This WACV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Uncertainty Estimation in Instance Segmentation with Star-convex Shapes

Qasim M. K. Siddiqui, Sebastian Starke, and Peter Steinbach
Hemholtz-Zentrum Dresden-Rossendorf
Dresden, Germany
q.siddiqui@hzdr.de, s.starke@hzdr.de, p.steinbach@hzdr.de

Abstract

Instance segmentation has witnessed promising advancements through deep neural network-based algorithms. However, these models often exhibit incorrect predictions with unwarranted confidence levels. Consequently, evaluating prediction uncertainty becomes critical for informed decision-making. Existing methods primarily focus on quantifying uncertainty in classification or regression tasks, lacking emphasis on instance segmentation. Our research addresses the challenge of estimating spatial certainty associated with the location of instances with star-convex shapes. Two distinct clustering approaches are evaluated which compute spatial and fractional certainty per instance employing samples by the Monte-Carlo Dropout or Deep Ensemble technique. Our study demonstrates that combining spatial and fractional certainty scores yields improved calibrated estimation over individual certainty scores. Notably, our experimental results show that the Deep Ensemble technique alongside our novel radial clustering approach proves to be an effective strategy. Our findings emphasize the significance of evaluating the calibration of estimated certainties for model reliability and decision-making.

1. Introduction

In the past decade, deep neural networks have made significant advancements and have become prevalent in the field of computer vision, achieving impressive state-of-the-art performance and even competing with human-level results in supervised learning tasks [13, 20, 26, 31]. However, these achievements are primarily observed in closed-set conditions, where the testing data exhibits overlapping characteristics with the training data. In contrast, a noticeable decline in performance occurs in open-set conditions [15, 40], where the testing data possess characteristics not present in the training data. In such scenarios, neural network models often make wrong predictions with high confidence [1], which raises critical concerns about the safety and reliability of deploying these models, particularly in applications where perception errors can have severe consequences [39]. To address this challenge, one promising approach is to explore the model's epistemic uncertainty, which arises due to a lack of data [19]. High epistemic uncertainty in predictions can potentially indicate open-set errors [24, 25], enabling models to identify and handle such detections appropriately.

Bayesian Neural Networks [2, 23, 28] offer a means to estimate epistemic uncertainty. However, their practical application is limited due to the higher computational cost and training complexity involved. To address these challenges, Monte-Carlo Dropout was introduced by Gal *et al.* in 2015 [6] as a computationally feasible approximation to Bayesian neural networks, providing uncertainty estimates for a model's confidence scores.

While Monte-Carlo Dropout offers a feasible technique for estimating uncertainty, it often requires extensive hyperparameter tuning to obtain well-calibrated predictive uncertainty estimates [7]. In response, Deep Ensemble techniques, a non-Bayesian solution [21], were introduced, which yield well-calibrated predictive uncertainty estimates with minimal hyperparameter tuning at the cost of requiring multiple model training.

Recently, Monte-Carlo Dropout and Deep Ensemble techniques have shown promising results in uncertainty estimation for image classification and regression tasks [6, 21]. However, their application to instance segmentation, which involves localizing and classifying multiple objects within a scene, remains relatively underexplored.

Our research aims to bridge this gap by applying Monte-Carlo Dropout and Deep Ensemble techniques to the specific domain of instance segmentation using the StarDist model [35, 41]. The effectiveness of these techniques in estimating certainty in instance segmentation was verified through extensive experiments on three different datasets. Specifically, our research evaluates and compares the effectiveness of both Monte-Carlo Dropout and Deep Ensemble techniques for the StarDist model. Furthermore, we investigate the effect of dropout rates and the location of the

dropout layer within the StarDist model for Monte-Carlo Dropout certainty estimation in instance segmentation.

By addressing these research gaps, this study enhances the reliability and robustness of deep neural networks by providing well-calibrated predictive certainty estimates for instance segmentation. These advancements enable more informed decision-making and improve the performance of deep neural networks in practical applications of instance segmentation with the StarDist model.

2. Related Work

Uncertainty estimation in deep neural networks has been a subject of extensive research, focusing on two types of uncertainty: aleatoric uncertainty and epistemic uncertainty [19].

Aleatoric uncertainty also known as data uncertainty, captures the inherent noise present in the data itself. It represents the statistical or sensory noise that cannot be reduced even with an increase in the amount of collected data [18]. Aleatoric uncertainty can be directly learned using neural networks by applying a distribution over the model's output, enabling the model to capture varying levels of uncertainty for different inputs [18].

Epistemic uncertainty also referred to as model uncertainty, arises from parameter ambiguity and limited knowledge about the model. It reflects the uncertainty in the model's predictions and requires additional techniques to estimate. One approach is to place a prior distribution over the model's parameters and analyze the variability of this distribution given the available data [2, 4, 28]. Estimating epistemic uncertainty is particularly crucial for safety-critical systems and models trained with small datasets, as it helps identify situations that lie beyond the model's training data.

Estimating epistemic uncertainty in deep neural networks presents challenges not encountered in traditional machine learning algorithms [36]. The inherent inability of deep neural networks to accurately quantify uncertainty has led to investigations into alternative approaches. Bayesian Neural Networks offer a means of predicting output uncertainty [2], but their computational demands often render them impractical [24, 38].

To address this, Gal *et al*. [6] proposed using dropout in the last layers of deep learning models during test time, with multiple forward passes, to approximate Bayesian inference over network parameters. This dropout-based sampling approach better captures input-specific uncertainty within the model. It has been successfully applied to tasks such as active learning for image categorization [9], melanoma identification [11], and object recognition using LiDAR data [5].

While dropout-based sampling initially focused on deep neural network classification tasks, Miller *et al*. [24] extended the concept to Single Shot Detection (SSD) for object detection [22]. In this more complex task, each forward pass generates multiple object detections that need to be matched and merged. By clustering detections based on spatial and semantic similarity, ambiguous detections are rejected, leading to improved object detection performance in both closed and open-set scenarios. Miller *et al*. [25] further explored this approach, evaluating alternative strategies for merging detections while incorporating dropout-based sampling in object detection.

Morrison *et al*. [27] built upon these foundations to address pixel-wise masked instance segmentation. They adapted the technique proposed in [24] to perform probabilistic instance segmentation. Leveraging the Mask-RCNN network [14], they employed dropout-based sampling during inference, following the principles established in Srivastava *et al*. [38] and Miller *et al*. [24]. This methodology resulted in a well-calibrated uncertainty estimation.

Overall, these studies highlight the significance of dropout-based sampling techniques in capturing and quantifying uncertainty in deep neural networks, spanning various tasks ranging from classification to object detection and instance segmentation.

Our study extends the work by Morrison *et al*. [27] to the StarDist model, which is a novel approach to instance segmentation, addressing the limitations of existing methods like Mask-RCNN in handling crowded instances [35, 41].

The StarDist model utilizes a U-Net architecture as its building block, which is well-suited for image segmentation tasks and has demonstrated state-of-the-art performance in biomedical image segmentation [34].

2.1. Revisiting StarDist

The StarDist model proposes the localization of convex-shaped instances using star-convex polygons, yielding satisfactory outcomes for densely populated scenarios [35]. The fundamental component of the StarDist model is depicted in Figure 1.

The StarDist model, akin to object detection approaches [17, 22, 32], employs a star-convex polygon prediction for each pixel. Specifically, for every pixel (x, y) within the scalar field of the input image size $(X, Y, channel)$, we predict the radial distances $\{r^i_{x,y}\}^n_{i=1}$ to the object boundary. These predictions are made along a predefined set of n radial directions with equidistant angles. The model independently predicts whether each pixel is part of an object, focusing on polygon proposals from pixels with sufficiently high object probability $d_{x,y}$. By considering these dense polygon candidates and their associated object probabilities, we employ non-maximum suppression to derive the final collection of polygons. Each polygon represents a distinct object instance.

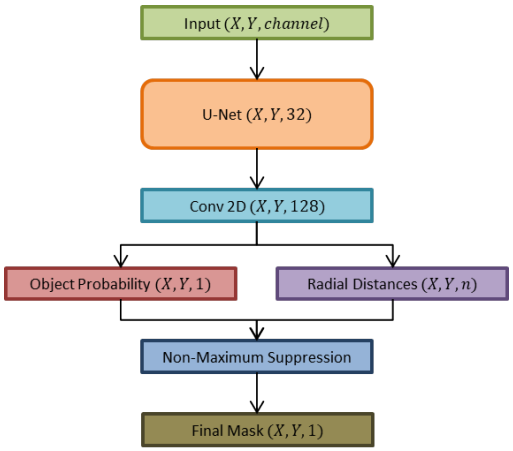


Figure 1. The fundamental structure of the StarDist model involves an input image with dimensions $(X, Y, channel)$. Within this model, a U-net is employed, featuring two output layers: Object Probability $d_{x,y} \in D$ and Radial Distances $\{r^i_{x,y}\}^n_{i=1} \in R$, both of which constitute scalar fields matching the input image size. The process concludes with the derivation of the Final Mask after applying non-maximum suppression to both the Object Probability (D) and Radial Distances (R).

3. Methods

In our research, the StarDist model is adapted to incorporate the Deep Ensemble [21] and Monte-Carlo Dropout [6] techniques during inference, enabling probabilistic instance segmentation similar to prior works of Miller *et al.* [24] and Morrison *et al.* [27]. These sampling techniques utilize multiple output samples obtained from F forward passes to assess the certainty associated with the model's predictions.

While sampling techniques can be employed for estimating epistemic certainty in tasks like classification or semantic segmentation by simply averaging the samples, instance segmentation poses unique challenges. In instance segmentation, accurate association and clustering of different detection samples are required to identify multiple instances within an image, as depicted in Figure 2. To account for spatial certainty, similar to the approaches in Miller *et al.* [24] and Miller *et al.* [25], certainty estimates are obtained by integrating instances from successive forward passes, as depicted in Figure 2.

The StarDist model comprises two sets of outputs: (a) the final mask and (b) pixel-wise object probability predictions and radial distance, each of these output sets can be utilized individually to calculate the certainty associated with the model's predictions. In the subsequent sections, we present approaches that leverage these output sets to quantify the model's certainty.

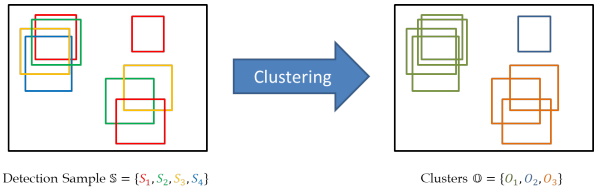


Figure 2. An illustration of image clustering based on predictions obtained from $F = 4$ forward passes, with $|\ S_1\ | = 3$, $|\ S_2\ | = 2$, $|\ S_3\ | = 2$, and $|\ S_4\ | = 1$. The aim is to group predicted instances that correspond to the same object into distinct clusters $\mathbb{O}$. In this example, $|\ O_1\ | = 4$, $|\ O_2\ | = 1$, and $|\ O_3\ | = 3$, where each cluster $O_m \mid m \in \{1, 2, 3\}$ represents an instance.

3.1. Pixel Approach

In the first approach for calculating certainty, we utilize the final mask containing the collection of polygons obtained after non-maximum suppression, which have the same dimensions as the input image. Each pixel in the final mask is assigned a $positive$ integer value if it is within a polygon representing an instance or $zero$ if it belongs to the background.

3.1.1 Clustering Technique

To estimate certainty with the final mask from the StarDist model, we adopt a clustering technique inspired by prior work on object detection and instance segmentation [24, 25, 27].

In this approach, each forward pass of input through the StarDist model generates a set of predicted instances $S = \{P_1, P_2, ..., P_K\}$, where K represents the number of predicted instances, which may vary across different forward passes. In the case of binary class prediction, $P_k \mid k \in \{1, 2, ..., K\}$ represents the pixel-wise mask of the instance for a given input image. By performing F forward passes of the input image through the model, we obtain a set of samples $\mathbb{S} = \{S_1, S_2, ..., S_F\}$, where each $S_f = \{P_{1_f}, P_{2_f}, ..., P_{k_f}\}$ contains a set of predictions, where $k_f \mid f \in \{1, 2, ..., F\}$ represents the number of predicted instances, which may vary across different forward passes.

Based on their spatial affinity, the predictions from the set of samples obtained through all forward passes $\mathbb{S}$ are grouped into individual clusters $\mathbb{O} = \{O_1, O_2, ..., O_M\}$. Ideally, each cluster $O_m \mid m \in \{1, 2, ..., M\}$ should represent a single instance within the image.

To perform the clustering of instances, we employ the Basic Sequential Algorithmic Scheme (BSAS) [25], where predictions P_{k_f} are sequentially assigned to clusters O_m if their mask Intersection-over-Union (IoU)[1] exceeds a

[1] See Supplementary Material for Intersection-over-Union (IoU) calculation.

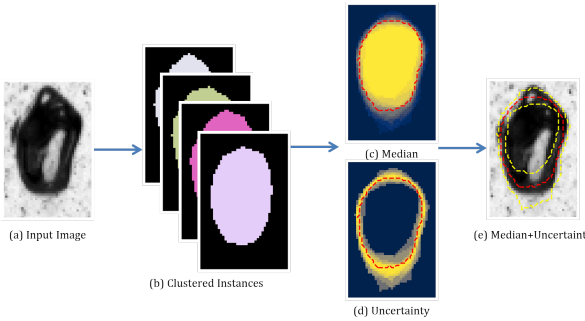


Figure 3. An illustration of the process of visualizing the uncertainty in the Pixel Approach. (a) Input Image, (b) Clustered instance (O_m), (c) Mean calculated for each pixel, (d) Standard deviation calculated for each pixel, and (e) Red polygons illustrate the median cluster prediction $\overline{P}_m$, and the region between the two yellow polygons indicates the uncertainty of the instance.

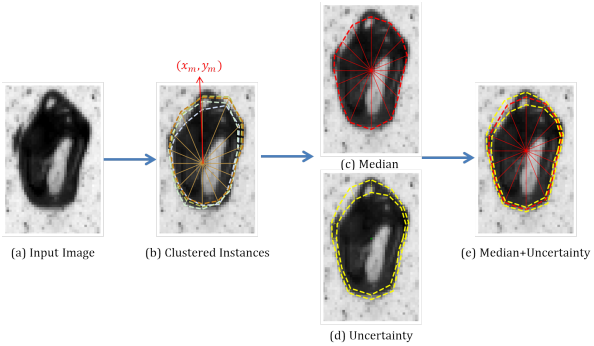


Figure 4. An illustration of the process of visualizing the uncertainty in the Pixel Approach. (a) Input Image, (b) Clustered instance (O_m) for polygon center (x_m, y_m), (c) Median calculated across all the radial distances, (d) 2.5^{th} and 97.5^{th} percentile across all the radial distances, and (e) Red polygons illustrate the median cluster prediction $\overline{P}_m$, the inner yellow polygons are indicated by the 2.5^{th} percentile of the radial distances, while the outer yellow polygons are indicated by the 97.5^{th} percentile of the radial distances.

threshold value θ_{IoU}. If the IoU between a prediction and every prediction in an existing cluster is above the threshold θ_{IoU}, the prediction is added to that cluster. If no existing cluster matches the prediction, a new cluster is created. The IoU threshold θ_{IoU} is subject to optimization. However we fix $\theta_{IoU} = 0.5$ [27]. The clustering algorithm in the Pixel Approach is given in the Supplementary Material.

3.1.2 Visualization of Prediction and Uncertainty

To visualize the median cluster prediction $\overline{P}_m$ for cluster O_m, we calculate the median across all pixels $(x_o, y_o) \mid o = \{1, 2, ..., \mid O_m \mid\}$. The pixel value of the median cluster prediction $\overline{P}_m$ is set to $integer$ if it corresponds to an instance and $zero$ otherwise. In Figure 3, the red polygons illustrate the median cluster prediction $\overline{P}_m$.

To visualize uncertainty, we determine the mean and the standard deviation values across all pixels $(x_o, y_o) \mid o = \{1, 2, ..., \mid O_m \mid\}$ within cluster O_m. A pixel with a low standard deviation signifies low uncertainty that it belongs to the instance while increasing standard deviation values reflect increasing uncertainty. In Figure 3, the contour of the standard deviation values is represented by the yellow line, conveying its associated uncertainty level. Pixels enclosed by the inner yellow polygons indicate a low degree of uncertainty in belonging to a specific instance, while those outside the outer yellow polygons indicate a low degree of uncertainty in not belonging to a specific instance in question. The region between the two yellow polygons indicates the uncertainty of a specific instance.

3.2. Radial Approach

In this approach for calculating certainty, we leverage the output structure of the StarDist model, consisting of pixel-wise object probability predictions and radial distance, which are obtained before non-maximum suppression. We will be using the term $DenseOutput$ ($G = \{D, R\}$) to refer to the scalar field of object probabilities ($d_{x,y} \in D$) and radial distances ($\{r^n_{x,y}\}^n_{i=1} \in R$) for improved clarity and conciseness.

The clustering approach of instance masks based on IoU from the Pixel Approach is replaced by identifying the centers of the instances from the sample mean of the set of $DenseOutput$, and clustering instances whose object probability ($d_{x,y} \in D$) is above a predefined threshold value (θ_d).

3.2.1 Clustering Technique

In this approach, each forward pass of an input through the StarDist model generates the $DenseOutput$ $G = \{D, R\}$. By performing F forward passes of the input image through the model, we obtain a set of $DenseOutput$ samples $\mathbb{G} = \{G_1, G_2, ..., G_F\}$, where $G_f = \{D_f, R_f\} \mid f \in \{1, 2, ..., F\}$.

To perform the clustering of instances, we identify the center for each instance by taking the mean of the set $\mathbb{G}$ to parameterize it into a single $DenseOutput$ $\mu_G = \{\mu_D, \mu_R\}$, where μ_D and μ_R represent the mean object probability and mean radial distance across the samples $\mathbb{G}$. Non-maximum suppression is applied to μ_G to obtain the set of polygon centers with the highest object probability, $\mathbb{C} = \{(x_1, y_1), (x_2, y_2), ..., (x_M, y_M)\}$.

The combination of the polygon center $(x_m, y_m) \mid m \in \{1, 2, ..., M\}$ and the radial distances at the corresponding polygon centers $\{\{r^n_{x_m,y_m}\}^n_{i=1}\}_f \in R_f \mid f \in \{1, 2, ..., F\}$ a set of predictions instance are generated $S =$

$\{P_1, P_2, ..., P_M\}$ for each forward pass. By performing F forward passes of the input image through the model, we obtain a set of samples $\mathbb{S} = \{S_1, S_2, ..., S_F\}$, where each $S_f = \{P_{1_f}, P_{2_f}, ..., P_{M_f}\}$ contains a set of predictions, where $M_f \mid f \in \{1, 2, ..., F\}$ represents the number of predicted instances, which in this case is fixed at M across different forward passes.

The predictions from the set of samples obtained through all forward pass $\mathbb{S}$ are grouped into individual clusters $\mathbb{O} = \{O_1, O_2, ..., O_M\}$. Ideally, each cluster $O_m \mid m \in \{1, 2, ..., M\}$ should represent a single object within the image with $(x_m, y_m) \mid m \in \{1, 2, ..., M\}$ as individual polygon center.

The predictions P_{m_f} are sequentially assigned to the cluster O_m if the object probability $\{d_{x_m,y_m}\}_f$ exceeds a threshold value θ_d. The object probability threshold θ_d is subject to optimization. However, for consistency with the Pixel Approach 3.1, we set $\theta_d = 0.5$. The clustering algorithm in the Radial Approach is given in the Supplementary Material.

3.2.2 Visualization of Prediction and Uncertainty

To visualize the median cluster prediction $\overline{P}_m$ for cluster O_m, we calculate the median across all the radial distances $\{\{r^n_{x_m,y_m}\}^n_{i=1}\}_o \mid o = \{1, 2, ..., \mid O_m \mid\}$. In Figure 4, the red polygons illustrate the median cluster prediction $\overline{P}_m$.

We visualize uncertainty, by determining the 2.5^{th} and 97.5^{th} percentile across all the radial distances $\{\{r^n_{x_m,y_m}\}^n_{i=1}\}_o \mid o = \{1, 2, ..., \mid O_m \mid\}$. In Figure 4, the inner yellow polygons are indicated by the 2.5^{th} percentile of the radial distances, while the outer yellow polygons are indicated by the 97.5^{th} percentile of the radial distances for the specific instance. The region between the two yellow polygons indicates the spatial uncertainty of a specific instance.

3.3. Certainty Quantification

Once all predictions $P_{\cdot_f}$ are grouped into clusters $\mathbb{O}$, we can quantify the certainty of our model's predictions using two scores for each cluster $O_m \mid m \in \{1, 2, ..., M\}$:

(i) Spatial Certainty: This score informs us about the model's confidence in the location of each instance. The spatial certainty for each cluster O_m is computed by averaging the intersection-over-union (IoU) between the median cluster prediction $\overline{P}_m$ and the prediction P_j of each of the $\mid O_m \mid$ predictions within the cluster O_m. The spatial certainty ranges from 0 to 1, where 1 indicates high certainty and 0 indicates low certainty.

$$c_{spl}(O_m) = \frac{1}{\mid O_m \mid} \sum_{j=1}^{|0_m|} IoU(P_j, \overline{P}_m) \tag{1}$$

(ii) Fractional Certainty: This score represents the model's confidence in detecting an instance across multiple forward passes. As not every forward pass may detect all instances, the fractional certainty for each cluster O_m is computed as the fraction of forward passes in which the instance is detected. Similarly, the fractional certainty ranges from 0 to 1, where 1 indicates the instance was predicted in all forward passes and 0 indicates the instance was not detected in any forward pass.

$$c_{frac}(O_m) = \frac{\mid O_m \mid}{N} \tag{2}$$

(iii) Hybrid Certainty: It has been observed by Morrison *et al*. [27] that combining the two certainty scores results in well-calibrated certainty estimates compared to using them separately.

$$c_{hyb}(O_m) = c_{spl}(O_m) \cdot c_{frac}(O_m) \tag{3}$$

3.4. Evaluating Certainty

Calibration serves as a valuable metric to evaluate the estimated certainty of the model. Model calibration refers to the accuracy and precision of the certainty score in indicating when the model is likely to make errors. Calibration is crucial for interpretability and building trust in users who are consuming model prediction. A well-calibrated model provides reliable certainty scores that align with its predictive expected accuracy [12].

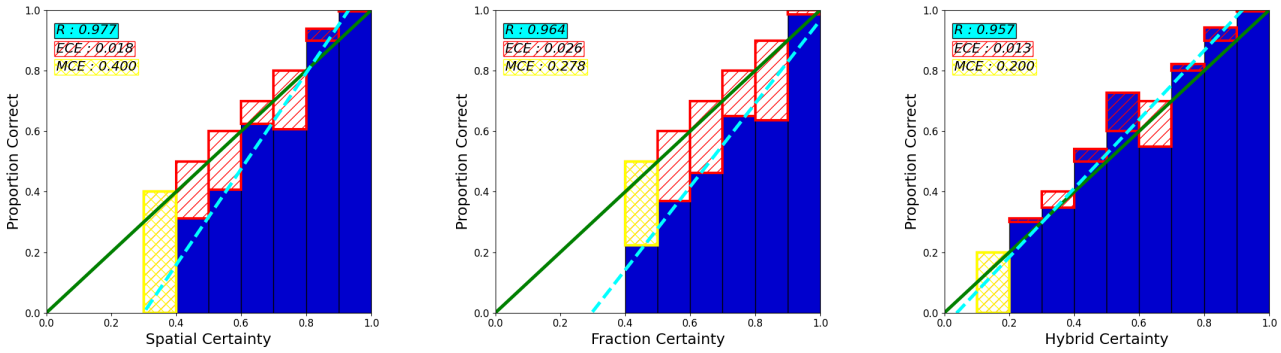
3.4.1 Calibration Diagram

A calibration diagram visualizes the expected sample accuracy as a function of the certainty score. To create the diagram, predictions are divided into B interval bins of size $1/B$. The expected accuracy and average certainty score within each bin are computed. If a model is perfectly calibrated, the expected accuracy and certainty score should be equal for each bin [3, 29]. This is visualized in Figure 5, which compares the three certainty quantification scores for the StarDist model with the Radial Approach and Monte-Carlo Dropout $d_{rate} = 0.8$, with $F = 20$.

3.4.2 Calibration Error

Scalar calibration summaries are practical for evaluating calibration diagrams. The commonly used metrics are the Pearson correlation coefficient, the Expected Calibration Error, and the Maximum Calibration Error, which can be derived from the calibration diagrams.

Pearson Correlation Coefficient (Pearson's R) measures the linear correlation between two sets of data. In this context, it quantifies the correlation between the identity function and the bin scores. The coefficient ranges from



(a) (b) (c)

Figure 5. Calibration diagrams depicting the estimation of certainty scores for the Bubble dataset are presented, employing a bin interval of $B = 10$. Panels (a), (b), and (c) show spatial certainty (c_{spl}), fractional certainty (c_{frac}), and hybrid certainty (c_{hyb}) scores, respectively. These scores are calculated using the Radial Approach and Monte-Carlo Dropout with a dropout rate of $d_{rate} = 0.8$, and $F = 20$ forward passes. Notably, the hybrid certainty scores (c_{hyb}) demonstrate superior calibration compared to individual certainty scores across three calibration error metrics: Pearson Correlation Coefficient (R), Expected Calibration Error (ECE), and Maximum Calibration Error (MCE).

-1 to 1, with 0 indicating no correlation, negative values indicating negative correlation, and positive values indicating positive correlation [33].

Expected Calibration Error (ECE) approximates the expected difference between the certainty score and expected accuracy. It involves dividing predictions into evenly spaced bins (similar to Calibration Diagram Figure 5) and averaging the accuracy difference within each bin. Expected Calibration Error provides a measure of miscalibration, with a perfectly calibrated model having an Expected Calibration Error of zero [30].

Maximum Calibration Error (MCE) summarizes miscalibration by measuring the maximum difference between the certainty score and expected accuracy. This metric is useful in high-risk applications where minimizing the worst-case difference is crucial. Maximum Calibration Error is defined as the maximum absolute difference across all bins. Like Expected Calibration Error, a perfectly calibrated model will have a Maximum Calibration Error of zero [30].

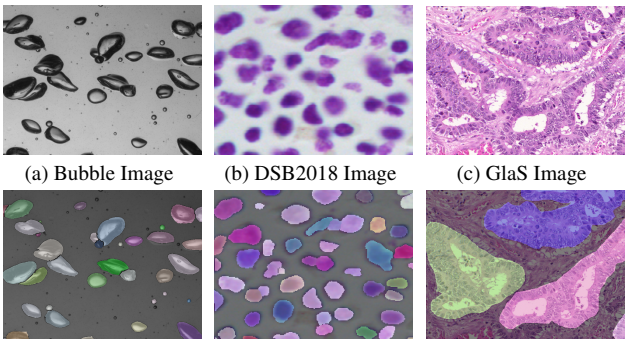
4. Experiments and Results

4.1. Datasets

Bubble: The Dataset consists of spherical, ellipsoidal, and wobbling bubbles, which are typically encountered in air-water bubbly flows [16]. The dataset includes 414 manually annotated images of dimensions $256 \times 512 \times 1$, along with corresponding ground truth masks.

DSB2018: Manually annotated real microscopy images of cell nuclei from the 2018 Data Science Bowl [10]. The dataset includes 670 manually annotated images of dimensions $256 \times 256 \times 3$, along with corresponding ground truth masks.

GlaS: Annotated gland segmentation images of Hematoxylin and Eosin stained slides [37].The dataset includes 165 manually annotated images of dimensions $522 \times 775 \times 3$, along with corresponding ground truth masks.



(a) Bubble Image (b) DSB2018 Image (c) GlaS Image

(d) Bubble True Mask (e) DSB2018 True Mask (f) GlaS True Mask

Figure 6. Panels (a) to (c) depict the input image for each dataset. Panels (d) to (f) represent the true mask for each dataset.

The Figures 6 present examples of input images along with their corresponding true masks for each dataset. The training set comprises 80% of the images, while the remaining 20% is allocated for validation. We trained our model on the training set and evaluated the estimated certainty on the validation set.

4.2. Training and Inference

Training: The StarDist model is employed, featuring three convolutional blocks for down/upsampling. Each block consists of two convolutional layers with $32 \cdot 2^z$ $(z = 0, 1, 2)$ filters of size 3×3 and ReLU activation. Following the final upsampling feature layer, an additional convolutional layer with 128 channels and ReLU activation, suggested by Schmidt et al. [35], is incorporated. A

$n = 16$ channel convolutional layer is used to compute the star-convex polygon distance in the StarDist model. The models are trained for 400 epochs.

For the Deep Ensemble technique to estimate certainty, multiple StarDist models are trained using the entire training dataset, with each model randomly initialized.

In contrast, for the Monte-Carlo Dropout technique to estimate certainty, a single StarDist model is trained with the entire training dataset, supplemented by dropout layers after the last upsampling feature map of the StarDist model. Additionally, in the Monte-Carlo Dropout technique, the impact of the dropout rate on the model's certainty is investigated. The model's certainty is evaluated for dropout rates, $d_{rate} \in \{0.1, 0.5, 0.8\}$. [2]

Inference: For the Deep Ensemble technique, the validation dataset is used for each of the F StarDist models, and the two approaches (Pixel Approach 3.1 and the Radial Approach 3.2) are utilized to cluster and estimate model's certainty. Similarly, in the Monte-Carlo Dropout technique, the validation dataset is processed through a single StarDist model F times to estimate the model's certainty.

4.3. Quality of Certainty Score

The calibration diagrams in Figure 5 with bin size $B = 10$, show the calibration quality of the different certainty scores using the Radial Approach with Monte-Carlo Dropout technique ($d_{rate} = 0.8$, $F = 20$) on the validation set for the Bubble dataset. The diagrams compare hybrid certainty score (c_{hyb}), spatial certainty score (c_{spl}), and fractional certainty score (c_{frac}).

We observe that the hybrid certainty score c_{hyb} exhibits better calibration as the certainty score closely approximates the expected accuracy (i.e. the bins align to the identity function). The hybrid certainty score considers both spatial similarities and the frequency of instance detections, making it well-suited for more robust comparisons. These comprehensive considerations of factors position the hybrid certainty score as a valuable metric for further comparative analyses.

We observe a consistent pattern on the DSB2018 and GlaS datasets, as illustrated in Figure 1. and Figure 2. of the Supplementary Material.

4.4. Effects of Forward Passes on Certainties Quality

The influence of the number of forward passes F on the calibration of the hybrid certainty (c_{hyb}) was also assessed and the results are visualized in Figure 7. The calibration errors, measured by Pearson's R, Expected Calibration Error (ECE), and Maximum Calibration Error (MCE), are plotted against F.

[2] See Supplementary Material for the choice of the three dropout rates.

We observe a convergence of errors as the number of forward passes increases, which aligns with the principles of the Central Limit Theorem. Additionally, gradually small changes are observed in the error trends after around 20 to 30 forward passes for the Monte-Carlo Dropout technique and 10 models for the Deep Ensemble technique. Furthermore, distinct convergence behaviors are observed for each dropout rate.

We observe consistent behavior when employing both the Pixel Approach and the Radial Approach on both the DSB2018 and GlaS datasets, as depicted in Figure 3. and Figure 4. of the Supplementary Material.

5. Discussion

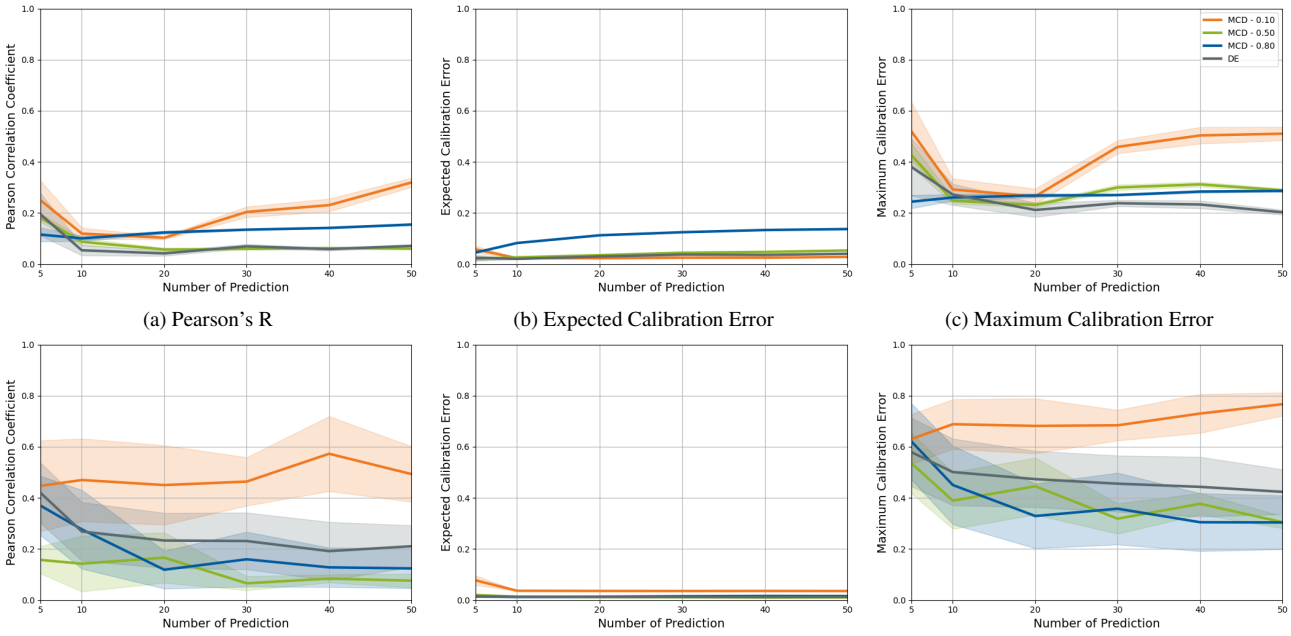
In this work, we introduced and evaluated two approaches to augment instance segmentation predictions by the StarDist model with certainty estimates. The Pixel Approach adapts the work of Morrison *et al*. [27] to estimate the model's certainty by clustering similar instances based on the IoU score. However, the clustering algorithm used does not scale well as the data increases due to the quadratic complexity. To address this, the Radial Approach implements a more efficient clustering algorithm leveraging the unique output structure of the StarDist model, scaling the clustering algorithm linearly with the data.

Both Deep Ensemble and Monte-Carlo Dropout techniques yield well-calibrated hybrid certainty estimates, surpassing the spatial and fractional certainty scores individually. The calibration errors decrease for $F = 10$ models in the Deep Ensemble technique, aligning with the findings of Lakshminarayanan et al. [21] in classification and regression tasks.

The results presented in Figure 7 further support Lakshminarayanan *et al*.'s [21] claim that the Deep Ensemble technique requires minimal hyperparameter tuning to achieve well-calibrated certainty estimates. They also validate Gal *et al*.'s [7] assertion that the Monte-Carlo Dropout technique's certainty estimates are not calibrated and that the dropout rate must be adjusted to match the model's certainty.

The elevated calibration errors observed in Figure 2. and Figure 4. of the Supplementary Material for the GlaS dataset are due to the dataset's incompatibility with the StarDist model. This leads to higher certainty scores for incorrect predictions, highlighting the critical importance of assessing the calibration of estimated certainties. Such evaluations are pivotal for ensuring model reliability and informed decision-making.

The randomness of the calibration error in Figure 6. Figure 7. and Figure 8. of the Supplementary Material suggests no discernible relationship between the dropout layer's location and its effect on model calibration. Consequently, a search over the locations is necessary for achieving a well-



(a) Pearson's R (b) Expected Calibration Error (c) Maximum Calibration Error

(d) Pearson's R (e) Expected Calibration Error (f) Maximum Calibration Error

Figure 7. Plot showing calibration errors as a function of the number of forward passes for the Monte-Carlo Dropout and Deep Ensemble techniques (calibration errors as a function of the number of models in the case of Deep Ensemble). Panels (a) to (c) depict certainty estimates using the Pixel Approach, while panels (d) to (f) represent certainty estimates using the Radial Approach. Notably, the Deep Ensemble technique exhibits faster convergence of calibration errors compared to the Monte-Carlo Dropout technique. Additionally, distinctive convergence patterns are observed for each dropout rate.

calibrated model.

In conclusion, the most efficient combination for estimating certainty in instance segmentation tasks by StarDist model entails using Deep Ensemble and the Radial Approach, employing $F = 10$ models. This configuration yields well-calibrated results with minimal tuning requirements and an efficient clustering algorithm.

Several possibilities for future expansion of our work exist. One intriguing avenue is exploring the connection between this work and active learning. The obtained certainty information could facilitate the creation of a new dataset containing unknown objects. Ground truth labels for this dataset could be acquired from users, allowing the model's capabilities to continuously adapt to its operating environment through ongoing training.

Additionally, explore the inter-annotator variability as a means to juxtapose the uncertainty inherent in human judgments with the models' epistemic uncertainty. This comparison will shed light on how well the models' uncertainty estimates align with the diversity of human perceptions and annotations, offering deeper insights into the model's capacity to capture uncertainties akin to those observed in human decision-making.

Furthermore, implementing the Concrete Dropout [8] variant for instance segmentation tasks could be beneficial. This principled extension of dropout enables the tuning of dropout rates, leading to better-calibrated uncertainty estimates in large models while avoiding the coarse and computationally expensive search over dropout rates [7].

6. Summary

This study addresses certainty estimation in instance segmentation using deep neural networks. While deep neural networks often produce overly confident incorrect predictions, accurate certainty assessment is crucial for informed decisions. Unlike existing methods mainly focused on classification and regression tasks, we pioneer spatial certainty estimation for instances with star-convex shapes.

This study introduces the Radial Approach, a novel clustering method, which combined with Deep Ensemble based certainty sampling provides efficient and well-calibrated certainty estimates.

This research reiterates the significance of calibration for accurate certainty assessment. Calibrating certainty estimates is vital for reliable decision-making and model trustworthiness.

References

[1] Abhijit Bendale and Terrance Boult. Towards open set deep networks, 2016. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 1563-1572, doi: 10.1109/CVPR.2016.173. 1

[2] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks, 2015. In Proceedings of the 32nd International Conference on Machine Learning (ICML 2015). 1, 2

[3] Morris H. DeGroot and Stephen E. Fienberg. The comparison and evaluation of forecasters, 1983. Journal of the Royal Statistical Society. Series D (The Statistician), 32(1/2), 12–22. https://doi.org/10.2307/2987588. 5

[4] John Denker and Yann LeCun. Transforming neural-net output levels to probability distributions, 1991. In Advances in Neural Information Processing Systems 3. 2

[5] Di Feng, Xiao Wei, Lars Rosenbaum, Atsuto Maki, and Klaus Dietmayer. Deep active learning for efficient training of a lidar 3d object detector, 2019. 2

[6] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning, 2016. In ICML. 1, 2, 3

[7] Yarin Gal, Jiri Hron, and Alex Kendall. Concrete dropout, 2017. Advances in Neural Information Processing Systems 30, pp. 3581-3590. Curran Associates, Inc. 1, 7, 8

[8] Yarin Gal, Jiri Hron, and Alex Kendall. Concrete dropout, 2017. 8

[9] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data, 2017. In Proceedings of the 34th International Conference on Machine Learning- Volume 70, pages 1183-1192. JMLR. org. 2

[10] Allen Goodman, Anne Carpenter, Elizabeth Park, jlefman nvidia, Josette BoozAllen, Kyle, Maggie, Nilofer, Peter Sedivec, and Will Cukierski. 2018 data science bowl, 2018. 6

[11] Marc Gorriz, X. Giro i Nieto, Axel Carlier, and Emmanuel Faure. Cost-effective active learning for melanoma segmentation, 2017. In ML4H: Machine Learning for Health Workshop at NIPS 2017. 2

[12] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks, 2016. https://doi.org/10.48550/arXiv.1706.04599. 5

[13] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, and Andrew Y. Ng. Deep speech: Scaling up end-to-end speech recognition, 2014. 1

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn, 2017. 2

[15] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks, 2017. in International Conference on Machine Learning (ICML). 1

[16] Hendrik Hessenkemper, Sebastian Starke, Yazan Atassi, Thomas Ziegenhein, and Dirk Lucas. Bubble identification from images with machine learning methods, 2022. 6

[17] Saumya Jetley, Michael Sapienza, Stuart Golodetz, and Philip H.S. Torr. Straight to shapes: Real-time detection of encoded shapes, 2017. In: CVPR. 2

[18] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision?, 2017. CoRR, abs/1703.04977. 2

[19] Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? does it matter?, 2009. Structural Safety, 31 (2):105-112. 1, 2

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks, 2012. In Advances in neural information processing systems. 1

[21] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles, 2016. arXiv preprint arXiv:1612.01474. 1, 3, 7

[22] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector, 2016. pages 21-37. 2

[23] David JC MacKay. A practical bayesian framework for backpropagation networks, 1992. Neural computation, 4(3):448-472. 1

[24] Dimity Miller, Lachlan Nicholson, Feras Dayoub, and Niko Sünderhauf. Dropout sampling for robust object detection in open-set conditions, 2018. In IEEE International Conference on Robotics and Automation (ICRA). IEEE. 1, 2, 3

[25] Dimity Miller, Lachlan Nicholson, Feras Dayoub, and Niko Sünderhauf. Evaluating merging strategies for sampling-based uncertainty techniques in object detection, 2019. In IEEE International Conference on Robotics and Automation (ICRA). 1, 2, 3

[26] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning, 2013. 1

[27] Doug Morrison, Anton Milan, and Nontas Antonakos. Estimating uncertainty in instance segmentation using dropout sampling, 2019. CVPR 2019 Robotic Vision Probabilistic Object Detection Challenge. 2, 3, 4, 5, 7

[28] Radford M Neal. Bayesian learning for neural networks, 1995. PhD thesis, University of Toronto. 1, 2

[29] Niculescu-Mizil, Alexandru, Caruana, and Rich. Predicting good probabilities with supervised learning, 2005. In ICML, pp. 625-632. 5

[30] Naeini Mahdi Pakdaman, Cooper Gregory F, and Hauskrecht Milos. Obtaining well calibrated probabilities using bayesian binning, 2015. In AAAI, pp. 2901. 6

[31] Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J Guibas, and Jascha Sohl-Dickstein. Deep knowledge tracing, 2015. 1

[32] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection, 2016. In: CVPR. 2

[33] Vijay K. Rohatgi and A. K. Md. Ehsanes Saleh. An introduction to probability and statistics, 2001. Wiley. 6

[34] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. arXiv:1505.04597 [cs.CV]. 2

[35] Uwe Schmidt, Martin Weigert, Coleman Broaddus, and Gene Myers. Cell detection with star-convex polygons, 2018. Medical Image Computing and Computer Assisted Intervention - MICCAI 2018 - 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II, Pages: 265–273. 1, 2, 6

[36] Burr Settles. Active learning literature survey, 2009. Technical report, University of Wisconsin-Madison Department of Computer Sciences. 2

[37] Korsuk Sirinukunwattana, Josien P. W. Pluim, Hao Chen, Xiaojuan Qi, Pheng-Ann Heng, Yun Bo Guo, Li Yang Wang, Bogdan J. Matuszewski, Elia Bruni, Urko Sanchez, Anton Böhm, Olaf Ronneberger, Bassem Ben Cheikh, Daniel Racoceanu, Philipp Kainz, Michael Pfeiffer, Martin Urschler, David R. J. Snead, and Nasir M. Rajpoot. Gland segmentation in colon histology images: The glas challenge contest, 2016. 6

[38] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, , and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting, 2014. Journal of Machine Learning Research, 15, 1929-1958. 2

[39] Niko Sünderhauf, Oliver Brock, Walter Scheirer, Raia Hadsell, Dieter Fox, Jürgen Leitner, Ben Upcroft, Pieter Abbeel, Wolfram Burgard, Michael Milford, and Peter Corke. The limits and potentials of deep learning for robotics, 2018. The International Journal of Robotics Research, vol. 37, no. 4-5, pp. 405-420. 1

[40] Antonio Torralba and Alexei A. Efros. Unbiased look at dataset bias, 2011. in Computer Vision and Pattern Recognition (CVPR). IEEE, pp. 1521-1528. 1

[41] Martin Weigert, Uwe Schmidt, Robert Haase, Ko Sugawara, and Gene Myers. Star-convex polyhedra for 3d object detection and segmentation in microscopy, 2020. The IEEE Winter Conference on Applications of Computer Vision (WACV). 1, 2