GEODESIC CONFORMAL RISK CONTROL FOR PROVABLY SAFE IMAGE SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Applying Conformal Risk Control (CRC) to image segmentation via standard pixel-wise thresholding often yields severely fragmented predictions under strict instance-level risk constraints. To address this, we propose Geodesic Conformal Risk Control (GeoCRC), a framework that ensures spatially coherent predictions while rigorously bounding the connected-component miss risk. GeoCRC leverages Isotonic Regression calibration to identify reliable seed regions and define traversal costs. Through multi-source geodesic propagation, it constructs a topologically well-behaved, nested family of regions parameterized by a geodesic radius, which is then calibrated using an instance-level miss loss. Theoretically, GeoCRC inherits the distribution-free marginal risk guarantees of standard CRC. Extensive experiments demonstrate that GeoCRC strictly satisfies user-specified risk constraints while drastically reducing spatial fragmentation and false-positive noise compared to baseline methods.

1 INTRODUCTION

The rapid development of deep learning and computer vision has significantly advanced image segmentation techniques, establishing them as the cornerstone of numerous visual perception tasks. From industrial defect detection and remote sensing to autonomous driving and medical diagnostics, existing image segmentation models have achieved remarkable performance across these diverse domains. However, deploying these models in safety-critical, high-risk scenarios requires not only high accuracy but also strict control over the risk of missed target regions.

Conformal Risk Control (CRC) (Angelopoulos et al., 2024) emerges as a principled statistical framework to address this challenge. Under the exchangeability assumption, CRC provides finite-sample, distribution-free guarantees on the expected value of bounded monotone loss functions. Nevertheless, directly applying standard CRC to image segmentation faces a severe metric bias problem. Standard pixel-level or image-level metrics are inherently dominated by large objects in the image. In tasks like medical lesion diagnosis or wildfire monitoring, missing tiny yet fatal instances has a negligible impact on the overall False Negative Rate (FNR) but can lead to catastrophic downstream risks. Therefore, it is imperative to shift the granularity of risk control from the pixel level to the connected-component level, ensuring that every independent target structure is explicitly accounted for in the risk assessment.

However, merely changing the evaluation metric does not resolve the fundamental structural flaws of CRC in spatial domains. The core mechanism of CRC relies on calibrating a given "nested prediction family," but it does not dictate the spatial morphology of this family. When this nested family is constructed simply by thresholding pixel-wise probabilities, enforcing strict instance-level miss risk constraints forces the model to adopt highly conservative cutoff thresholds. This inevitably introduces a massive amount of isolated noise, leads to severely fragmented prediction regions, and generates topological holes. The spatial incoherence of the predictions not only degrades interpretability but also severely hinders downstream tasks such as lesion delineation and therapy planning.

To address the aforementioned challenges, we propose Geodesic Conformal Risk Control (GeoCRC), an instance-level risk control framework that constructs spatially coherent nested prediction regions through geodesic propagation. Figure 1 provides an overview of GeoCRC and contrasts it with raw probability-threshold CRC. Our main contributions are summarized as follows:

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

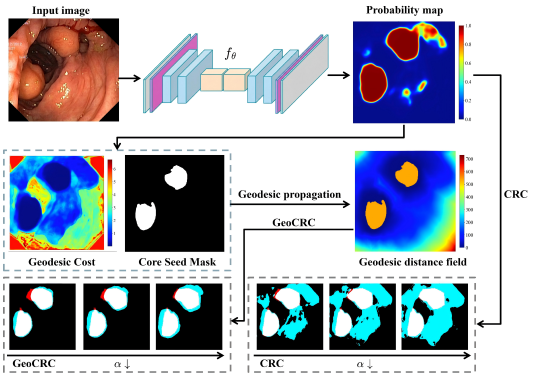


Figure 1: Motivation and overview of Geodesic Conformal Risk Control (GeoCRC). In the final prediction maps, white pixels represent true positives, red pixels represent false negatives, cyan pixels represent false positives, and orange pixels represent core seeds. Compared with raw probability-threshold CRC, GeoCRC significantly suppresses spatial fragmentation and false-positive noise while preserving spatially coherent target regions.

- **Pioneering Instance-Level Risk Control:** To the best of our knowledge, we are the first to identify the spatial and structural limitations of standard CRC in segmentation tasks, and we pioneer the study of instance-level risk control to prevent critical misses of small targets.
- **Calibrated Geodesic Framework:** We propose a spatial region construction method based on isotonic-mapped probabilities to identify reliable seeds and define pixel-wise traversal costs. This constructs a spatially coherent nested family of regions through multi-source geodesic propagation.
- **Rigorous Theoretical Guarantees:** We mathematically prove that our proposed GeoCRC framework inherits the distribution-free marginal risk guarantees of standard CRC, ensuring that the expected instance-level miss rate is strictly bounded under the exchangeability assumption.
- **Superior Empirical Performance:** Extensive experimental results across multiple image segmentation benchmarks demonstrate that, compared to raw probability-threshold CRC, our method significantly suppresses spatial fragmentation and false-positive noise while strictly satisfying the user-specified risk constraints.

2 Related Work

2.1 Conformal Prediction for Image Analysis

Conformal prediction (CP) (Vovk et al., 2005) provides statistically valid uncertainty quantification for image analysis by converting model outputs into prediction sets with distribution-free coverage guarantees. This is particularly valuable in high-stakes visual applications, where standard confidence scores may be poorly calibrated. Early work applied CP to large-scale histopathology images (Wieslander et al., 2020) and image-level quality control (Wundram et al., 2024).

CP has since been applied to image classification through adaptive label sets (Romano et al., 2020), and extended to structured tasks such as semantic segmentation (Mossina et al., 2024), object detection (Timans et al., 2024), and vision foundation models (Chen et al., 2025). Conformal risk control further enables guarantees on task-specific visual errors beyond standard coverage (Angelopoulos et al., 2024).

2.2 Conformal Risk Control for Image Segmentation

CRC (Angelopoulos et al., 2024) is a distribution-free conformal prediction method with provable theoretical guarantees for bounded monotone losses, which has been widely adopted for image segmentation in recent research.

In the context of segmentation, existing works have primarily focused on controlling pixel-level or image-level metrics. Recent studies have optimized the FNR or false positive rate (FPR) via semantic-specific calibration (Teneggi et al., 2023; 2025), subgroup risk control (Fischer et al., 2024), and image-adaptive thresholding (Brunekreef et al., 2024; Blot et al., 2025; Luo et al., 2026). Another branch of work expands CRC to constrain non-monotonic quantities like the false discovery rate (FDR) (Angelopoulos, 2026; Aldirawi et al., 2026). However, these methods fundamentally operate at the independent pixel level or focus solely on image-level marginal coverage, frequently producing fragmented predictions with isolated noise and topological holes.

To encourage spatial coherence, a few recent studies have introduced spatial priors into the conformal pipeline, such as distance-based penalties (Bereska et al., 2025), morphological margins (Mossina et al., 2025), and feature-based random walks (Gaillochet et al., 2026). However, in safety-critical applications such as medical lesion diagnosis, industrial defect detection, and wildfire monitoring, images frequently contain multiple distinct instances of varying sizes. Standard pixel-level CRC is often dominated by large targets, masking critical misses on smaller instances and leaving a significant gap in reliable instance-level risk control.

In addition, CRC can bound the expected instance-level miss rate, yet directly constraining raw pixel probabilities yields overly conservative thresholds that cause fragmented masks, isolated noise and topological holes. To tackle these structural flaws, we propose Geodesic Conformal Risk Control (GeoCRC). It replaces naive probability thresholding with geodesic propagation based on isotonic-calibrated probabilities, building spatially coherent regions from reliable seed masks and calibrated-probability traversal costs. We further calibrate the geodesic radius via a connected-component miss loss, unifying intact topology and rigorous instance-level risk control.

3 Background

3.1 Problem Setup

For image segmentation, we consider an input image X_i and its ground-truth binary mask $Y_i \in \{0,1\}^N$. The primary objective is to construct a prediction set $\widehat{\mathcal{C}}(X_i) \subset \{1, 2, \ldots, N\}$ that accurately delineates the regions of interest.

Most prior works on segmentation-based CRC focus on deriving bounds for pixel-level metrics, such as FNR and FPR. However, these metrics are inherently dominated by large objects. In safety-critical tasks like wildfire, industrial defect and medical lesion detection, missing tiny critical instances barely impacts overall FNR but yields severe downstream risks.

To address this limitation, we shift our focus from pixel-level metrics to component-level coverage. Let $G_i = \{j \in \{1, \ldots, N\} : Y_i[j] = 1\}$ denote the ground-truth foreground. For an image with $|G_i| > 0$, let $\{G_{ik}\}_{k=1}^{K_i}$ be its connected components, each representing a distinct instance. We consider an instance G_{ik} to be successfully "covered" if the prediction set encompasses at least a predefined proportion $\eta \in (0, 1]$ of its pixels. The instance-level coverage rate of an image is defined as:

$$\mathrm{CovRate}(\widehat{\mathcal{C}}(X_i), Y_i) = \frac{1}{K_i} \sum_{k=1}^{K_i} \mathbb{I}\left[\frac{|\widehat{\mathcal{C}}(X_i) \cap G_{ik}|}{|G_{ik}|} \geq \eta \right]. \tag{1}$$

Our objective is to apply CRC to ensure that the expected instance-level miss rate is controlled at a user-specified risk level $\alpha \in (0, 1)$:

$$\mathbb{E}\Big[1 - \mathrm{CovRate}(\widehat{\mathcal{C}}(X_i), Y_i)\Big] \leq \alpha, \tag{2}$$

where the expectation is taken over random draws of the test data containing at least one ground-truth instance.

Though direct pixel thresholding achieves coverage targets, strict instance coverage forces conservative cutoffs that create fragmented false-positive regions. Such artifacts degrade spatial consistency, impeding interpretability and downstream tasks including lesion delineation and therapy planning. Therefore, we seek to maximize spatial coherence and suppress noise in prediction sets whilst adhering to the strict coverage target given by Eq. equation 2.

3.2 CONFORMAL RISK CONTROL

CRC (Angelopoulos et al., 2024) provides distribution-free guarantees on the expected value of any bounded monotone loss function. In our context, the loss function is the instance-level miss rate:

$$\ell(\widehat{\mathcal{C}}(X_i), Y_i) = 1 - \mathrm{CovRate}(\widehat{\mathcal{C}}(X_i), Y_i) \in [0, 1]. \tag{3}$$

CRC achieves risk control by calibrating a nested family of prediction sets generated by a base segmentation model. Given a model that outputs a pixel-wise probability map $\hat{p}(X_i) = (\hat{p}_1(X_i), \ldots, \hat{p}_N(X_i))$ for an input image X_i, the CRC procedure proceeds in the following three steps:

1. **Prediction Set Definition:** For a given threshold τ, we define a binary prediction set $\widehat{\mathcal{C}}(X_i, \tau)$. This set includes all pixels j whose predicted probability is at least τ:

$$\widehat{\mathcal{C}}(X_i, \tau) = \{j \in \{1, \ldots, N\} : \hat{p}_j(X_i) \geq \tau\}. \tag{4}$$

2. **Calibrated Threshold Computation:** We use a calibration dataset $\mathcal{D}_{\mathrm{cal}}$ of size n to determine the calibrated threshold $\hat{\tau}$. It is chosen as the most stringent threshold that satisfies the empirical risk constraint:

$$\hat{\tau} = \sup \left\{ \tau : \frac{1}{n+1} \left(\sum_{X_i \in \mathcal{D}_{\mathrm{cal}}} \ell(\widehat{\mathcal{C}}(X_i, \tau), Y_i) + B \right) \leq \alpha \right\}, \tag{5}$$

   where $B = 1$ is the upper bound of the instance-miss loss function.

3. **Final Prediction Set:** For a new test dataset X_{test}, the final prediction set is constructed using the calibrated parameter $\hat{\tau}$:

$$\widehat{\mathcal{C}}(X_{\mathrm{test}}) = \widehat{\mathcal{C}}(X_{\mathrm{test}}, \hat{\tau}). \tag{6}$$

This methodology mathematically guarantees that the expected instance-level miss rate over the test distribution is bounded by α.

4 GEODESIC CONFORMAL RISK CONTROL

This section introduces the GeoCRC pipeline based on monotonic isotonic calibration. Compared to direct pixel-wise thresholding, this method improves spatial coherence while controlling the risk of missing lesions through conformal risk control.

4.1 POINT-WISE PROBABILITY MAPPING VIA ISOTONIC REGRESSION.

The segmentation model first generates a pixel-wise probability map $\hat{p}_i(u)$ for image X_i at pixel u. We fit a monotonic isotonic mapping on a disjoint fitting split $\mathcal{D}_{\mathrm{Iso}}$. The rectified probability is:

$$p_i^{\mathrm{Iso}}(u) = \mathrm{Iso}(\hat{p}_i(u)), \tag{7}$$

where $\mathrm{Iso}(\cdot)$ denotes the fitted binned isotonic mapping.

4.2 SEEDED GEODESIC PROPAGATION.

To enforce spatial coherence and prevent the fragmented artifacts typical of direct pixel-wise thresholding, we grow our prediction sets from highly reliable seed regions. we extract a seed mask using the mapped probability:

$$M_i = \left\{ u : p_i^{\mathrm{Iso}}(u) \geq \tau_{\mathrm{core}} \right\}, \tag{8}$$

where τ_{core} is the confidence threshold. By adjusting τ_{core} for various tasks, we can obtain either a single seed region or multiple seed regions.

Next, to govern the spatial expansion from these seeds, we define a traversal cost for each pixel:

$$c_i(u) = -\log\left(\max\left\{p_i^{\mathrm{Iso}}(u), \varepsilon\right\}\right), \tag{9}$$

where $\varepsilon > 0$ is a numerical stability constant. Therefore, highly calibrated pixels have a lower traversal cost, while low-confidence pixels have a high traversal cost.

Starting from all pixels in M_i, a multi-source minimum-cost propagation computes the geodesic distance for every pixel u:

$$D_i(u, M_i) = \min_{\gamma: u \rightsquigarrow M_i} \sum_{v \in \gamma} c_i(v), \tag{10}$$

where γ represents a discrete path connecting u to any seed pixel in M_i.

Finally, for a given distance radius $\lambda \geq 0$, the corresponding prediction region is defined by thresholding the geodesic distance:

$$\widehat{\mathcal{C}}^{\mathrm{geo}}(X_i; \lambda) = \{u : D_i(u, M_i) \leq \lambda\}. \tag{11}$$

Crucially, this distance-based formulation guarantees that the resulting family of sets is strictly nested. This nested property is a fundamental prerequisite for the subsequent Conformal Risk Control calibration.

4.3 CRC Calibration and Guarantee.

We substitute the geodesic prediction family from Eq. equation 11 into the general CRC criterion in Eq. equation 5. The calibrated distance radius is selected as the smallest feasible value that satisfies the empirical risk constraint:

$$\widehat{\lambda}_{\mathrm{geo}} = \inf\left\{\lambda : \frac{1}{n+1}\left(\sum_{(X_i,Y_i)\in\mathcal{D}_{\mathrm{cal}}} \ell\big(\widehat{\mathcal{C}}^{\mathrm{geo}}(X_i;\lambda), Y_i\big) + 1\right) \leq \alpha\right\}. \tag{12}$$

Once calibrated, $\widehat{\lambda}_{\mathrm{geo}}$ is fixed, and the final geodesic prediction for any new test image X_{n+1} is given by $\widehat{\mathcal{C}}^{\mathrm{geo}}(X_{n+1}) = \widehat{\mathcal{C}}^{\mathrm{geo}}(X_{n+1}; \widehat{\lambda}_{\mathrm{geo}})$.

Theorem 1 (Coverage Guarantees)**.** *Let $\mathcal{D}_{\mathrm{cal}} = \{(X_i, Y_i)\}_{i=1}^n$ be the calibration set and (X_{n+1}, Y_{n+1}) be a test sample. Suppose $\{(X_i, Y_i)\}_{i=1}^{n+1}$ are exchangeable and independent of the isotonic calibration split $\mathcal{D}_{\mathrm{Iso}}$. Then the final prediction set $\widehat{\mathcal{C}}^{\mathrm{geo}}(X_{n+1})$ satisfies:*

$$\mathbb{E}\Big[1 - \mathrm{CovRate}\big(\widehat{\mathcal{C}}^{\mathrm{geo}}(X_{n+1}), Y_{n+1}\big)\Big] \leq \alpha. \tag{13}$$

Proof. Conditioning on $\mathcal{D}_{\mathrm{Iso}}$ fixes the isotonic calibration mapping and the geodesic cost function. For each sample (X_i, Y_i), we define a parameterized loss function $\ell_i(\lambda)$ for $\lambda \in [0, \infty)$:

$$\ell_i(\lambda) = 1 - \mathrm{CovRate}\big(\widehat{\mathcal{C}}^{\mathrm{geo}}(X_i; \lambda), Y_i\big). \tag{14}$$

By the definition of the geodesic family in Eq. equation 11, the prediction set $\widehat{\mathcal{C}}^{\mathrm{geo}}(X_i; \lambda)$ expands as λ increases. Consequently, the instance-level coverage rate is non-decreasing, making the loss function $\ell_i(\lambda)$ a non-increasing function of λ. Furthermore, it is bounded such that $\sup_\lambda \ell_i(\lambda) \leq 1$.

We compute the empirical risk on the calibration set as:

$$\hat{R}(\lambda) = \frac{1}{n+1}\sum_{i=1}^{n} \ell_i(\lambda) + \frac{1}{n+1}. \tag{15}$$

The calibrated radius defined in Eq. equation 12 can be equivalently written as:

$$\widehat{\lambda}_{\mathrm{geo}} = \inf\left\{\lambda : \hat{R}(\lambda) \leq \alpha\right\}. \tag{16}$$

Since $\ell_i(\lambda)$ is a non-increasing and bounded function, and the samples $\{(X_i, Y_i)\}_{i=1}^{n+1}$ are exchangeable, we can directly apply Theorem 1 in Angelopoulos et al. (2024), which yields:

$$\mathbb{E}\Big[\ell_{n+1}(\widehat{\lambda}_{\mathrm{geo}}) \,\Big|\, \mathcal{D}_{\mathrm{Iso}}\Big] \leq \alpha. \tag{17}$$

Finally, taking the expectation over the random draws of $\mathcal{D}_{\text{Iso}}$ preserves the inequality:

$$\mathbb{E}\Big[\ell_{n+1}(\widehat{\lambda}_{\text{geo}})\Big] \leq \alpha. \tag{18}$$

□

Algorithm 1 Geodesic Conformal Risk Control

Input: labeled data $\mathcal{D}_{\text{train}}$, unlabel test data $\mathcal{D}_{\text{test}}$, target instance-miss risk $\alpha \in (0, 1)$, "covered" ratio η.
Randomly split $\mathcal{D}_{\text{train}}$ into $\mathcal{D}_1$, $\mathcal{D}_{\text{Iso}}$, and $\mathcal{D}_{\text{cal}}$.
Train a semantic segmentation model using $\mathcal{D}_1$.
▷ **Phase 1: Monotonic Probability Mapping**
Compute the mapped probability $p_i^{\text{Iso}}(u)$.
Fit an isotonic probability mapping using $\mathcal{D}_{\text{Iso}}$.
▷ **Phase 2: Region Construction & CRC Calibration**
for each $(X_i, Y_i) \in \mathcal{D}_{\text{cal}}$ **do**
  Extract conservative seed mask M_i.
  Calculate the traversal cost $c_i(u)$ at location u.
  Compute the geodesic distance field $D_i(\cdot, M_i)$.
  Construct nested geodesic regions:
    $\widehat{\mathcal{C}}^{\text{geo}}(X_i; \lambda) \leftarrow \{u : D_i(u, M_i) \leq \lambda\}$.
end for
Compute calibrated radius $\widehat{\lambda}_{\text{geo}}$ using loss ℓ.
▷ **Phase 3: Test-Time Inference**
for each $X_i \in \mathcal{D}_{\text{test}}$ **do**
  Compute the geodesic distance field $D_i(\cdot, M_i)$.
  Construct the final prediction region:
    $\widehat{\mathcal{C}}^{\text{geo}}(X_i) \leftarrow \{u : D_i(u, M_i) \leq \widehat{\lambda}_{\text{geo}}\}$.
end for
Output: $\widehat{\mathcal{C}}^{\text{geo}}(X_i)$ for $X_i \in \mathcal{D}_{\text{test}}$.

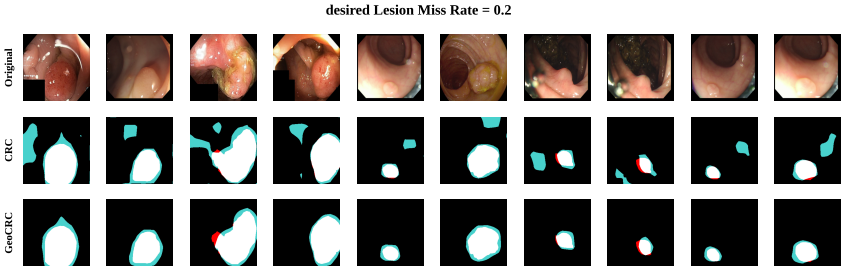


Figure 2: Qualitative comparison of CRC and GeoCRC prediction sets at a significance level of $\alpha = 0.2$. The top row shows original polyp images, the middle row displays CRC prediction sets, and the bottom row presents GeoCRC prediction sets. White pixels represent true positives, red false negatives, and cyan false positives. FNR values are given for each prediction.

5 Experiments

5.1 Experimental Setup

We evaluate our method on several segmentation datasets, including Polyp (Jha et al., 2019; Bernal et al., 2017; Vázquez et al., 2017; Tajbakhsh et al., 2015; Silva et al., 2014), BRISC (Fateh et al., 2026), ACDC (Bernard et al., 2018), and Fire (Aktaş, 2023). The training set sizes are 1,450, 1,344, 3,933, and 21,968, respectively. For each dataset, the remaining data are split into 70% for calibration-related purposes and 30% for testing. Standard CRC uses this entire 70% as the

Dataset	Model	α	Method	Lesion Miss Rate	ASSD ↓	HD95 ↓	Comp ↓	FP Comp ↓
Polyp	Deeplab v3+	0.1	CRC	0.105 (0.024)	80.806 (1.464)	170.518 (2.196)	1.194 (0.047)	0.784 (0.045)
			GeoCRC	0.102 (0.028)	**48.934** (5.575)	**84.738** (6.887)	**1.103** (0.038)	**0.244** (0.033)
		0.2	CRC	0.204 (0.028)	79.807 (1.129)	163.379 (1.793)	2.655 (0.358)	2.201 (0.334)
			GeoCRC	0.202 (0.029)	**28.036** (2.044)	**62.853** (4.123)	**1.325** (0.058)	**0.368** (0.055)
	UNet	0.1	CRC	0.094 (0.017)	102.847 (1.470)	177.411 (1.786)	1.310 (0.086)	0.909 (0.089)
			GeoCRC	0.094 (0.025)	**89.961** (4.064)	**144.345** (5.510)	**1.008** (0.005)	**0.514** (0.038)
		0.2	CRC	0.197 (0.033)	95.759 (1.475)	167.939 (1.726)	3.525 (0.413)	3.079 (0.401)
			GeoCRC	0.205 (0.036)	**63.304** (3.570)	**115.415** (5.868)	**1.205** (0.052)	**0.428** (0.035)
	PSPNet	0.1	CRC	0.102 (0.025)	80.076 (1.789)	171.839 (2.878)	1.119 (0.022)	0.583 (0.031)
			GeoCRC	0.094 (0.028)	**46.379** (6.043)	**76.404** (7.241)	**1.054** (0.020)	**0.186** (0.024)
		0.2	CRC	0.195 (0.035)	78.949 (1.446)	170.265 (2.651)	1.332 (0.038)	0.649 (0.029)
			GeoCRC	0.189 (0.032)	**28.249** (2.133)	**54.248** (3.556)	**1.110** (0.017)	**0.179** (0.024)
	SINet	0.1	CRC	0.104 (0.020)	93.818 (1.208)	178.574 (1.855)	**1.000** (0.000)	0.592 (0.018)
			GeoCRC	0.092 (0.018)	**71.091** (7.975)	**105.850** (14.200)	1.042 (0.077)	**0.350** (0.101)
		0.2	CRC	0.187 (0.022)	85.143 (1.625)	175.500 (1.787)	**1.017** (0.007)	0.590 (0.028)
			GeoCRC	0.196 (0.032)	**39.238** (4.277)	**65.412** (10.325)	1.101 (0.164)	**0.230** (0.172)
Brisc	Deeplab v3+	0.1	CRC	0.107 (0.021)	76.458 (0.479)	152.322 (1.320)	9.787 (3.914)	9.574 (3.856)
			GeoCRC	0.106 (0.020)	**9.572** (0.758)	**17.779** (1.357)	**1.043** (0.010)	**0.046** (0.010)
		0.2	CRC	0.202 (0.024)	76.556 (2.932)	145.885 (4.333)	126.673 (2.128)	125.548 (2.121)
			GeoCRC	0.195 (0.033)	**6.236** (0.507)	**13.501** (0.984)	**1.060** (0.018)	**0.059** (0.018)
	UNet	0.1	CRC	0.102 (0.023)	81.698 (1.492)	153.270 (1.772)	15.979 (5.680)	15.889 (5.678)
			GeoCRC	0.095 (0.024)	**48.023** (2.534)	**99.190** (3.459)	**1.763** (0.183)	**0.925** (0.131)
		0.2	CRC	0.196 (0.029)	78.654 (2.106)	146.251 (3.057)	166.785 (10.609)	166.024 (10.444)
			GeoCRC	0.192 (0.027)	**36.862** (1.377)	**86.277** (2.673)	**2.328** (0.087)	**1.357** (0.082)
	PSPNet	0.1	CRC	0.098 (0.021)	94.092 (0.700)	168.443 (0.932)	2.502 (0.360)	2.045 (0.318)
			GeoCRC	0.094 (0.023)	**14.692** (0.886)	**23.887** (1.294)	**1.039** (0.011)	**0.067** (0.015)
		0.2	CRC	0.185 (0.029)	65.691 (4.677)	155.145 (5.646)	7.208 (0.366)	6.261 (0.373)
			GeoCRC	0.194 (0.035)	**10.141** (0.587)	**18.595** (1.103)	**1.053** (0.019)	**0.071** (0.020)
	SINet	0.1	CRC	0.094 (0.023)	115.147 (1.146)	179.344 (1.090)	**1.000** (0.000)	0.929 (0.015)
			GeoCRC	0.107 (0.024)	**10.236** (1.059)	**17.037** (1.421)	1.018 (0.007)	**0.032** (0.009)
		0.2	CRC	0.203 (0.028)	110.042 (1.031)	178.480 (0.884)	**1.000** (0.000)	0.931 (0.015)
			GeoCRC	0.214 (0.035)	**6.990** (0.452)	**13.215** (0.709)	1.028 (0.008)	**0.033** (0.012)
Acdc	Deeplab v3+	0.1	CRC	0.093 (0.031)	88.880 (4.314)	154.336 (3.807)	18.094 (7.217)	17.457 (7.149)
			GeoCRC	0.094 (0.035)	**22.417** (5.740)	**26.063** (6.218)	**1.954** (0.634)	**1.021** (0.630)
		0.2	CRC	0.178 (0.038)	40.888 (10.029)	102.544 (14.831)	20.669 (10.806)	19.739 (10.800)
			GeoCRC	0.182 (0.040)	**21.401** (4.488)	**24.931** (4.836)	**1.991** (0.916)	**1.064** (0.918)
	UNet	0.1	CRC	0.102 (0.028)	122.945 (2.830)	172.272 (2.467)	3.215 (0.486)	3.206 (0.484)
			GeoCRC	0.094 (0.032)	**61.397** (4.663)	**110.770** (4.636)	**2.550** (0.277)	**1.721** (0.258)
		0.2	CRC	0.196 (0.032)	115.698 (3.028)	171.168 (2.460)	6.741 (0.762)	6.712 (0.754)
			GeoCRC	0.188 (0.041)	**55.140** (3.957)	**103.654** (4.290)	**3.458** (0.218)	**2.584** (0.213)
	PSPNet	0.1	CRC	0.091 (0.022)	100.975 (4.014)	159.120 (3.301)	2.706 (0.755)	1.978 (0.675)
			GeoCRC	0.086 (0.029)	**30.047** (3.911)	**35.763** (4.023)	**1.009** (0.007)	**0.114** (0.020)
		0.2	CRC	0.192 (0.037)	59.084 (5.648)	114.571 (8.035)	4.857 (0.690)	3.965 (0.675)
			GeoCRC	0.183 (0.034)	**27.314** (4.585)	**33.043** (4.969)	**1.034** (0.046)	**0.129** (0.047)
	SINet	0.1	CRC	0.096 (0.031)	129.424 (4.868)	175.276 (3.570)	**1.000** (0.000)	1.000 (0.000)
			GeoCRC	0.100 (0.033)	**26.691** (6.049)	**30.524** (6.049)	1.025 (0.011)	**0.121** (0.023)
		0.2	CRC	0.189 (0.046)	122.517 (4.164)	173.704 (3.147)	**1.000** (0.000)	1.000 (0.000)
			GeoCRC	0.189 (0.025)	**23.720** (5.508)	**27.352** (5.589)	1.022 (0.010)	**0.112** (0.020)
Fire	Deeplab v3+	0.1	CRC	0.100 (0.007)	37.870 (1.014)	116.720 (1.658)	25.013 (0.807)	23.138 (0.812)
			GeoCRC	0.100 (0.005)	**27.472** (2.078)	**40.309** (2.084)	**1.157** (0.027)	**0.095** (0.013)
		0.2	CRC	0.199 (0.010)	15.200 (0.433)	67.639 (1.669)	8.008 (0.233)	5.987 (0.226)
			GeoCRC	0.200 (0.009)	**7.938** (0.367)	**24.377** (1.029)	**1.600** (0.031)	**0.035** (0.008)
	UNet	0.1	CRC	0.101 (0.008)	88.428 (0.456)	166.625 (0.753)	10.909 (0.321)	10.894 (0.322)
			GeoCRC	0.099 (0.007)	**30.878** (0.419)	**81.335** (1.220)	**2.964** (0.055)	**1.601** (0.055)
		0.2	CRC	0.199 (0.008)	87.487 (0.437)	163.073 (0.578)	30.977 (1.000)	30.944 (0.998)
			GeoCRC	0.202 (0.010)	**28.022** (0.362)	**79.060** (0.824)	**3.417** (0.033)	**1.894** (0.037)
	PSPNet	0.1	CRC	0.101 (0.007)	39.965 (1.110)	110.559 (1.898)	4.760 (0.096)	3.581 (0.099)
			GeoCRC	0.102 (0.007)	**30.995** (5.413)	**47.654** (6.989)	**1.142** (0.057)	**0.164** (0.049)
		0.2	CRC	0.200 (0.011)	16.951 (0.356)	52.638 (1.202)	2.749 (0.040)	1.418 (0.036)
			GeoCRC	0.200 (0.008)	**15.282** (3.756)	**33.385** (5.218)	**1.369** (0.150)	**0.096** (0.025)
	SINet	0.1	CRC	0.101 (0.005)	7.700 (0.165)	29.624 (0.738)	3.175 (0.049)	1.130 (0.032)
			GeoCRC	0.101 (0.007)	**7.075** (0.215)	**18.050** (0.672)	**2.140** (0.034)	**0.218** (0.015)
		0.2	CRC	0.201 (0.013)	5.780 (0.143)	22.947 (0.626)	2.933 (0.042)	0.845 (0.031)
			GeoCRC	0.200 (0.013)	**5.105** (0.123)	**15.969** (0.479)	**2.226** (0.028)	**0.210** (0.016)

Table 1: Quantitative comparison of standard CRC and our proposed GeoCRC. Performance is evaluated across four datasets (Polyp, BRISC, ACDC, and Fire) and four base models at risk levels $\alpha \in \{0.1, 0.2\}$.

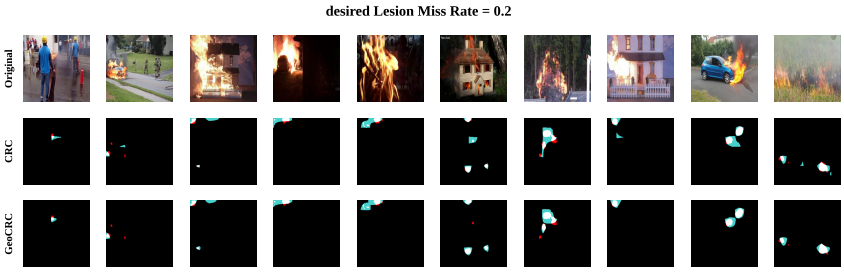


Figure 3: Qualitative comparison of CRC and GeoCRC prediction sets at a significance level of $\alpha = 0.2$.

Hit Thresh	α	Method	Lesion Miss Rate	ASSD ↓	HD95 ↓	Comp ↓	FP Comp ↓
0.8	0.1	CRC	0.096 (0.026)	85.282 (1.587)	167.871 (2.576)	3.456 (0.747)	2.527 (0.711)
		GeoCRC	0.095 (0.024)	**28.003** (4.035)	**54.185** (4.301)	**1.120** (0.027)	**0.187** (0.025)
	0.2	CRC	0.201 (0.034)	33.115 (3.625)	64.484 (4.363)	1.920 (0.336)	0.977 (0.319)
		GeoCRC	0.196 (0.023)	**18.074** (2.092)	**43.202** (3.433)	**1.177** (0.022)	**0.234** (0.027)
0.85	0.1	CRC	0.097 (0.015)	84.516 (1.223)	168.066 (1.696)	2.274 (0.172)	1.421 (0.170)
		GeoCRC	0.097 (0.021)	**30.300** (3.942)	**56.417** (5.428)	**1.105** (0.023)	**0.184** (0.027)
	0.2	CRC	0.199 (0.030)	33.548 (3.190)	84.900 (10.502)	3.204 (0.747)	2.245 (0.736)
		GeoCRC	0.205 (0.035)	**19.219** (1.743)	**44.258** (3.333)	**1.172** (0.024)	**0.231** (0.025)
0.9	0.1	CRC	0.099 (0.022)	81.799 (1.393)	168.796 (1.895)	1.676 (0.067)	0.877 (0.064)
		GeoCRC	0.090 (0.024)	**34.209** (5.652)	**61.061** (7.121)	**1.092** (0.022)	**0.171** (0.018)
	0.2	CRC	0.199 (0.030)	49.842 (10.098)	131.704 (21.739)	5.821 (1.153)	4.830 (1.140)
		GeoCRC	0.195 (0.022)	**20.680** (1.862)	**46.064** (3.273)	**1.160** (0.018)	**0.222** (0.023)
0.95	0.1	CRC	0.097 (0.018)	80.216 (0.972)	169.617 (1.960)	1.464 (0.050)	0.736 (0.038)
		GeoCRC	0.109 (0.032)	**35.344** (4.891)	**62.018** (5.837)	**1.086** (0.022)	**0.177** (0.018)
	0.2	CRC	0.203 (0.030)	81.205 (4.433)	168.363 (2.413)	6.486 (0.802)	5.475 (0.799)
		GeoCRC	0.200 (0.035)	**22.552** (1.280)	**48.148** (2.376)	**1.155** (0.023)	**0.211** (0.018)

Table 2: Quantitative results of the ablation study on different hit thresholds and α values. Experiments were conducted on the Polyp dataset using the PSPNet model. The best performing metrics are highlighted in bold.

Method	Lesion Miss Rate	ASSD ↓	HD95 ↓	Comp ↓	FP Comp ↓
ScoreGeo	0.095 ± 0.033	81.335 ± 1.650	168.223 ± 2.915	1.488 ± 0.233	0.823 ± 0.162
PatchScoreGeo	0.093 ± 0.021	81.174 ± 1.538	168.667 ± 2.644	1.452 ± 0.180	0.795 ± 0.127
BestPatch (Center)	0.098 ± 0.028	66.632 ± 4.502	103.954 ± 6.964	$\mathbf{1.000} \pm 0.000$	0.302 ± 0.045
BestPatch (Region)	0.095 ± 0.029	65.903 ± 5.108	102.701 ± 7.407	$\mathbf{1.000} \pm 0.000$	0.283 ± 0.046
GeoCRC	0.102 (0.028)	**48.934** (5.575)	**84.738** (6.887)	1.103 (0.038)	**0.244** (0.033)

Table 3: Quantitative results of the ablation study. Experiments were conducted on the Polyp dataset using the PSPNet model. Results are reported as mean $\pm$ standard deviation. The best performing mean metrics are highlighted in bold.

calibration set, while GeoCRC divides it equally into an isotonic regression mapping split and a calibration split. All experiments are repeated over 20 random splits. We employ four standard segmentation architectures as base models: UNet (Ronneberger et al., 2015), PSPNet (Zhao et al., 2017), DeepLabV3+ (Chen et al., 2018), and SINet (Fan et al., 2020). The models are trained using standard hyperparameters, with an Adam optimizer and a learning rate of 5×10^{-5}. We set η to 1.

5.2 Evaluation Metrics

We assess segmentation performance using the following metrics. For brevity, let $\widehat{C}_i = \widehat{C}(X_i)$ denote the prediction set for image X_i.

- **Lesion Miss Rate:** The fraction of ground-truth lesions not adequately covered by $\widehat{C}_i$:

$$\text{Lesion Miss Rate} = \frac{1}{K_i} \sum_{k=1}^{K_i} \mathbb{I}\big(|\widehat{C}_i \cap G_{ik}| < \eta |G_{ik}|\big).$$

- **ASSD:** Average Symmetric Surface Distance. Let ∂G_i and $\partial \widehat{C}_i$ be the boundary sets, and $d(u, S) = \min_{v \in S} \|u - v\|_2$:

$$\text{ASSD} = \frac{\sum_{u \in \partial G_i} d(u, \partial \widehat{C}_i) + \sum_{v \in \partial \widehat{C}_i} d(v, \partial G_i)}{|\partial G_i| + |\partial \widehat{C}_i|}.$$

- **HD95:** The 95th percentile of all point-to-set surface distances between ∂G_i and $\partial \widehat{C}_i$:

$$\text{HD95} = P_{95}\big(\{d(u, \partial \widehat{C}_i)\}_{u \in \partial G_i} \cup \{d(v, \partial G_i)\}_{v \in \partial \widehat{C}_i}\big).$$

- **Components:** The total number of predicted connected components, $\widehat{K}_i$, reflecting the degree of spatial fragmentation.

- **FP Components:** The number of predicted components with less than τ_{FP} (set to 0.05) overlap with the ground truth:

$$\text{FP Components} = \sum_{k=1}^{\widehat{K}_i} \mathbb{I}\big(|\widehat{G}_{ik} \cap G_i| < \tau_{\text{FP}}|\widehat{G}_{ik}|\big).$$

As shown in the table 1, both methods successfully bound the expected risk, with the empirical Lesion Miss Rate closely aligning with the target α. However, standard CRC achieves this at the cost of severe over-segmentation, evidenced by high ASSD and HD95. In contrast, GeoCRC significantly improves spatial coherence. This demonstrates that our geodesic formulation preserves the intrinsic topological manifold of targets, yielding much tighter prediction boundaries.

Furthermore, as shown in the figures 2 and 3, standard CRC frequently generates isolated, fragmented false-positive regions, which degrades downstream clinical utility. By growing prediction sets from reliable seeds via calibration-guided geodesic propagation, GeoCRC drastically reduces these artifacts. Across all benchmarks, GeoCRC consistently suppresses scattered noise while maintaining rigorous instance-level coverage guarantees.

5.3 Ablation Experiment

To evaluate the robustness of our proposed GeoCRC under varying degrees of stringency for lesion coverage, we conduct an ablation study on the hit threshold η. The hit threshold dictates the minimum overlap fraction required for a ground-truth lesion to be considered successfully detected. We vary $\eta \in \{0.8, 0.85, 0.90, 0.95\}$ to simulate increasingly rigorous clinical requirements and report the performance on the Polyp dataset using the PSPNet architecture.

As detailed in Table 2, the proposed GeoCRC method exhibits outstanding stability. Leveraging uncertainty-aware geodesic propagation, our approach fundamentally suppresses fragmented spatial artifacts and only permits the expansion of prediction regions along the topological manifold of lesion targets. Regardless of the preset value of η, GeoCRC maintains spatial contiguity of predicted regions, consistently achieves lower ASSD and HD95, yields near-optimal counts of connected components, and generates very few false positives. These results verify that GeoCRC possesses excellent robustness against hyperparameter variations.

Furthermore, to isolate the effect of seed generation, we compare four RawCalibrator-based variants: ScoreGeo (pixel thresholding), PatchScoreGeo (5×5 smoothing), BestPatch (Center) (single maximum pixel), and BestPatch(Region) (5×5 window around the maximum). As detailed in Table 3, our full GeoCRC, equipped with isotonic calibration and multi-seed geodesic propagation, attains the lowest ASSD and HD95, preserves near-optimal component counts, and suppresses false positives, demonstrating the essential synergy of calibrated multi-seed expansion.

6 Conclusion

In this paper, we identified the spatial and structural limitations of CRC in image segmentation and proposed GeoCRC to achieve rigorous instance-level risk control. By replacing naive pixel-wise thresholding with calibration-guided geodesic propagation, GeoCRC effectively disentangles spatial region construction from statistical calibration. Our approach leverages Isotonic Regression-calibrated probabilities to guide the expansion of prediction sets, ensuring spatial coherence and mitigating the severe fragmentation typical of prior methods. We mathematically prove that GeoCRC preserves the distribution-free coverage guarantees of standard CRC under the exchangeability assumption. Extensive experiments across multiple benchmarks confirm that GeoCRC strictly bounds the expected instance-level miss rate while yielding highly coherent, interpretable prediction regions with significantly reduced false-positive noise.

AI USE STATEMENT

Generative AI tools were not used as an important, original, or non-standard component of the core methodology. Generative AI tools were used only for language editing and formatting assistance. All AI-assisted material was reviewed by the authors, who take responsibility for the final content, claims, and artifacts.

ETHICS STATEMENT

This work introduces a GeoCRC framework to provide rigorous safety guarantees for image segmentation in high-stakes domains such as medical diagnosis and disaster monitoring. All experiments utilize publicly available benchmark datasets without involving new human subjects, and thus IRB approval was not required. GeoCRC is strictly designed as a decision-support tool to assist human experts by bounding instance-level miss risks, rather than serving as a fully autonomous agent for critical decisions. Furthermore, while our method provides distribution-free theoretical guarantees under the exchangeability assumption, practitioners should remain aware of potential performance variations under severe out-of-distribution shifts.

REPRODUCIBILITY STATEMENT

To ensure the full reproducibility of our work, we provide comprehensive theoretical, algorithmic, and empirical details. A rigorous mathematical proof of the finite-sample marginal risk guarantees under exchangeability is established in Theorem 1 of Section 4. Additionally, complete implementation details—including binned isotonic probability mapping, multi-source geodesic propagation, and risk calibration—are fully outlined through the pseudocode algorithms in Appendix B. Comprehensive experimental configurations, including dataset split ratios, software versions, and hardware specifications, are documented in Appendix A, and all source codes alongside trained model checkpoints will be released publicly upon acceptance.

REFERENCES

M. Aktaş. Fire segmentation dataset - kaggle, 2023.

Tareq Aldirawi, Yun Li, and Wenge Guo. Non-monotonicity in conformal risk control. *arXiv preprint arXiv:2604.01502*, 2026.

Anastasios N Angelopoulos. Conformal risk control for non-monotonic losses. *arXiv preprint arXiv:2602.20151*, 2026.

Anastasios Nikolas Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=33XGfHLtZg.

Leonard Bereska, Anjany Sekuboyina, Diana Waldmannstetter, Bjoern Menze, Benedikt Wiestler, and Ben Glocker. Sacp: Spatially-adaptive conformal prediction in uncertainty quantification of medical image segmentation. In *Medical Imaging with Deep Learning*, 2025.

Jorge Bernal, Nima Tajkbaksh, Francisco Javier Sanchez, Bogdan J Matuszewski, Hao Chen, Lequan Yu, Quentin Angermann, Olivier Romain, Bjørn Rustad, Ilangko Balasingham, et al. Comparative validation of polyp detection methods in video colonoscopy: results from the miccai 2015 endoscopic vision challenge. *IEEE transactions on medical imaging*, 36(6):1231–1249, 2017.

Olivier Bernard, Alain Lalande, Clement Zotti, Frederick Cervenansky, Xin Yang, Pheng-Ann Heng, Irem Cetin, Karim Lekadir, Oscar Camara, Miguel Angel Gonzalez Ballester, et al. Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: is the problem solved? *IEEE transactions on medical imaging*, 37(11):2514–2525, 2018.

Vincent Blot, Anastasios Nikolas Angelopoulos, Michael Jordan, and Nicolas JB Brunel. Automatically adaptive conformal risk control. In *International Conference on Artificial Intelligence and Statistics*, pp. 19–27. PMLR, 2025.

Jelle Brunekreef, Ethan Marcus, and Eric Nalisnick. Kandinsky conformal prediction: Efficient calibration of image segmentation algorithms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22891–22900, 2024.

Danhui Chen, Ziquan Liu, Chuxi Yang, Dan Wang, Yan Yan, Yi Xu, and Xiangyang Ji. Conformalsam: Unlocking the potential of foundational segmentation models in semi-supervised semantic segmentation with conformal prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 24045–24055, 2025.

Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 801–818, 2018.

Deng-Ping Fan, Ge-Peng Ji, Guolei Sun, Ming-Ming Cheng, Jianbing Shen, and Ling Shao. Camouflaged object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Amirreza Fateh, Yasin Rezvani, Sara Moayedi, Sadjad Rezvani, Fatemeh Fateh, Mansoor Fateh, and Vahid Abolghasemi. Brisc: Annotated dataset for brain tumor segmentation and classification. *Scientific Data*, 2026.

Paul Fischer, Alexander M. Wundram, Lisa M. Koch, Michael Zapf, and Klaus Maier-Hein. Subgroup-specific risk-controlled dose estimation in radiotherapy. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2024.

Mélanie Gaillochet, Christian Desrosiers, and Hervé Lombaert. Anatomically-aware conformal prediction for medical image segmentation with random walks. *arXiv preprint arXiv:2601.18997*, 2026.

Debesh Jha, Pia H Smedsrud, Michael A Riegler, Pål Halvorsen, Thomas De Lange, Dag Johansen, and Håvard D Johansen. Kvasir-seg: A segmented polyp dataset. In *International conference on multimedia modeling*, pp. 451–462. Springer, 2019.

Rui Luo, Jie Bao, Xiaoyi Su, Wen Jung Li, and Suqun Cao. Enhancing image-conditional coverage in segmentation: Adaptive thresholding via differentiable miscoverage loss. In *The Fourteenth International Conference on Learning Representations*, 2026. URL `https://openreview.net/forum?id=Gd2AiWes1J`.

Luca Mossina, Joseba Dalmau, and Léo Andéol. Conformal semantic image segmentation: Post-hoc quantification of predictive uncertainty. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3574–3584, 2024.

Luca Mossina, Caroline Friedrich, and Sébastien Gerchinovitz. Conformal prediction for image segmentation using morphological prediction sets. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2025.

Yaniv Romano, Matteo Sesia, and Emmanuel Candes. Classification with valid and adaptive coverage. *Advances in neural information processing systems*, 33:3581–3591, 2020.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Juan Silva, Aymeric Histace, Olivier Romain, Xavier Dray, and Bertrand Granado. Toward embedded detection of polyps in wce images for early diagnosis of colorectal cancer. *International journal of computer assisted radiology and surgery*, 9(2):283–293, 2014.

Nima Tajbakhsh, Suryakanth R Gurudu, and Jianming Liang. Automated polyp detection in colonoscopy videos using shape and context information. *IEEE transactions on medical imaging*, 35(2):630–644, 2015.

Jacopo Teneggi, Matthew Tivnan, Joseph W. Stayman, and Jeremias Sulam. How to trust your diffusion model: A convex optimization approach to conformal risk control. arXiv preprint arXiv:2302.03791, 2023.

Jacopo Teneggi, Joseph W. Stayman, and Jeremias Sulam. Conformal risk control for semantic uncertainty quantification in computed tomography. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2025.

Alexander Timans, Christoph-Nikolas Straehle, Kaspar Sakmann, and Eric Nalisnick. Adaptive bounding box uncertainties via two-step conformal prediction. In *European Conference on Computer Vision*, pp. 363–398. Springer, 2024.

David Vázquez, Jorge Bernal, F Javier Sánchez, Gloria Fernández-Esparrach, Antonio M López, Adriana Romero, Michal Drozdzal, and Aaron Courville. A benchmark for endoluminal scene segmentation of colonoscopy images. *Journal of healthcare engineering*, 2017(1):4037190, 2017.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*, volume 29. Springer, 2005.

Håkan Wieslander, Philip J Harrison, Gabriel Skogberg, Sonya Jackson, Markus Fridén, Johan Karlsson, Ola Spjuth, and Carolina Wählby. Deep learning with conformal prediction for hierarchical analysis of large-scale whole-slide tissue images. *IEEE journal of biomedical and health informatics*, 25(2):371–380, 2020.

Anna M Wundram, Paul Fischer, Michael Mühlebach, Lisa M Koch, and Christian F Baumgartner. Conformal performance range prediction for segmentation output quality control. In *International Workshop on Uncertainty for Safe Utilization of Machine Learning in Medical Imaging*, pp. 81–91. Springer, 2024.

Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2881–2890, 2017.

A Appendix A: Experimental Details

Reproducibility Statement: The source code, trained model checkpoints, data splits, and scripts used to generate all reported results will be made publicly available upon acceptance.

Implementation Details: All experiments were conducted on a workstation equipped with an NVIDIA GeForce RTX 4090 GPU with 24 GB of memory. Our implementation used Python 3.9.23, PyTorch 2.7.1 with CUDA 11.8, cuDNN 9.1, NumPy 1.26.4, SciPy 1.13.0, scikit learn 1.6.1, and scikit-image 0.24.0. All base segmentation models were trained for 20 epochs.

B Appendix B: Core Implementation Details and Pseudocode

This appendix provides anonymous Python-style pseudo-code for the core implementation of GeoCRC.

B.1 Probability Mapping and Geodesic Construction

B.2 CRC Calibration and Evaluation

Algorithm 2 Python Pseudo-Code for GeoCRC: Isotonic Probability Mapping.

```python
class BinnedIsotonicMapping:
    def __init__(self, centers, mapped_values):
        self.centers = centers
        self.mapped_values = mapped_values

    @classmethod
    def fit(cls, probability_maps, masks, max_pixels, num_bins, seed):
        # Pool all score-label pairs from the mapping split.
        scores = concatenate([p.ravel() for p in probability_maps])
        labels = concatenate([y.ravel() for y in masks])

        # Randomly subsample pixels when the pooled set is large.
        if len(scores) > max_pixels:
            scores, labels = random_subsample(
                scores, labels, max_pixels, seed
            )

        # Quantile binning of the raw prediction scores.
        edges = unique(quantile(scores, linspace(0, 1, num_bins + 1)))
        bin_id = digitize(scores, edges[1:-1])
        bin_id = clip(bin_id, 0, len(edges) - 2)

        counts = bincount(bin_id, minlength=len(edges) - 1)
        positives = bincount(
            bin_id, weights=labels, minlength=len(edges) - 1
        )
        nonempty = counts > 0

        centers = weighted_bin_means(scores, bin_id, counts)[nonempty]
        bin_means = positives[nonempty] / counts[nonempty]
        weights = counts[nonempty]

        # Fit a monotone isotonic probability mapping.
        mapped_values = isotonic_regression(
            x=centers, y=bin_means, sample_weight=weights,
            lower_bound=0.0, upper_bound=1.0
        )
        return cls(centers, mapped_values)

    def transform(self, probability_map):
        x = clip(probability_map, self.centers[0], self.centers[-1])
        return linear_interpolate(x, self.centers, self.mapped_values)
```

Algorithm 3 Python Pseudo-Code for GeoCRC: Core Extraction and Geodesic Distance Construction.

```
def extract_core_regions(mapped_probability, core_threshold,
                         fallback_percentile, min_component_size):
    # Select high-confidence pixels as initial seed candidates.
    core = mapped_probability >= core_threshold

    # Use a percentile fallback when no primary core exists.
    if not any(core):
        positive_values = mapped_probability[mapped_probability > 0]
        if len(positive_values) == 0:
            return zeros_like(mapped_probability, dtype=bool)
        cutoff = percentile(positive_values, fallback_percentile)
        core = mapped_probability >= cutoff

    # Remove isolated seed components below the minimum size.
    labels, num_components = connected_components(core, connectivity=8)
    component_sizes = component_areas(labels, num_components)
    valid_ids = ids_with_size_at_least(
        component_sizes, min_component_size
    )
    filtered_core = is_in(labels, valid_ids)

    # Retain the unfiltered core if filtering removes every seed.
    if any(filtered_core):
        return filtered_core
    return core


def compute_geodesic_distance(mapped_probability, core_threshold,
                              fallback_percentile, min_component_size):
    core = extract_core_regions(
        mapped_probability, core_threshold,
        fallback_percentile, min_component_size
    )
    if not any(core):
        return full_like(mapped_probability, inf), core

    # High mapped probability gives low traversal cost.
    cost = -log(clip(mapped_probability, 1e-6, 1.0))

    # Multi-source minimum-cost propagation from all core pixels.
    seed_points = coordinates_where(core)
    distance = multi_source_geodesic_shortest_path(cost, seed_points)
    distance[core] = 0.0

    return distance, core
```

Algorithm 4 Python Pseudo-Code for GeoCRC: Connected-Component Risk and RawCRC Threshold Calibration.

```python
def component_miss_rate(region, mask, hit_threshold):
    labels, num_lesions = connected_components(mask, connectivity=8)
    if num_lesions == 0:
        return None

    missed = 0
    for lesion_id in range(1, num_lesions + 1):
        lesion = (labels == lesion_id)
        coverage = mean(region[lesion])
        missed += (coverage < hit_threshold)

    return missed / num_lesions


def empirical_component_risk(regions, masks, hit_threshold):
    losses = []
    for region, mask in zip(regions, masks):
        loss = component_miss_rate(region, mask, hit_threshold)
        if loss is not None:
            losses.append(loss)
    return mean(losses) if len(losses) > 0 else 0.0


def crc_bound(empirical_risk, num_positive_images):
    return (
        num_positive_images / (num_positive_images + 1)
    ) * empirical_risk + 1.0 / (num_positive_images + 1)


def calibrate_raw_threshold(probability_maps, masks, alpha,
    hit_threshold):
    n = sum(any(mask) for mask in masks)
    candidate_scores = unique(concatenate([
        probability_map[mask]
        for probability_map, mask in zip(probability_maps, masks)
        if any(mask)
    ]))
    candidates = concatenate([[-inf], candidate_scores])

    # Find the largest threshold satisfying the finite-sample CRC bound
        .
    best_threshold = -inf
    for threshold in candidates:
        regions = [p > threshold for p in probability_maps]
        risk = empirical_component_risk(regions, masks, hit_threshold)
        if crc_bound(risk, n) <= alpha:
            best_threshold = threshold

    return best_threshold
```

Algorithm 5 Python Pseudo-Code for GeoCRC: Geodesic Radius Calibration.

```
def calibrate_geodesic_radius(distance_maps, masks, alpha,
    hit_threshold):
   n = sum(any(mask) for mask in masks)

   candidate_radii = unique(concatenate([
      distance[mask & isfinite(distance)]
      for distance, mask in zip(distance_maps, masks)
      if any(mask)
   ]))
   candidates = concatenate([candidate_radii, [inf]])

   # Find the smallest radius satisfying the finite-sample CRC bound.
   for radius in candidates:
      regions = [distance <= radius for distance in distance_maps]
      risk = empirical_component_risk(regions, masks, hit_threshold)
      if crc_bound(risk, n) <= alpha:
         return radius

   return inf
```

Algorithm 6 Python Pseudo-Code for GeoCRC: Split-wise Calibration and Held-Out Inference.

```
def run_geocrc_trial(probability_maps, masks, alpha, test_ratio, seed):
    # Create disjoint mapping, CRC, and held-out test splits.
    mapping_split, crc_split, test_split = random_three_way_split(
        probability_maps, masks, test_ratio, seed
    )

    # Fit the isotonic mapping on the mapping split only.
    mapper = BinnedIsotonicMapping.fit(
        mapping_split.probabilities, mapping_split.masks,
        max_pixels=MAX_PIXELS, num_bins=NUM_BINS, seed=seed
    )

    # Construct geodesic distance fields on the CRC split.
    crc_distances = []
    for probability_map in crc_split.probabilities:
        mapped_probability = mapper.transform(probability_map)
        distance, _ = compute_geodesic_distance(
            mapped_probability, CORE_THRESHOLD,
            FALLBACK_PERCENTILE, MIN_COMPONENT_SIZE
        )
        crc_distances.append(distance)

    # RawCRC uses both non-test splits; GeoCRC uses the CRC split.
    raw_threshold = calibrate_raw_threshold(
        mapping_split.probabilities + crc_split.probabilities,
        mapping_split.masks + crc_split.masks,
        alpha, HIT_THRESHOLD
    )
    geodesic_radius = calibrate_geodesic_radius(
        crc_distances, crc_split.masks, alpha, HIT_THRESHOLD
    )

    results = []
    for probability_map, mask in zip(
        test_split.probabilities, test_split.masks
    ):
        raw_prediction = probability_map > raw_threshold

        mapped_probability = mapper.transform(probability_map)
        distance, _ = compute_geodesic_distance(
            mapped_probability, CORE_THRESHOLD,
            FALLBACK_PERCENTILE, MIN_COMPONENT_SIZE
        )
        geodesic_prediction = distance <= geodesic_radius

        results.append(evaluate_predictions(
            raw_prediction, geodesic_prediction, mask
        ))

    return results
```