A Statistical Significance

In Table 3, we present the results of one-sided one-sample t-tests conducted to evaluate the statistical significance of the performance gains achieved by our method in comparison to the strongest baselines. For each dataset and evaluation metric, we compare the results obtained from multiple independent runs of our best-performing configuration against the corresponding baseline means.

Table 3: **Statistical significance tests** of metric values compared between our best model and the best baseline values using the U-Net base model.

Dataset	Metric	t-value	p-value
BoMBR(Raina et al., 2024)	DSC	0.591	0.293
	clDice	0.828	0.227
	JSI	0.297	0.391
	FNR	-1.857	0.068
	FPR	-0.717	0.257
DRIVE(Hassan et al., 2015)	DSC	0.967	0.194
	clDice	0.206	0.423
	JSI	0.983	0.191
	FNR	2.053	0.945
	FPR	1.428	0.887
Cracks(Tomaszkiewicz & Owerko, 2023)	DSC	1.669	0.085
	clDice	2.467	0.035
	JSI	2.241	0.044
	FNR	5.072	0.996
	FPR	2.772	0.975
Drone[1]	DSC	1.4771	0.107
	clDice	1.690	0.083
	JSI	1.464	0.109
	FNR	-1.609	0.092
	FPR	-2.338	0.039

We observe that, while not all improvements reach statistical significance, there are multiple encouraging trends in favour of our approach. Notably, on the Cracks dataset, our method shows statistically significant improvements in both clDice ($p = 0.035$) and JSI ($p = 0.044$), suggesting reliable gains in capturing structural and overlap quality. Similarly, in the Drone dataset, a significant reduction in the false positive rate (FPR; $p = 0.039$) is observed, indicating better precision in delineating relevant regions.

Several other metrics, such as DSC and clDice on Drone, and DSC on Cracks, approach the significance threshold ($p < 0.1$), pointing to consistent, if not conclusive, improvements. On the BoMBR and DRIVE datasets, although most differences are not statistically significant, the metric values achieved by our method remain competitive with the baselines.

B The N-Factor

As per the function shown in Equation 1, the ESL loss function is given as:

$$\mathscr{L}_{\text{ESL}} = -\frac{\sum_{i\in\Omega} y_i\,\hat{y}_i}{N + \sum_{i\in\Omega} y_i(1-\hat{y}_i)}$$

The numerator counts the True Positives (TP), while the denominator combines the normalization term N with the FN. The constant N serves to stabilize the loss magnitude across images of different sizes or pixel counts, preventing the loss from becoming excessively large when many false negatives occur.

From a gradient perspective, consider the derivative of $\mathscr{L}_{\text{ESL}}$ with respect to a predicted pixel $\hat{y}_i$:

$$\begin{aligned}
\frac{\partial \mathscr{L}_{\text{ESL}}}{\partial \hat{y}_i} &= -\frac{v\,\frac{\partial u}{\partial \hat{y}_i} - u\,\frac{\partial v}{\partial \hat{y}_i}}{v^2} \\
&= -\frac{\left(N + \sum_{j\in\Omega} y_j(1-\hat{y}_j)\right) y_i - \left(\sum_{j\in\Omega} y_j \hat{y}_j\right)(-y_i)}{\left(N + \sum_{j\in\Omega} y_j(1-\hat{y}_j)\right)^2} \\
&= -\frac{y_i\left(N + \sum_{j\in\Omega} y_j(1-\hat{y}_j) + \sum_{j\in\Omega} y_j \hat{y}_j\right)}{\left(N + \sum_{j\in\Omega} y_j(1-\hat{y}_j)\right)^2} \\
&= -\frac{y_i\left(N + \sum_{j\in\Omega} y_j\right)}{\left(N + \sum_{j\in\Omega} y_j(1-\hat{y}_j)\right)^2}.
\end{aligned}$$

We may observe the following from the derived expression:

1. The gradient is proportional to y_i:

   $$\frac{\partial \mathscr{L}_{\text{ESL}}}{\partial \hat{y}_i} \propto y_i.$$

   Therefore, if $y_i = 0$ (corresponding to a negative pixel), then

   $$\frac{\partial \mathscr{L}_{\text{ESL}}}{\partial \hat{y}_i} = 0.$$

   This shows that True Negatives (TN) and FP pixels do not contribute to the gradient, and the loss specifically emphasizes the positive pixels ($y_i = 1$), i.e., the FN regions.
2. The numerator term ($N + \sum_{j\in\Omega} y_j$) is constant for a given image. Only the denominator

   $$D = N + \sum_{j\in\Omega} y_j(1-\hat{y}_j)$$

   varies with the predicted values, and it decreases as the number of correctly predicted positive pixels increases. Since

   $$\sum_{j\in\Omega} y_j(1-\hat{y}_j) \le \sum_{j\in\Omega} y_j,$$

   the denominator is always bounded below by N, preventing the gradient magnitude from becoming excessively large.

Thus, including N in the denominator ensures numerical stability:

$$\left|\frac{\partial \mathscr{L}_{\text{ESL}}}{\partial \hat{y}_i}\right| = \frac{y_i\left(N + \sum_{j\in\Omega} y_j\right)}{\left(N + \sum_{j\in\Omega} y_j(1-\hat{y}_j)\right)^2} \le \frac{y_i\left(N + \sum_{j\in\Omega} y_j\right)}{N^2} \le \frac{y_i N}{N^2},$$

$$\Longrightarrow \left|\frac{\partial \mathscr{L}_{\text{ESL}}}{\partial \hat{y}_i}\right| \le \frac{y_i}{N}.$$

Since $y_i \in \{0, 1\}$, we have

$$\left|\frac{\partial \mathscr{L}_{\text{ESL}}}{\partial \hat{y}_i}\right| \le \frac{1}{N}.$$

,

which guarantees bounded and stable gradients even for sparse positive targets, ensuring stable optimization throughout training.

This analysis highlights that the gradient flow for our loss function is entirely concentrated on positive pixels, directly targeting the FN regions while ignoring TN and FP contributions. Moreover, the presence of N in the denominator effectively scales the gradient, ensuring that its magnitude remains bounded by $1/N$ regardless of the number of positive pixels or their predictions. Consequently, the loss maintains sensitivity to challenging regions without causing unstable or excessively large updates, supporting consistent and stable training even for sparse masks. Thus, N acts as a normalization factor, balancing sensitivity to false negatives with overall numerical stability.

C Testing SMM on SegNet

To further demonstrate the versatility and robustness of our proposed framework, we evaluated both variants of SMM on SegNet (Badrinarayanan et al., 2017), a widely used segmentation architecture that differs from U-Net in its encoder-decoder design and feature propagation strategy. This experiment highlights the architecture-agnostic nature of our approach, showing that the unified mask modulation and generalizable training strategy can be applied to diverse segmentation networks while maintaining high performance. Table 4 presents the test set metrics for SegNet trained under

Table 4: **Test set metrics** of SegNet models trained using different strategies. SMM*v1* and SMM*v2* present results for both versions of our proposed framework.

Method	DSC ↑	clDice ↑	JSI ↑	FNR ↓	FPR ↓
			BoMBR (Raina et al., 2024)		
SegNet	64.35 ± 1.82	61.56 ± 1.97	54.26 ± 2.05	26.11 ± 2.02	8.17 ± 0.65
SegNet + SRL	64.67 ± 2.89	62.33 ± 3.05	54.83 ± 2.89	27.00 ± 3.09	8.08 ± 1.08
SegNet + BL	64.12 ± 2.00	61.94 ± 1.42	54.86 ± 2.16	28.16 ± 0.72	8.43 ± 1.07
SMM*v1*	$\mathbf{66.52 \pm 1.07}$	$\mathbf{64.17 \pm 1.21}$	$\mathbf{56.99 \pm 1.19}$	$\mathbf{25.62 \pm 1.72}$	$\mathbf{7.33 \pm 0.20}$
SMM*v2*	65.76 ± 1.02	63.87 ± 0.93	56.31 ± 1.14	25.92 ± 1.21	7.83 ± 0.83
			DRIVE (Hassan et al., 2015)		
SegNet	66.52 ± 5.41	66.32 ± 5.61	50.13 ± 5.94	38.07 ± 5.03	3.33 ± 0.71
SegNet + SRL	63.96 ± 3.10	63.70 ± 3.04	47.17 ± 3.42	38.28 ± 3.15	4.30 ± 0.46
SegNet + BL	65.51 ± 4.90	66.10 ± 5.27	48.99 ± 5.55	42.83 ± 6.07	$\mathbf{2.32 \pm 0.49}$
SMM*v1*	66.63 ± 4.92	66.02 ± 4.95	50.23 ± 5.44	38.54 ± 5.10	3.14 ± 0.50
SMM*v2*	$\mathbf{67.06 \pm 4.96}$	$\mathbf{66.79 \pm 5.06}$	$\mathbf{50.72 \pm 5.50}$	$\mathbf{36.22 \pm 5.06}$	3.63 ± 0.62

various strategies, including baseline training, self-regularized learning (SRL), boundary loss (BL), and our proposed SMM variants. Across both datasets, SMM consistently improves segmentation performance compared to standard training strategies, achieving higher Dice, clDice, and Jaccard scores while reducing false negative and false positive rates. These results confirm that the effectiveness of SMM is not confined to a single architecture, underscoring its potential for broad deployment across different segmentation models.

D Versional Design Led Superiority

The results presented in Table 2 highlight the distinct behaviors of the two variants of SMM across datasets with different characteristics. For clarity, we classify the datasets into two categories:

1. **Negative-dominant datasets**: The background is considerably more diverse and substantially larger than the foreground. Segmentation in such cases is particularly challenging due to bias toward the more abundant negative class. Representative datasets include DRIVE (Hassan et al., 2015) and Cracks (Tomaszkiewicz & Owerko, 2023).
2. **Balanced or foreground-rich datasets**: The classes are approximately balanced, or foreground pixels slightly dominate. Here, the primary challenge lies in capturing fine struc-

tural details and ensuring accurate delineation of class boundaries. Examples include BoMBR (Raina et al., 2024) and Drone[1].

For architectures such as U-Net, SMM*v1* consistently improves overlap- and topology-oriented metrics relative to vanilla baselines, with the largest gains observed in Category 1 datasets. This observation emphasizes the effectiveness of the ESL loss in tasks dominated by negative samples, thereby positioning SMM*v1* as the more robust variant under such conditions. In contrast, SMM*v2* demonstrates superior performance on Category 2 datasets, reflecting its suitability for scenarios where the segmentation task depends less on class imbalance and more on semantic precision and fine-grained contextual reasoning. These findings are further substantiated by the qualitative results shown in Figure 4.

It should be noted that the aforementioned trends are observed in architectures that are not specifically tailored to a particular domain or task (e.g., U-Net). We do not assert these patterns as universal across all segmentation models.

Interestingly, the trend reverses when considering the results in Table 4. In this case, SMM*v1* exhibits stronger performance on Category 2, whereas SMM*v2* proves more effective for Category 1 datasets.