

By identifying translation invariance and decision preservation as structural properties of post-hoc calibration, this work provides practical design principles for future calibration methods in dense prediction. Translation invariance, in particular, is a lightweight, architecture-agnostic constraint that can be readily incorporated into a wide range of pipelines by using log probabilities instead of logits as input to the calibrator. Beyond its theoretical motivation, our results suggest that translation invariance is a beneficial design choice in practice and should therefore be considered by default when designing post-hoc calibration methods.

More generally, calibration plays a crucial role in safety-critical applications such as medical image segmentation by making predictive confidence better reflect observed accuracy, thereby improving downstream decision-making in both human and automated settings. It should therefore be viewed as a tool for decision support rather than a substitute for expert judgment.

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The Cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Maria Correia de Verdier, Rachit Saluja, Louis Gagnon, Dominic LaBella, Ujjwall Baid, Nourel Hoda Tahon, Martha Foltyn-Dumitru, Jikai Zhang, Maram Alafif, Saif Baig, Ken Chang, Gennaro D'Anna, Lisa Deptula, Diviya Gupta, Muhammad Ammar Haider, Ali Hussain, Michael Iv, Marinos Kontzialis, Paul Manning, Farzan Moodi, Teresa Nunes, Aaron Simon, Nico Sollmann, David Vu, Mariam Aboian, and others. The 2024 Brain Tumor Segmentation (BraTS) Challenge: Glioma segmentation on post-treatment MRI. *arXiv preprint arXiv:2405.18368*, 2024.

Zhipeng Ding, Xu Han, Peirong Liu, and Marc Niethammer. Local Temperature Scaling for probability calibration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Bradley Efron and Robert Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, 1994.

Stanislav Fort, Huiyi Hu, and Balaji Lakshminarayanan. Deep ensembles: A loss landscape perspective. In *International Conference on Neural Information Processing Systems (NeurIPS)*, 2020.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pp. 1321–1330, 2017.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations (ICLR)*, 2019.

Fabian Isensee, Paul F. Jaeger, Simon A. A. Kohl, Jens Petersen, and Klaus H. Maier-Hein. nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature Methods*, 18(2): 203–211, February 2021. ISSN 1548-7105.

Kim-Celine Kahl, Carsten T Lüth, Maximilian Zenk, Klaus Maier-Hein, and Paul F Jaeger. ValUES: A framework for systematic validation of uncertainty estimation in semantic segmentation. In *International Conference on Learning Representations (ICLR)*, 2024.

Christoph Kamann and Carsten Rother. Benchmarking the robustness of semantic segmentation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8828–8838, 2020.

Mijoo Kim and Junseok Kwon. Uncertainty calibration with energy based instance-wise scaling in the wild dataset. In *Eur. Conf. Comput. Vis. (ECCV)*, 2024.

Meelis Kull, Miquel Perello-Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. Beyond temperature scaling: Obtaining well-calibrated multiclass probabilities with Dirichlet calibration. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Aviral Kumar, Sunita Sarawagi, and Ujjwal Jain. Trainable calibration measures for neural networks from kernel mean embeddings. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2805–2814. PMLR, 2018.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Wenhao Liang, Wei Zhang, Lin Yue, Miao Xu, Olaf Maennel, and Weitong Chen. We care each pixel: Calibrating medical segmentation models. In *Proceedings of the 28th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2025.

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:21464–21475, 2020.

Alireza Mehrtash, William M. Wells, Clare M. Tempany, Purang Abolmaesumi, and Tina Kapur. Confidence calibration and predictive uncertainty estimation for deep medical image segmentation. *IEEE Transactions on Medical Imaging (T-MI)*, 39(12):3868–3878, 2020.

Dimitrios Milios, Raffaello Camoriano, Pietro Michiardi, Lorenzo Rosasco, and Maurizio Filippone. Dirichlet-based gaussian processes for large-scale calibrated classification. *Advances in Neural Information Processing Systems (NeurIPS)*, 31, 2018.

Volodymyr Mnih. *Machine Learning for Aerial Image Labeling*. PhD thesis, University of Toronto, 2013.

Jishnu Mukhoti, Viveka Kulharia, Amartya Sanyal, Stuart Golodetz, Philip H. S. Torr, and Puneet K. Dokania. Calibrating deep neural networks using focal loss. In *Advances in Neural Information Processing Systems*, volume 33, pp. 15288–15299, 2020.

Rafael Müller, Simon Kornblith, and Geoffrey Hinton. When does label smoothing help? In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D. Sculley, Sebastian Nowozin, Joshua V. Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using Bayesian binning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 29(1), Feb. 2015.

Adam Paszke, Sam Gross, and Francisco Massa. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

John Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.

Rahul Rahaman and Alexandre H. Thiery. Uncertainty quantification and deep ensembles. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Amir Rahimi, Amirreza Shaban, Ching-An Cheng, Richard Hartley, and Byron Boots. Intra order-preserving functions for calibration of multi-class neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 13456–13467, 2020.

Axel-Jan Rousseau, Thijs Becker, Simon Appeltans, Matthew Blaschko, and Dirk Valkenborg. Post hoc calibration of medical segmentation models. *Discover Applied Sciences*, 7(3):180, 2025.

Carole H Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *International Workshop on Deep Learning in Medical Image Analysis*, pp. 240–248. Springer, 2017.

Pauli Virtanen, Ralf Gommers, and Travis E. Oliphant. SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17:261–272, 2020.

Dongdong Wang, Boqing Gong, and Liqiang Wang. On calibrating semantic segmentation models: Analyses and an algorithm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Jonathan Wenger, Hedvig Kjellström, and Rudolph Triebel. Non-parametric calibration for classification. In *International Conference on Artificial Intelligence and Statistics*, pp. 178–190. PMLR, 2020.

Bianca Zadrozny and Charles Elkan. Learning and making decisions when costs and probabilities are both unknown. In *Proceedings of the seventh ACM SIGKDD International Conference on Knowledge Discovery and Data mining*, pp. 204–213, 2001.

Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pp. 694–699, New York, NY, USA, 2002. Association for Computing Machinery. ISBN 158113567X.

Tal Zeevi, Eléonore V. Lieffrig, Lawrence H. Staib, and John A. Onofrey. Spatially-aware evaluation of segmentation uncertainty. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2025.

Maximilian Zenk, David Zimmerer, Fabian Isensee, Jeremias Traub, Tobias Norajitra, Paul F. Jäger, and Klaus Maier-Hein. Comparative benchmarking of failure detection methods in medical image segmentation: Unveiling the role of confidence aggregation. *Medical Image Analysis*, 101:103392, 2025. ISSN 1361-8415.

Jize Zhang, Bhavya Kailkhura, and T Yong-Jin Han. Mix-n-Match: Ensemble and compositional methods for uncertainty calibration in deep learning. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pp. 11117–11128. PMLR, 2020.

Appendix

This appendix contains the following items:

- **A. Additional Proofs**
- **B. Implementation and Training Details**
- **C. Spatial Calibration Assessment**
- **D. Spatial Patterns of Label Flips**
- **E. Sensitivity to Ensemble Size**
- **F. Sensitivity to Calibration Set Size**
- **G. Sensitivity to Binning**
- **H. Calibration under Corruption-Based Covariate Shift**
- **I. Complete Quantitative Results with Uncertainty Intervals**
- **J. Reliability Diagrams**

We will publicly release the calibration code, hyperparameter-search configurations, evaluation scripts, and figure-generation commands.

A Additional Proofs

A.1 Translation Invariance of Local Temperature Scaling

We analyze when Local Temperature Scaling (LTS) is translation-invariant. Let

$$g_\theta^{\mathrm{LTS}}(\mathbf{z}(v)) = S\left(\frac{\mathbf{z}(v)}{\tau_\theta(v;\mathbf{Z},\xi)}\right),$$

where $\tau_\theta(v;\mathbf{Z},\xi) > 0$ is a temperature predicted from the full logit field $\mathbf{Z}$ and optional features ξ. Consider a spatially varying additive shift $c(v)$ from which we derive a full logit field $\mathbf{Z}'$ with $\mathbf{z}'(v) = \mathbf{z}(v) + c(v)\mathbf{1}$. If the temperature predictor is itself invariant to such shifts, we have: $\tau_\theta(v;\mathbf{Z}',\xi) = \tau_\theta(v;\mathbf{Z},\xi)$.

For simplicity, and since this case is of limited practical relevance, we restrict ourselves to non-degenerate logits, i.e., $\mathbf{z}(v) \notin \mathrm{span}\{\mathbf{1}\}$. In the degenerate case, where all class logits are equal, the softmax output is uniform and independent of the temperature, so any violation of shift invariance of τ_θ has no effect on the output of LTS.

Since softmax is invariant to additive constants, LTS is translation-invariant if the transformed logits differ only by an additive constant, i.e.,

$$\frac{\mathbf{z}'(v)}{\tau_\theta(v;\mathbf{Z}',\xi)} - \frac{\mathbf{z}(v)}{\tau_\theta(v;\mathbf{Z},\xi)} \in \mathrm{span}\{\mathbf{1}\}.$$

If the temperature predictor is itself invariant to such shifts, we get:

$$\frac{\mathbf{z}'(v)}{\tau_\theta(v;\mathbf{Z}',\xi)} - \frac{\mathbf{z}(v)}{\tau_\theta(v;\mathbf{Z},\xi)} = \frac{c(v)}{\tau_\theta(v;\mathbf{Z},\xi)}\mathbf{1} \in \mathrm{span}\{\mathbf{1}\}.$$

Thus, LTS is translation-invariant.

Conversely, suppose that for some location v, $\tau_\theta(v;\mathbf{Z}',\xi) \neq \tau_\theta(v;\mathbf{Z},\xi)$. Let

$$\lambda' = \frac{1}{\tau_\theta(v;\mathbf{Z}',\xi)}, \qquad \lambda = \frac{1}{\tau_\theta(v;\mathbf{Z},\xi)}.$$

Then

$$\frac{\mathbf{z}'(v)}{\tau_\theta(v;\mathbf{Z}',\xi)} - \frac{\mathbf{z}(v)}{\tau_\theta(v;\mathbf{Z},\xi)} = (\lambda' - \lambda)\mathbf{z}(v) + \lambda' c(v)\mathbf{1}. \tag{13}$$

Since $\mathbf{z}(v) \notin \mathrm{span}\{\mathbf{1}\}$ and since $\lambda' \neq \lambda$, the vector of Eq. 13 is not in $\mathrm{span}\{\mathbf{1}\}$. Therefore, under the hypothesis that $\mathbf{z}(v) \notin \mathrm{span}\{\mathbf{1}\}\forall v$, LTS is translation-invariant if and only if the temperature predictor is invariant to additive logit shifts.

Note that the original LTS formulation, whose temperature network takes raw logits as input, does not guarantee translation invariance. In contrast, feeding the temperature network with canonical log-probabilities $\log S(\mathbf{Z})$ makes τ_θ invariant to additive shifts and yields a translation-invariant LTS pipeline.

A.2 Order- and Argmax-Preserving Affine Calibrators Collapse to Temperature Scaling

We first consider the bias-free case and write the calibrated probabilities as $g(\mathbf{z}) = S(\mathbf{W}\mathbf{z})$, where S is the softmax function. For each class $i \in \{1,\dots,C\}$, we define the cone $\Psi_1^{(i)} = \{\mathbf{z} \in \mathbb{R}^C \mid \mathbf{z}_i \geq \mathbf{z}_k,\ \forall k \neq i\}$, i.e., the set of logits whose maximum is attained at class i. A matrix $\mathbf{W}$ preserves the argmax if and only if each cone $\Psi_1^{(i)}$ is invariant under $\mathbf{W}$, i.e., $\mathbf{z} \in \Psi_1^{(i)} \Rightarrow \mathbf{W}\mathbf{z} \in \Psi_1^{(i)}$.

As shown in Section 3.2.1, the invariance of $\Psi_1^{(1)}$ (denoted Ψ_1 therein) under $\mathbf{W}$ is equivalent to the existence of a matrix $\mathbf{W}'$ such that: $\mathbf{W} = \mathbf{G}^{-1}\mathbf{W}'\mathbf{G}$, with $\mathbf{W}' = \begin{pmatrix} a & \mathbf{r}^\top \\ \mathbf{0} & \mathbf{A} \end{pmatrix}$, where $a \in \mathbb{R}$, $\mathbf{r} \in \mathbb{R}^{C-1}$, and $\mathbf{A} \in \mathbb{R}_+^{(C-1)\times(C-1)}$, and $\mathbf{G}$ defined in Eq. 12. In the following, we denote by $\mathbf{W}^{(i)}$ a matrix leaving $\Psi_1^{(i)}$ invariant.

For simplicity, we illustrate the structure of $\mathbf{W}^{(1)}$ for $C = 4$ and proceed with the proof without loss of generality in this setting. A direct computation yields:

$$\mathbf{W}^{(1)} = \begin{pmatrix} s & -r_1 & -r_2 & -r_3 \\ s-\rho_2 & a_{11}-r_1 & a_{12}-r_2 & a_{13}-r_3 \\ s-\rho_3 & a_{21}-r_1 & a_{22}-r_2 & a_{23}-r_3 \\ s-\rho_4 & a_{31}-r_1 & a_{32}-r_2 & a_{33}-r_3 \end{pmatrix}, \quad s = a + \sum_{i=1}^{3} r_i, \quad \rho_{i+1} = \sum_{j=1}^{3} a_{ij}.$$

The structure of $\mathbf{W}^{(1)}$ is fully determined by the following two constraints: (i) all rows have identical row sums; (ii) in each column $j > 1$, the first entry is a minimal element. These constraints in turn imply that the first entry of the first column is a maximal element.

The characterization of $\mathbf{W}^{(i)}$ can be obtained from that of $\mathbf{W}^{(1)}$ by simultaneously permuting the i-th row and column with the first row and column. For instance, exchanging rows and columns 1 and 2 yields

$$\mathbf{W}^{(2)} = \begin{pmatrix} a_{11}-r_1 & s-\rho_2 & a_{12}-r_2 & a_{13}-r_3 \\ -r_1 & s & -r_2 & -r_3 \\ a_{21}-r_1 & s-\rho_3 & a_{22}-r_2 & a_{23}-r_3 \\ a_{31}-r_1 & s-\rho_4 & a_{32}-r_2 & a_{33}-r_3 \end{pmatrix}.$$

By construction, $\Psi_1^{(2)}$ is invariant under $\mathbf{W}^{(2)}$. We now require consistency with the invariance of $\Psi_1^{(1)}$, i.e., that the ordering constraints derived for $\mathbf{W}^{(1)}$ remain valid under this permutation, while the row-sum condition is already preserved. We obtain in particular $a_{12} = a_{13} = 0$, together with $\rho_2 \geq \rho_3$ and $\rho_2 \geq \rho_4$. The corresponding constraints on the first column, namely $a_{11} \geq a_{21}$ and $a_{11} \geq a_{31}$, become redundant once the constraints from $\Psi_1^{(3)}$ and $\Psi_1^{(4)}$ are incorporated, as these imply $a_{21} = a_{31} = 0$.

Indeed, repeating the argument for $\Psi_1^{(3)}$ and $\Psi_1^{(4)}$ yields $a_{21} = a_{23} = 0$, $\rho_3 \geq \rho_2$, $\rho_3 \geq \rho_4$, and $a_{31} = a_{32} = 0$, $\rho_4 \geq \rho_2$, $\rho_4 \geq \rho_3$. Combining all these constraints, we conclude that $a_{ij} = 0$ for all $i \neq j$, and $\rho_2 = \rho_3 = \rho_4 = \lambda > 0$. Finally, argmax-preserving matrices take the form

$$\mathbf{W} = \begin{pmatrix} s & -r_1 & -r_2 & -r_3 \\ s-\lambda & \lambda-r_1 & -r_2 & -r_3 \\ s-\lambda & -r_1 & \lambda-r_2 & -r_3 \\ s-\lambda & -r_1 & -r_2 & \lambda-r_3 \end{pmatrix}. \tag{14}$$

This can be expressed as $\mathbf{W} = \lambda\mathbf{I} + \mathbf{1}\mathbf{c}^\top$, where $\lambda > 0$ and $\mathbf{I}$ denotes the identity matrix. The vector $\mathbf{c} \in \mathbb{R}^C$ is defined by $c_1 = s - \lambda$ and $c_{i+1} = -r_i$ for $i \in \{1, 2, 3\}$. In particular, $\mathbf{c}$ is arbitrary since both $s \in \mathbb{R}$ (corresponding to the free parameter a in $\mathbf{W}'$) and $\mathbf{r} \in \mathbb{R}^{C-1}$ are unrestricted. In the general case ($C \geq 2$), the same reasoning applies within each of the C cones.

We now consider the affine case $g(\mathbf{z}) = S(\mathbf{W}\mathbf{z} + \mathbf{b})$. Since the argmax cones must remain invariant under translation, the bias vector must belong to all cones simultaneously, which implies $\mathbf{b} = k\mathbf{1}$ ($k \in \mathbb{R}$).

Finally, using the translation invariance of the softmax, $S(\mathbf{x} + k\mathbf{1}) = S(\mathbf{x})$, we obtain

$$S(\mathbf{W}\mathbf{z} + \mathbf{b}) = S(\lambda\mathbf{z} + \mathbf{1}\mathbf{c}^\top\mathbf{z} + k\mathbf{1}) = S(\lambda\mathbf{z}) = S\left(\frac{\mathbf{z}}{T}\right), \qquad T = \frac{1}{\lambda}.$$

Therefore, MS calibrators that preserve the argmax reduce exactly to TS. Since TS is also order preserving, the same conclusion holds under order preservation.

B Implementation and Training Details

Additional implementation details. All models are implemented in PyTorch (Paszke et al., 2019). The training loss is the voxel-wise cross-entropy (negative log-likelihood) for all methods, with the following

Dirichlet-style regularizer described in Kull et al. (2019) for affine calibrators (MS, MS_c, DC) :

$$\mathcal{R}(W, b) = \lambda \frac{1}{C(C-1)} \sum_{i \neq j} W_{ij}^2 + \mu \frac{1}{C} \sum_{j=1}^{C} b_j^2, \tag{15}$$

which penalizes off-diagonal weights and large biases. For CMS, the regularization is applied for each one of the C models. For LTS, ℓ_2 regularization is applied, following Ding et al. (2021).

Hyperparameters are optimized with Optuna (Akiba et al., 2019) using a TPE sampler and a budget of 20 trials per learned calibrator, selecting configurations by validation NLL on the validation split of each calibration repeat. The search space is defined as follows:

- Learning rate (LR): log-uniform in $[10^{-4}, 3 \times 10^{-2}]$,
- Early stopping patience (P): in $\{10, 20, 50\}$,
- Regularization strengths λ, μ: mixture of exact zero and log-uniform in $[10^{-6}, 10^{-1}]$,
- Weight decay α for ℓ_2 regularization (LTS only): mixture of exact zero and log-uniform in $[10^{-6}, 10^{-2}]$.

Optimization uses Adam, early stopping on validation NLL, and `ReduceLROnPlateau` scheduling. The C class-wise experts of each class-conditional affine calibrator share the same hyperparameters. All calibrators are parameterized to represent identity (absence of calibration) at initialization.

Local Temperature Scaling (LTS). Following Ding et al. (2021), the temperature predictor τ_θ follows a tree-style gated convolutional design, combining multiple candidate temperature maps with learned gates. Positivity is enforced with a softplus transform, and convolutions are applied in-plane with kernel size 5 and dilation 2.

C Spatial Calibration Assessment

We compute the boundary-aware expected calibration error (BA-ECE) following Zeevi et al. (2025). The main idea is to assess calibration in regions defined by their distance to the ground-truth boundary, so as to place more emphasis on locations where segmentation errors are typically more critical.

More precisely, let $d(v)$ denote the unsigned Euclidean distance (in voxels) from voxel v to the ground-truth boundary, with $d(v) = 0$ on the boundary itself. We partition the image into distance bands

$$D_k = \{v \mid d(v) \in \Delta_k\},$$

using the default intervals: $\Delta_1 = [0, 3), \quad \Delta_2 = [3, 7), \quad \Delta_3 = [7, 15), \quad \Delta_4 = [15, +\infty)$.

BA-ECE is then computed over these regions as in Zeevi et al. (2025), using inverse-distance weighting so that regions closer to the boundary contribute more strongly. Empty bands are ignored.

D Spatial Patterns of Label Flips

Calibration may change the predicted class if it is not argmax-preserving. $\mathrm{MS}(\bar{z})$ is one such pipeline that may flip labels and alter the segmentation. Figure 4 provides illustrative examples on BraTS and Cityscapes datasets, showing that label flips are not uniformly distributed: they concentrate primarily along tumor and semantic boundaries. This occurs because likelihood-based calibration can locally alter the class ordering when margins are small, leading to spatially structured label changes.

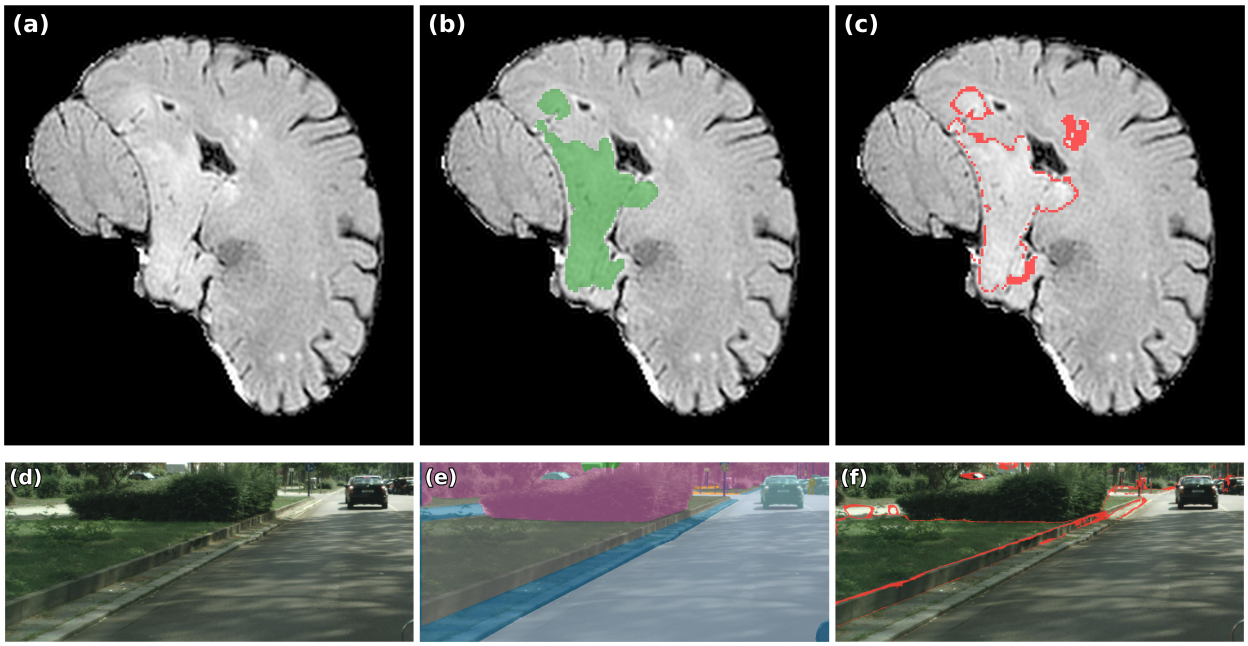

Figure 4: Spatial patterns of label flips under MS($\bar{\mathbf{z}}$) on BraTS and Cityscapes. Top row: BraTS example with (a) MRI slice, (b) ground-truth foreground mask (green), and (c) flipped voxels after calibration (red). Bottom row: Cityscapes example with (d) input image, (e) ground-truth semantic segmentation overlay, and (f) flipped pixels after calibration (red). The Cityscapes view is cropped for visualization. In both cases, flips are concentrated near semantic boundaries, indicating that non–argmax-preserving calibration induces structured local label changes rather than spatially uniform perturbations.

Dataset	n	$M = 1$	$M = 2$	$M = 3$	$M = 4$	$M = 5$
RoadSeg	243	0.97 [0.65, 0.96, 1.21]	1.04 [0.70, 1.02, 1.30]	1.09 [0.72, 1.08, 1.35]	1.06 [0.71, 1.05, 1.31]	1.08 [0.73, 1.06, 1.34]
BraTS	271	8.41 [8.21, 8.33, 8.52]	8.31 [8.11, 8.23, 8.41]	8.99 [8.78, 8.91, 9.09]	8.86 [8.65, 8.78, 8.96]	8.77 [8.56, 8.69, 8.87]
Cityscapes	500	2.71 [2.52, 2.66, 2.87]	2.67 [2.46, 2.60, 2.83]	2.56 [2.32, 2.47, 2.71]	2.56 [2.32, 2.48, 2.73]	2.54 [2.30, 2.46, 2.71]

Table 5: **Spatial variability of pooled-logit offsets across ensemble sizes.** For each $M \in \{1, 2, 3, 4, 5\}$, we report $\mathbb{E}[\mathrm{Std}_v(F_{\mathrm{pool}}(v))]$. Main values are means across held-out cases; brackets show $[p_{25}, p_{50}, p_{75}]$. Offset variability remains substantial for all ensemble sizes, motivating the use of translation-invariant calibration.

E Sensitivity to Ensemble Size

Table 5 reports $\mathbb{E}[\mathrm{Std}_v(F_{\mathrm{pool}}(v))]$ for pooled ensembles of size $M \in \{1, 2, 3, 4, 5\}$. The statistic varies primarily by dataset (RoadSeg lowest, Cityscapes intermediate, BraTS highest) and varies only slightly with M. Importantly, substantial spatial variation is present for every tested ensemble size, including small ensembles ($M = 1, 2$), indicating that the offset variability motivating TI calibration is not an artifact of using $M = 5$.

F Sensitivity to Calibration Set Size

We vary the number of held-out samples used to fit the post-hoc calibrator: $n \in \{5, 25, 50, 75, 99\}$ for Cityscapes and $n \in \{3, 14, 27, 41, 50\}$ for BraTS. The test split is kept fixed throughout; increasing n only changes the calibration/validation partition of the non-test pool. Performance is reported as the difference

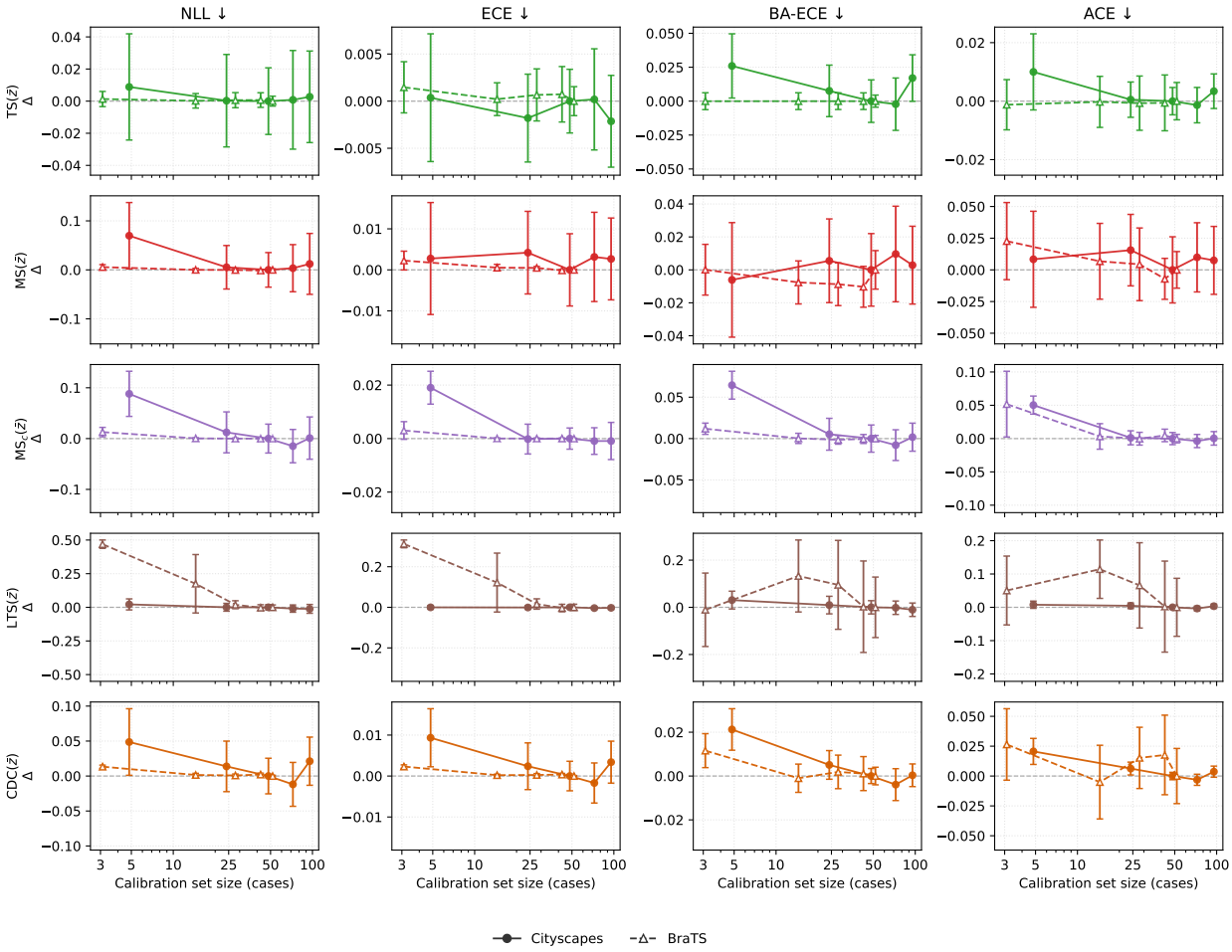

Figure 5: **Calibration set size sensitivity.** Effect of calibration set size n on NLL, ECE, BA-ECE, and ACE for Cityscapes and BraTS, reported as Δ relative to the $n = 50$ reference configuration used in the main experiments.

(Δ) relative to the reference configuration used in our main experiments, corresponding to $n = 50$ samples for both datasets.

For computational efficiency, we consider a representative subset of calibrators covering different model capacities and constraints: TS($\bar{z}$), MS($\bar{z}$), MS$_c$($\bar{z}$), LTS($\bar{z}$), and CDC($\bar{z}$).

Fig. 5 shows that performance is generally stable around the reference budget of 50 calibration samples. The largest variability occurs in the low-data regime, where the more expressive calibrators can exhibit increased NLL or calibration error and wider confidence intervals. Differences between moderate and larger calibration sets are small relative to the uncertainty bands, supporting our use of $n = 50$ as a pragmatic trade-off between stable calibration and preserving held-out data.

G Sensitivity to Binning

Expected Calibration Error (ECE) (Guo et al., 2017) and ACE (Kahl et al., 2024) both depend on the discretization of the confidence axis. To ensure our conclusions are not driven by a particular binning choice, we recompute ECE for $B \in \{20, 30, 40, 50\}$ bins and ACE for $B \in \{15, 20, 30, 40\}$ bins across datasets and methods (Fig. 6). Following Appendix F, we restrict this sensitivity analysis to a representative subset of calibrators. Across datasets, changing the number of bins produces only small changes in ECE for

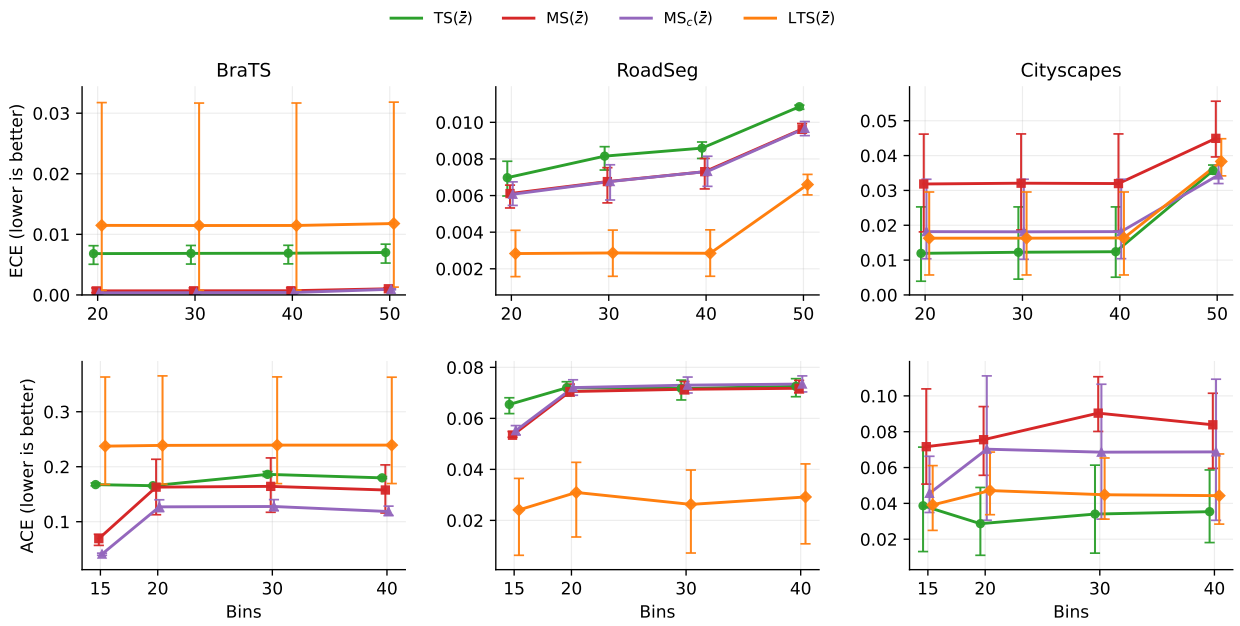

Figure 6: **Sensitivity of ECE and ACE to binning.** We report, for each dataset, ECE with $B \in \{20, 30, 40, 50\}$ and ACE for $B \in \{15, 20, 30, 40\}$. Error bars show 95% bootstrap confidence intervals for ECE and $[min, max]$ over repeats for ACE. The ranking of methods is stable across bin counts, indicating that our conclusions are not driven by a particular choice of ECE/ACE discretization.

most methods. ACE is more sensitive in absolute value because it averages calibration gaps uniformly over non-empty bins; increasing the number of bins can therefore amplify sparsely populated confidence regions. The range includes the two settings used in the main experiments: $B = 15$ for ACE, chosen as a compromise between resolution, stability, and readable reliability diagrams, and $B = 50$ for ECE. The qualitative comparisons used in the paper do not change with binning.

H Calibration under Corruption-Based Covariate Shift

To assess robustness under covariate shift, we evaluate the Cityscapes calibrators from the main paper on corrupted test images. All calibrators are fitted on the clean in-distribution Cityscapes calibration set, exactly as in the main experiments, and are then evaluated without refitting on the corrupted test images. We focus on Cityscapes because standardized corruption protocols are available; RoadSeg shows limited sensitivity to calibration choices, and realistic covariate shifts for BraTS are beyond the scope of this work.

We generate two corrupted Cityscapes test sets with *mild* and *strong* severities, following the common-corruptions paradigm of ImageNet-C (Hendrycks & Dietterich, 2019) and its adaptation to Cityscapes (Kamann & Rother, 2020). For each severity level, each image is transformed exactly once by sampling a perturbation type from a fixed perturbation bank using a deterministic hash of the filename and a fixed random seed. This yields a one-to-one correspondence between corrupted images and ground-truth labels and ensures reproducibility. We restrict perturbations to photometric transformations in order to induce controlled covariate shift while preserving pixel-level label alignment. The perturbation bank consists of Gaussian noise, Gaussian blur, brightness scaling, JPEG compression, and fog/haze. No geometric transformations are applied (Table 6).

Figure 7 shows representative corrupted examples under *strong* severity, and Table 7 reports the resulting calibration performance under corruption-based covariate shift. We discuss the main trends below.

Perturbation	Mild	Strong
Gaussian noise	$\sigma = 0.03$	$\sigma = 0.10$
Gaussian blur	$k = 5,\ \sigma = 1.0$	$k = 13,\ \sigma = 3.0$
Brightness shift	factor = 1.2	factor = 1.8
JPEG compression	quality = 60	quality = 10
Fog/haze	$\alpha = 0.1$	$\alpha = 0.5$

Table 6: Photometric corruption settings used to induce test-time covariate shift on Cityscapes. For each corruption type, we define two severity levels, *mild* and *strong*. All corruptions are label-preserving and are applied only at evaluation time.

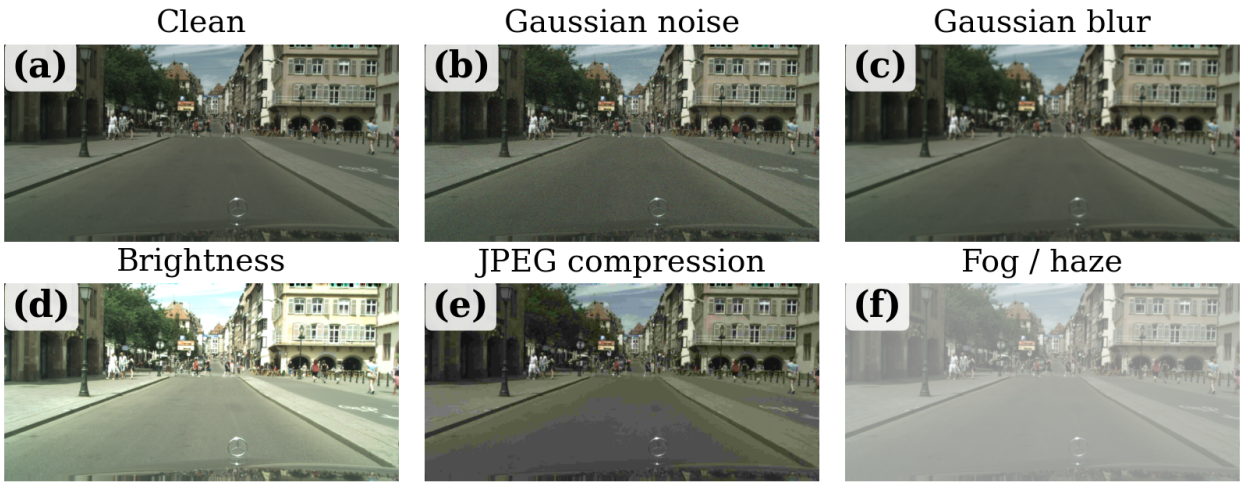

Figure 7: **Cityscapes corruption types.** Example from the test set under *strong* severity corruption: (**a**) in-distribution (clean) image; (**b**) Gaussian noise; (**c**) Gaussian blur; (**d**) brightness shift; (**e**) JPEG compression; (**f**) fog/haze. All variants preserve pixel-level correspondence with the original labels.

1. Ensembles improve segmentation but do not reliably fix calibration. (Table 7(a)) The same trends observed in-distribution persist under covariate shift. Probability pooling improves DSC over the single model (mild: 77.2 vs. 70.2; strong: 68.2 vs. 58.3), while logit pooling again achieves the highest segmentation accuracy (mild: 78.4; strong: 70.2). Pooling reduces some calibration errors under shift, especially for $\bar{p}$, but the effect remains pooling- and metric-dependent; ensembling alone therefore does not reliably correct miscalibration under covariate shift.

2. TI pipelines appear more robust under shift. (Table 7(b)) The in-distribution trends largely persist under covariate shift, but the gap between offset-sensitive and TI pipelines becomes more pronounced, especially for NLL and ECE. In the matched MS comparison, the unconstrained $\mathrm{MS}(\bar{z})$ consistently underperforms its two TI counterparts $\mathrm{MS}_c(\bar{z})$ and $\mathrm{DC}(\bar{z})$. A similar pattern is observed for LTS: using the TI input representation $\mathrm{LTS}(\log S(\bar{z}))$ improves NLL, ECE, and BA-ECE at both corruption severities. ACE is slightly worse under mild corruption (0.078 vs. 0.074), but better under strong corruption (0.090 vs. 0.095). Overall, these results suggest that TI pipelines are more robust under shift, with improvements in most comparisons, although the gains are not uniform across all metrics and severities.

3. For Cityscapes, decision preservation improves segmentation without compromising calibration. (Table 7(d)) The pattern observed in-distribution for Cityscapes becomes more pronounced under strong shift: non-argmax-preserving pipelines can suffer large DSC drops. Under strong corruption, $\mathrm{CDC}(\bar{p})$ drops to 57.7 DSC with an 11.1% flip rate, whereas $\mathrm{CMS}_{\mathrm{ap}}(\bar{p})$ preserves the pooled segmentation and maintains 68.2 DSC, with improvements in most calibration metrics. A similar pattern holds for logit pooling. Overall, as in the in-distribution setting, enforcing decision preservation improves segmentation performance and also leads to improvements in most calibration metrics.

Pipeline	Cityscapes (*mild* corruption)						Cityscapes (*strong* corruption)					
	NLL↓	**DSC↑**	**ECE↓**	**BA-ECE↓**	**ACE↓**	**Flip↓**	**NLL↓**	**DSC↑**	**ECE↓**	**BA-ECE↓**	**ACE↓**	**Flip↓**
(a) Uncalibrated baselines												
p_0	0.539	70.2	0.070	0.195	0.132	—	1.173	58.3	0.153	0.273	0.179	—
$\bar{p}$	0.406	77.2	0.046	0.142	0.090	—	0.756	68.2	0.087	0.190	0.112	—
$\bar{z}$	0.467	**78.4**	0.062	0.180	0.128	—	0.893	**70.2**	0.124	0.246	0.164	—
(b) Translation invariance: MS vs. MS_c and DC												
$\mathrm{MS}(\bar{z})^{\times}$	0.445	72.3	0.048	0.163	0.088	4.4	0.949	64.4	0.110	0.220	0.129	8.1
$\mathrm{DC}(\bar{z})$	0.352	72.8	0.037	0.106	0.071	**2.7**	0.757	63.6	0.082	0.160	0.105	**5.6**
$\mathrm{MS}_c(\bar{z})$	0.359	72.1	0.037	0.106	0.070	2.9	0.764	63.6	0.090	0.166	0.108	6.4
(c) Translation invariance: LTS with logits vs. log-probabilities												
$\mathrm{LTS}(\bar{z})^{\times}$	0.372	**78.4**	0.040	0.083	0.074	—	0.696	**70.2**	0.072	0.131	0.095	—
$\mathrm{LTS}(\log S(\bar{z}))$	0.355	**78.4**	0.037	0.077	0.078	—	0.656	**70.2**	0.063	0.115	0.090	—
(d) Decision preservation: CDC vs CMS												
$\mathrm{CDC}(\bar{p})$	0.364	68.2	**0.035**	0.113	0.073	5.8	0.730	57.7	0.076	0.156	0.099	11.1
$\mathrm{CMS}_{ap}(\bar{p})$	**0.341**	77.2	0.040	0.091	0.036	—	0.702	68.2	0.070	0.133	0.061	—
$\mathrm{CMS}_{op}(\bar{p})$	0.355	77.2	0.040	0.075	**0.030**	—	0.658	68.2	0.065	0.113	**0.038**	—
$\mathrm{CDC}(\bar{z})$	0.369	73.8	0.037	0.123	0.078	5.4	0.773	65.1	0.085	0.175	0.114	9.7
$\mathrm{CMS}_{ap}(\bar{z})$	0.354	**78.4**	0.042	0.099	0.054	—	0.742	**70.2**	0.080	0.153	0.118	—
$\mathrm{CMS}_{op}(\bar{z})$	0.368	**78.4**	0.039	0.086	0.037	—	0.698	**70.2**	0.073	0.138	0.084	—
(e) Standard temperature-based calibrators (order-preserving, TI): TS/ETS												
$\mathrm{TS}(\bar{p})$	0.356	77.2	0.039	0.073	0.078	—	**0.653**	68.2	**0.062**	**0.109**	**0.089**	—
$\mathrm{TS}(\bar{z})$	0.368	**78.4**	0.038	0.086	0.071	—	0.686	**70.2**	0.071	0.136	0.093	—
$\mathrm{ETS}(\bar{p})$	0.356	77.2	0.038	0.075	0.077	—	**0.653**	68.2	**0.062**	**0.109**	**0.089**	—
$\mathrm{ETS}(\bar{z})$	0.368	**78.4**	0.038	0.086	0.071	—	0.686	**70.2**	0.071	0.136	0.093	—
(f) Standard affine calibrator (non order-preserving, TI): DC												
$\mathrm{DC}(\bar{p})$	0.349	66.6	0.036	0.103	**0.069**	3.9	0.736	56.9	0.079	0.152	0.098	8.5
$\mathrm{DC}(\bar{z})$	0.352	72.8	0.037	0.106	0.071	**2.7**	0.757	63.6	0.082	0.160	0.105	**5.6**

Table 7: **Calibration on Cityscapes under covariate shift.** Numbered and lightly colored row blocks follow Table 4: blue for translation-invariance comparisons, green for decision-preservation comparisons, and beige for standard calibrators. DSC and Flip are reported in %. Flip is the fraction of pixels whose predicted class changes relative to the corresponding uncalibrated pooled prediction; — indicates not applicable. $\times$ denotes pipelines that are not translation-invariant. **Bold** indicates the best value per corruption severity and metric.

I Complete Quantitative Results with Uncertainty Intervals

Table 8 reports the quantitative results for all datasets, pipelines, and metrics, together with uncertainty intervals. The point estimates are identical to those reported in the main paper. Except for ACE, intervals are 95% percentile confidence intervals (CI) computed using a hierarchical bootstrap over the $R = 3$ repeats considered in the paper, resampling both repeats and cases within each repeat, following standard bootstrap methodology (Efron & Tibshirani, 1994). For ACE, the table reports the min–max range across repeat-level ACE values under the convention described in the evaluation protocol. The procedure was implemented using SciPy (Virtanen et al., 2020).

RoadSeg Mnih (2013)						
Pipeline	**NLL↓**	**DSC (%)↑**	**ECE↓**	**BA-ECE↓**	**ACE↓**	**Flip (%)↓**
(a) Uncalibrated baselines						
p_0	0.091 [0.084, 0.099]	76.3 [75.2, 77.4]	0.017 [0.015, 0.018]	0.094 [0.090, 0.098]	0.119 [0.118, 0.119]	—
$\bar{p}$	0.085 [0.079, 0.092]	**76.7 [75.6, 77.7]**	0.015 [0.014, 0.016]	0.087 [0.084, 0.091]	0.105 [0.105, 0.106]	—
$\bar{z}$	0.088 [0.083, 0.092]	**76.7 [76.2, 77.3]**	0.016 [0.015, 0.017]	0.091 [0.088, 0.093]	0.117 [0.116, 0.117]	—
(b) Translation invariance: MS vs. MS_c and DC						
$\mathrm{MS}(\bar{z})^{\times}$	0.068 [0.063, 0.073]	76.5 [75.4, 77.6]	0.010 [0.009, 0.010]	0.056 [0.052, 0.061]	0.053 [0.053, 0.055]	0.2 [0.2, 0.3]
$\mathrm{DC}(\bar{z})$	0.067 [0.063, 0.072]	**76.7 [75.6, 77.7]**	0.008 [0.008, 0.009]	0.047 [0.042, 0.052]	0.040 [0.038, 0.042]	**0.1 [0.0, 0.1]**
$\mathrm{MS}_c(\bar{z})$	0.068 [0.063, 0.073]	76.5 [75.3, 77.5]	0.010 [0.009, 0.010]	0.057 [0.052, 0.062]	0.055 [0.053, 0.057]	0.2 [0.2, 0.3]
(c) Translation invariance: LTS with logits vs. log-probabilities						
$\mathrm{LTS}(\bar{z})^{\times}$	0.067 [0.064, 0.070]	**76.7 [76.2, 77.3]**	0.004 [0.003, 0.005]	0.047 [0.038, 0.056]	0.024 [0.006, 0.036]	—
$\mathrm{LTS}(\log S(\bar{z}))$	0.066 [0.063, 0.068]	**76.7 [76.2, 77.3]**	**0.002 [0.001, 0.002]**	**0.040 [0.040, 0.041]**	**0.007 [0.002, 0.009]**	—
(d) Decision preservation: CDC vs CMS						
$\mathrm{CDC}(\bar{p})$	**0.065 [0.060, 0.071]**	75.7 [74.5, 76.8]	0.006 [0.006, 0.007]	0.043 [0.037, 0.049]	0.013 [0.012, 0.015]	0.5 [0.5, 0.6]
$\mathrm{CMS}_{ap}(\bar{p})$	0.067 [0.062, 0.072]	**76.7 [75.7, 77.7]**	0.007 [0.006, 0.008]	0.046 [0.041, 0.050]	0.036 [0.034, 0.038]	—
$\mathrm{CMS}_{op}(\bar{p})$	0.067 [0.062, 0.072]	**76.7 [75.7, 77.7]**	0.007 [0.006, 0.008]	0.046 [0.042, 0.051]	0.037 [0.035, 0.039]	—
$\mathrm{CDC}(\bar{z})$	0.066 [0.061, 0.071]	75.7 [74.6, 76.8]	0.006 [0.006, 0.007]	0.041 [0.035, 0.045]	0.012 [0.009, 0.014]	0.6 [0.5, 0.6]
$\mathrm{CMS}_{ap}(\bar{z})$	0.067 [0.062, 0.072]	**76.7 [75.7, 77.9]**	0.008 [0.007, 0.008]	0.047 [0.042, 0.052]	0.043 [0.040, 0.046]	—
$\mathrm{CMS}_{op}(\bar{z})$	0.067 [0.063, 0.072]	**76.7 [75.6, 77.7]**	0.007 [0.007, 0.008]	0.047 [0.043, 0.052]	0.043 [0.041, 0.044]	—
(e) Standard temperature-based calibrators (order-preserving, TI): TS/ETS						
$\mathrm{TS}(\bar{p})$	0.068 [0.065, 0.070]	**76.7 [75.6, 77.7]**	0.008 [0.008, 0.008]	0.059 [0.056, 0.063]	0.056 [0.053, 0.058]	—
$\mathrm{TS}(\bar{z})$	0.068 [0.065, 0.071]	**76.7 [76.2, 77.3]**	0.009 [0.008, 0.009]	0.061 [0.057, 0.065]	0.065 [0.062, 0.068]	—
$\mathrm{ETS}(\bar{p})$	0.068 [0.065, 0.070]	**76.7 [75.6, 77.7]**	0.008 [0.008, 0.009]	0.059 [0.056, 0.063]	0.055 [0.054, 0.058]	—
$\mathrm{ETS}(\bar{z})$	0.068 [0.066, 0.071]	**76.7 [76.2, 77.3]**	0.009 [0.008, 0.009]	0.061 [0.057, 0.065]	0.065 [0.063, 0.068]	—
(f) Standard affine calibrator (non order-preserving, TI): DC						
$\mathrm{DC}(\bar{p})$	0.067 [0.062, 0.072]	76.6 [75.5, 77.7]	0.008 [0.007, 0.008]	0.046 [0.042, 0.052]	0.033 [0.031, 0.034]	**0.1 [0.1, 0.1]**
$\mathrm{DC}(\bar{z})$	0.067 [0.063, 0.072]	**76.7 [75.6, 77.7]**	0.008 [0.008, 0.009]	0.047 [0.042, 0.052]	0.040 [0.038, 0.042]	**0.1 [0.0, 0.1]**

BraTS 2024 GLI Correia de Verdier et al. (2024)						
Pipeline	**NLL↓**	**DSC (%)↑**	**ECE↓**	**BA-ECE↓**	**ACE↓**	**Flip (%)↓**
(a) Uncalibrated baselines						
p_0	0.524 [0.520, 0.527]	75.2 [73.6, 76.8]	0.340 [0.338, 0.342]	0.049 [0.045, 0.053]	0.112 [0.110, 0.114]	—
$\bar{p}$	0.522 [0.518, 0.526]	76.0 [74.5, 77.5]	0.340 [0.337, 0.342]	0.041 [0.037, 0.045]	0.087 [0.084, 0.092]	—
$\bar{z}$	0.523 [0.522, 0.524]	**76.9 [76.0, 77.4]**	0.340 [0.339, 0.341]	0.046 [0.043, 0.048]	0.096 [0.088, 0.100]	—
(b) Translation invariance: MS vs. MS_c and DC						
$\mathrm{MS}(\bar{z})^{\times}$	0.008 [0.006, 0.011]	68.4 [63.0, 73.1]	0.001 [0.001, 0.001]	0.043 [0.032, 0.057]	0.069 [0.057, 0.077]	**0.1 [0.1, 0.2]**
$\mathrm{DC}(\bar{z})$	0.008 [0.007, 0.010]	67.6 [57.2, 74.5]	0.001 [0.001, 0.001]	0.040 [0.032, 0.051]	0.065 [0.038, 0.110]	**0.1 [0.1, 0.1]**
$\mathrm{MS}_c(\bar{z})$	**0.007 [0.007, 0.008]**	70.2 [67.3, 72.6]	0.001 [0.001, 0.001]	0.033 [0.029, 0.037]	0.041 [0.039, 0.043]	**0.1 [0.1, 0.1]**
(c) Translation invariance: LTS with logits vs. log-probabilities						
$\mathrm{LTS}(\bar{z})^{\times}$	0.018 [0.005, 0.041]	**76.9 [76.0, 77.4]**	0.012 [0.001, 0.032]	0.134 [0.044, 0.301]	0.237 [0.168, 0.363]	—
$\mathrm{LTS}(\log S(\bar{z}))$	0.014 [0.005, 0.028]	**76.9 [76.0, 77.4]**	0.008 [0.001, 0.021]	0.090 [0.029, 0.189]	0.216 [0.117, 0.304]	—
(d) Decision preservation: CDC vs CMS						
$\mathrm{CDC}(\bar{p})$	0.008 [0.007, 0.009]	72.0 [68.6, 74.9]	0.001 [0.001, 0.001]	0.034 [0.030, 0.039]	0.046 [0.039, 0.053]	**0.1 [0.0, 0.1]**
$\mathrm{CMS}_{ap}(\bar{p})$	**0.007 [0.006, 0.008]**	76.0 [74.5, 77.5]	0.001 [0.001, 0.001]	0.029 [0.024, 0.034]	0.044 [0.031, 0.055]	—
$\mathrm{CMS}_{op}(\bar{p})$	**0.007 [0.006, 0.008]**	76.0 [74.4, 77.5]	0.001 [0.001, 0.001]	**0.028 [0.024, 0.033]**	0.065 [0.052, 0.077]	—
$\mathrm{CDC}(\bar{z})$	0.008 [0.006, 0.010]	74.0 [72.6, 75.5]	0.001 [0.001, 0.001]	0.035 [0.027, 0.042]	**0.033 [0.017, 0.057]**	**0.1 [0.0, 0.2]**
$\mathrm{CMS}_{ap}(\bar{z})$	**0.007 [0.007, 0.008]**	**76.9 [75.4, 78.4]**	0.001 [0.001, 0.001]	0.031 [0.026, 0.035]	0.041 [0.029, 0.057]	—
$\mathrm{CMS}_{op}(\bar{z})$	**0.007 [0.006, 0.008]**	**76.9 [75.3, 78.4]**	0.001 [0.001, 0.001]	0.030 [0.025, 0.035]	0.055 [0.039, 0.077]	—
(e) Standard temperature-based calibrators (order-preserving, TI): TS/ETS						
$\mathrm{TS}(\bar{p})$	0.025 [0.024, 0.025]	76.0 [74.5, 77.5]	0.005 [0.004, 0.006]	0.046 [0.043, 0.048]	0.146 [0.138, 0.157]	—
$\mathrm{TS}(\bar{z})$	0.029 [0.028, 0.029]	**76.9 [76.0, 77.4]**	0.007 [0.005, 0.008]	0.048 [0.045, 0.049]	0.167 [0.165, 0.170]	—
$\mathrm{ETS}(\bar{p})$	0.010 [0.010, 0.010]	76.0 [74.5, 77.5]	**0.000 [0.000, 0.000]**	0.047 [0.045, 0.049]	0.166 [0.163, 0.169]	—
$\mathrm{ETS}(\bar{z})$	0.010 [0.010, 0.010]	**76.9 [76.0, 77.4]**	**0.000 [0.000, 0.000]**	0.048 [0.046, 0.050]	0.187 [0.182, 0.192]	—
(f) Standard affine calibrator (non order-preserving, TI): DC						
$\mathrm{DC}(\bar{p})$	0.008 [0.007, 0.010]	68.4 [63.2, 72.9]	0.001 [0.001, 0.001]	0.037 [0.030, 0.043]	0.057 [0.029, 0.089]	**0.1 [0.1, 0.1]**
$\mathrm{DC}(\bar{z})$	0.008 [0.007, 0.010]	67.6 [57.2, 74.5]	0.001 [0.001, 0.001]	0.040 [0.032, 0.051]	0.065 [0.038, 0.110]	**0.1 [0.1, 0.1]**

Cityscapes Cordts et al. (2016)						
Pipeline	**NLL↓**	**DSC (%)↑**	**ECE↓**	**BA-ECE↓**	**ACE↓**	**Flip (%)↓**
(a) Uncalibrated baselines						
p_0	0.485 [0.457, 0.515]	71.7 [71.1, 72.3]	0.063 [0.059, 0.067]	0.182 [0.179, 0.185]	0.125 [0.124, 0.126]	—
$\bar{p}$	0.377 [0.352, 0.401]	78.1 [77.6, 78.6]	0.044 [0.040, 0.048]	0.136 [0.133, 0.139]	0.079 [0.076, 0.083]	—
$\bar{z}$	0.434 [0.427, 0.448]	**79.4 [79.3, 79.5]**	0.058 [0.057, 0.059]	0.170 [0.168, 0.172]	0.122 [0.120, 0.125]	—
(b) Translation invariance: MS vs. MS_c and DC						
$\mathrm{MS}(\bar{z})^{\times}$	0.404 [0.373, 0.439]	73.4 [70.7, 75.9]	0.045 [0.037, 0.055]	0.156 [0.135, 0.179]	0.072 [0.051, 0.104]	3.8 [3.4, 4.2]
$\mathrm{DC}(\bar{z})$	0.324 [0.301, 0.351]	70.4 [67.8, 73.5]	0.035 [0.032, 0.038]	0.096 [0.083, 0.119]	0.040 [0.019, 0.070]	**2.3 [2.1, 2.5]**
$\mathrm{MS}_c(\bar{z})$	0.328 [0.304, 0.359]	72.8 [70.2, 76.6]	0.034 [0.031, 0.038]	0.098 [0.085, 0.117]	0.046 [0.035, 0.066]	2.4 [1.9, 2.8]
(c) Translation invariance: LTS with logits vs. log-probabilities						
$\mathrm{LTS}(\bar{z})^{\times}$	0.344 [0.339, 0.354]	**79.4 [79.3, 79.5]**	0.016 [0.005, 0.029]	0.075 [0.048, 0.106]	0.039 [0.025, 0.061]	—
$\mathrm{LTS}(\log S(\bar{z}))$	0.342 [0.323, 0.366]	**79.4 [79.3, 79.5]**	**0.010 [0.009, 0.010]**	0.075 [0.070, 0.080]	0.029 [0.022, 0.042]	—
(d) Decision preservation: CDC vs CMS						
$\mathrm{CDC}(\bar{p})$	0.335 [0.314, 0.361]	69.3 [67.8, 71.2]	0.032 [0.029, 0.036]	0.108 [0.106, 0.111]	0.052 [0.045, 0.058]	5.1 [3.6, 6.0]
$\mathrm{CMS}_{ap}(\bar{p})$	**0.314 [0.296, 0.338]**	78.1 [77.6, 78.6]	0.035 [0.032, 0.039]	0.086 [0.076, 0.104]	0.041 [0.030, 0.062]	—
$\mathrm{CMS}_{op}(\bar{p})$	0.326 [0.306, 0.350]	78.1 [77.6, 78.6]	0.035 [0.031, 0.038]	0.071 [0.061, 0.086]	0.032 [0.020, 0.039]	—
$\mathrm{CDC}(\bar{z})$	0.341 [0.319, 0.370]	74.2 [73.6, 74.8]	0.035 [0.031, 0.039]	0.115 [0.110, 0.120]	0.067 [0.062, 0.071]	5.0 [4.4, 5.5]
$\mathrm{CMS}_{ap}(\bar{z})$	0.327 [0.308, 0.353]	**79.4 [78.9, 79.8]**	0.038 [0.033, 0.042]	0.091 [0.082, 0.106]	0.081 [0.067, 0.095]	—
$\mathrm{CMS}_{op}(\bar{z})$	0.339 [0.317, 0.367]	**79.4 [78.9, 79.8]**	0.035 [0.031, 0.038]	0.079 [0.069, 0.089]	0.041 [0.014, 0.056]	—
(e) Standard temperature-based calibrators (order-preserving, TI): TS/ETS						
$\mathrm{TS}(\bar{p})$	0.326 [0.320, 0.337]	78.1 [77.6, 78.6]	0.011 [0.007, 0.017]	**0.069 [0.059, 0.086]**	**0.025 [0.020, 0.033]**	—
$\mathrm{TS}(\bar{z})$	0.338 [0.332, 0.350]	**79.4 [79.3, 79.5]**	0.012 [0.005, 0.025]	0.079 [0.067, 0.099]	0.039 [0.013, 0.071]	—
$\mathrm{ETS}(\bar{p})$	0.326 [0.320, 0.337]	78.1 [77.6, 78.6]	**0.010 [0.005, 0.018]**	0.071 [0.060, 0.089]	0.026 [0.019, 0.040]	—
$\mathrm{ETS}(\bar{z})$	0.338 [0.332, 0.350]	**79.4 [79.3, 79.5]**	0.013 [0.005, 0.026]	0.079 [0.066, 0.099]	0.039 [0.012, 0.072]	—
(f) Standard affine calibrator (non order-preserving, TI): DC						
$\mathrm{DC}(\bar{p})$	0.319 [0.292, 0.356]	64.8 [62.0, 67.0]	0.033 [0.030, 0.037]	0.098 [0.079, 0.128]	0.037 [0.020, 0.066]	3.2 [2.8, 3.5]
$\mathrm{DC}(\bar{z})$	0.324 [0.301, 0.351]	70.4 [67.8, 73.5]	0.035 [0.032, 0.038]	0.096 [0.083, 0.119]	0.040 [0.019, 0.070]	**2.3 [2.1, 2.5]**

Table 8: **Complete calibration results on the test split.** — indicates not applicable; × denotes non–translation-invariant pipelines. **Bold** indicates the best mean value per dataset and metric.

J Reliability Diagrams

Figure 8 reports test-split reliability curves under the same convention as Table 4, with one panel per dataset comparing the single-model baseline, the uncalibrated pooling distribution, the best-ACE calibrated method selected without imposing order preservation, and the best-ACE order-preserving calibrated method.

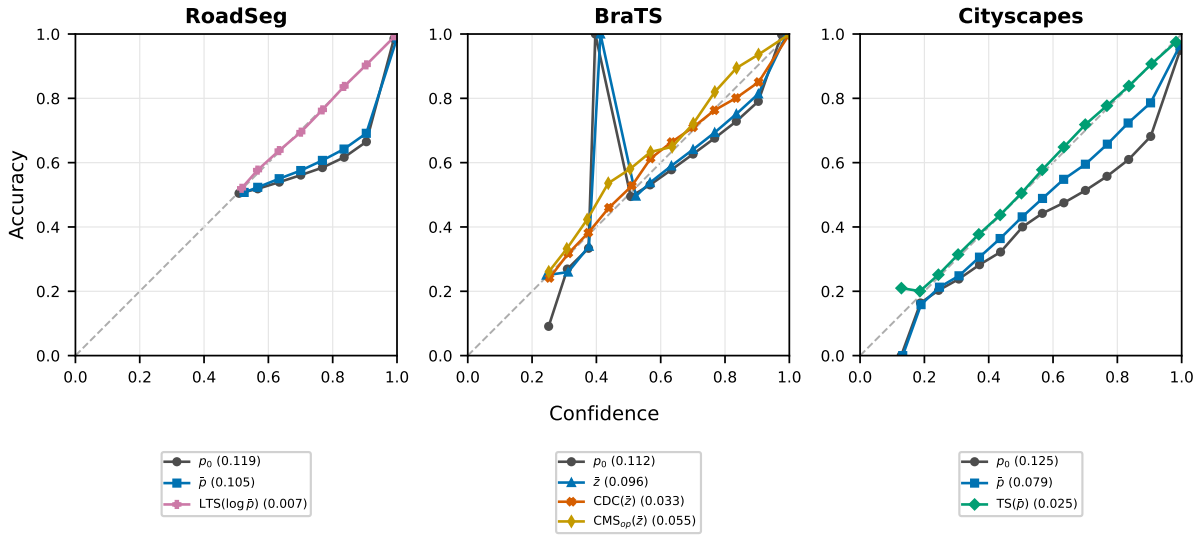

Figure 8: **Reliability diagrams.** Confidence–accuracy curves on the test split under the same ACE convention used in Table 4. Numbers in parentheses are mean test ACE across repeats. For each dataset, calibrated curves are selected by lowest ACE either over all calibrated methods or over the order-preserving subset; when both selections coincide, only one calibrated curve is shown. The dashed diagonal denotes perfect calibration.