SUPER-Net: Trustworthy Image Segmentation via Uncertainty Propagation in Encoder-Decoder Networks

Giuseppina Carannante[1], Nidhal C. Bouaynaya[1], Hassan M. Fathallah-Shaykh[2], Ghulam Rasool[3], Dimah Dera[4]

[1]*Department of Electrical and Computer Engineering, Rowan University, Glassboro, NJ, USA*
[2]*Department of Neurology, University of Alabama at Birmingham School of Medicine, Birmingham, AL, USA*
[3]*Machine Learning Department, Moffit Cancer Center, Tampa, FL, USA*
[4]*Chester F. Carlson Center for Imaging Science, Rochester Institute of Technology, Rochester, NY, USA*

Abstract

Deep Learning (DL) holds great promise in reshaping the industry owing to its precision, efficiency, and objectivity. However, the brittleness of DL models to noisy and out-of-distribution inputs is ailing their deployment in sensitive fields. Current models often lack uncertainty quantification, providing only point estimates. We propose SUPER-Net, a Bayesian framework for trustworthy image segmentation via uncertainty propagation. Using Taylor series approximations, SUPER-Net propagates the mean and covariance of the model's posterior distribution across nonlinear layers. It generates two outputs simultaneously: the segmented image and a pixel-wise uncertainty map, eliminating the need for expensive Monte Carlo sampling. SUPER-Net's performance is extensively evaluated on MRI and CT scans under various noisy and adversarial conditions. Results show that SUPER-Net outperforms state-of-the-art models in robustness and accuracy. The uncertainty map identifies low-confidence areas affected by noise or attacks, allowing the model to self-assess segmentation reliability, particularly when errors arise from noise or adversarial examples.

Keywords: Bayesian deep learning, Encoder-decoder networks, Reliability, Segmentation, Trustworthiness, Uncertainty estimation

1. Introduction

Driven by the superior performance achieved in many areas, various deep learning (DL) models have been advanced to analyze medical data, e.g., radiological images and pathology slides. Several methods have achieved, if not surpassed, prognosis parity with specialized medical personnel [1]. However, their successful deployment in clinical settings remains limited. While several autonomous algorithms are doubtlessly employed for many everyday tasks — e.g., spam filters for emails or biometrics that unlock our cellphones —, there is a less assertive willingness to utilize the same algorithms for risky, sensitive data, such as medical images.

The main challenge that hinders the widespread and effective use of DL in clinical settings is the lack of reliable and trustworthy predictions [2]. For example, when encountering test examples that differ significantly from its training data, a DL system will still produce a prediction. However, without uncertainty information, there is no way to determine how reliable that prediction is. This concern is further exacerbated by the vulnerability of DL models to adversarial inputs — perturbations that are imperceptible to human observers yet cause a trained DL model to produce erroneous predictions [3]. In the literature, there are studies highlighting the vulnerability of medical models to adversarial perturbations [4]. As a result, DL in medicine is particularly susceptible due to both technical weaknesses and financial incentives [4].

Addressing these challenges requires DL models not only to produce accurate predictions but also to quantify the uncertainty associated with those predictions. Uncertainty Quantification (UQ) serves as a key mechanism for assessing the reliability of predictions, allowing users to be aware of the level of confidence in the models' predictions. UQ could be very useful when the DL model is essentially guessing at random due to excessive noise in the input or possible adversarial attacks. Unfortunately, as most DL models are inherently deterministic, a measure of confidence or uncertainty is not readily available at their output.

Estimating the confidence of a model requires a probabilistic interpretation of the model's parameters, i.e., treating model parameters as random variables endowed with a probability distribution. Through Bayesian inference, the posterior distribution of the

model parameters can be found. At test time, the second moment, i.e., the covariance, of the predictive distribution can serve as a measure of confidence or uncertainty in the predicted output. Several Bayesian models have been developed for the classification and regression problems [5]. Trade-offs between prediction accuracy, confidence estimation, and scalability are at the heart of these different approaches [5]. Recently, Dera *et al.* proposed a variational moments' propagation (VMP) framework that provides a meaningful and scalable framework for uncertainty propagation and estimation in Convolutional neural network (CNN) classifiers [6].

A relatively small amount of work focuses on quantifying uncertainty in pixel-level segmentation tasks using Bayesian DL models. The challenge in learning uncertainty for each pixel arises from propagating high-dimensional posterior distributions of the model's parameters through multiple stages of non-linearities in the encoder-decoder architecture. Furthermore, the model must provide an *instantaneous* uncertainty map at test time, i.e., simultaneously output the prediction (the segmentation) and corresponding pixel-level uncertainty map *without* resorting to expensive Monte Carlo sampling techniques or model averaging (ensemble).

Previous work focused solely on uncertainty quantification in classification neural network models, notably Convolution Neural Networks (CNNs) [6]. The mathematical derivations presented in [6] are not sufficient for uncertainty propagation in encode-decoder-based segmentation neural networks. The challenges in adopting the VMP framework in segmentation lie in the nature of the learning task: semantic segmentation requires the extraction of both global and local contextual information by encoding and then decoding the input data. Consequently, segmentation networks are fundamentally different than classification networks, e.g., CNNs. The mathematical derivations for the decoder part were never presented previously, and the flow of uncertainty from the encoder to the decoder was never considered within an analytical and systematic framework. The decoder presents specific non-linearities and operations, e.g., up-sampling, padding, and concatenation, that require new mathematical derivations to track the propagation of the uncertainty. In addition, our previous work estimated a scalar (or a vector) value of variance that is associated with the predicted class [6]. This work introduces the notion of a dense, pixel-level uncertainty map that

is provided simultaneously along with the predicted segmentation.

In this paper, we develop a VMP framework for segmentation tasks, SUPER-Net, and extensively evaluate it for various medical imaging datasets under various noisy and adversarial conditions. By leveraging key concepts from probability density tracking in nonlinear and non-Gaussian systems [7], we propagate the first and the second moments of the posterior distribution of network parameters through the nonlinear layers of an encoder-decoder type segmentation architecture. The developed approach is tested using various medical segmentation datasets consisting of Magnetic Resonance Images (MRIs) and Computed Tomography (CT) scans. The proposed VMP formulation and the derived mathematical relationships presented in the paper are applicable to various DNN architectures.

The contributions of this paper are summarized as follows:

(1) Formalize a scalable Bayesian framework that *simultaneously* learns pixel-wise prediction and confidence in encoder-decoder segmentation networks by analytically approximating and maximizing the evidence lower bound (ELBO). Using first-order Taylor series approximation, we derived closed-form expressions to propagate the first two moments (mean and covariance) of the posterior distribution of the model parameters given the training data and update them during backpropagation; thus, effectively learning the intrinsic uncertainty of the model. We derive mathematical relations for all operations involved, rendering a method that is adaptable to other models, e.g., Variational Autoencoders, and to other tasks as well.

(2) Develop a Bayesian DL architecture that instantaneously outputs two maps: (1) the segmented image and (2) the uncertainty map of the predicted segmentation. These two maps are delivered simultaneously and *without* requiring any Monte Carlo sampling at inference time. That is, the generated uncertainty was intrinsically learned by the model rather than estimated post-training.

(3) Extensively evaluate the performance of the proposed SUPER-Net for various medical segmentation tasks and under various signal-to-noise ratios (SNRs) and conditions. A thorough robustness analysis is conducted by assessing the performance of the model and uncertainty map under these perturbations of the input data.

2. Related Work

Image segmentation is a fundamental problem in computer vision with applications ranging from medical image analysis to scene understanding for autonomous vehicles. DL techniques, particularly Fully Convolutional Networks (FCNs), have been widely used for pixel-level segmentation [8]. FCNs modify traditional CNN architectures by replacing fully connected layers with upsampling operations to generate segmentation masks. *Encoder-decoder* architectures have since become the dominant paradigm for semantic segmentation [9]. The *encoder* extracts low-dimensional (salient) features of the data, while the *decoder* reconstructs the spacial information to perform pixel-wise classification. Various improvements have been introduced, e.g., skip connections with attention mechanisms [10], dilated convolutions [11], wide contest blocks or compression extraction modules [12].

More recently, Transformer-based architectures have been explored for segmentation, leveraging attention mechanisms to capture long-range dependencies [13]. Readers interested in further details are directed to recent surveys on the application of Transformers to various segmentation tasks [14], particularly within the medical domain [15]. Inspired by the success of foundational models in natural language processing, the Segment Anything Model (SAM) [16] introduced a zero-shot approach to segmentation, which has also been evaluated for medical imaging tasks [17].

These architectural advances, however, focused on improving accuracy which, no doubt, is an important metric but it does not convey the full picture. Reliability, robustness, and trustworthiness are important metrics for these models. An unreliable model can jeopardize the clinical system by exposing it to technical vulnerabilities, financial risks, and even patient harm [4]. In the context of semantic segmentation, there are two main approaches for UQ: Monte Carlo (MC) dropout [18] and model ensemble [19].

MC dropout is widely used due to its simplicity and compatibility with existing NN architectures [18]. The uncertainty information is obtained, at inference time, from the sample variance of multiple MC forward passes through the network. Several studies have applied this technique for various segmentation tasks [20]. For instance, a full-resolution residual network is used for brain segmentation in [21], the QuickNAt

architecture is used in [22], and a $3D$ U-Net is proposed in [23].

In ensemble methods [19], after training multiple networks, usually with random initialization, several segmentation estimates are produced, and their variation is used as a measure of confidence. For example, [24] uses a ResUNet architecture with soft dice loss and two regularization terms to diversify the ensemble members. Authors in [25] generate diversity in the ensemble by considering predictions generated by different architectures and models. Other researchers proposed weighting ensemble members based on sensitivity and precision to improve calibration [26]. Some works combined the two approaches; for example, in [27] ensemble members are generated by changing the dropout rate. Other techniques, e.g., hierarchical probabilistic models [28], Evidential Deep Learning [29] and Normalized Softmax Entropy [30], have also been explored to quantify uncertainty. However, most existing UQ approaches share common limitations.

Post-hoc methods estimate uncertainty only at inference time—using multiple forward passes or MC sampling—rather than integrating it into training. This prevents the model from refining uncertainty estimates based on training data. Moreover, these methods approximate uncertainty through empirical sample variance, which may not reflect true confidence, leading to overconfident incorrect predictions. They are also computationally expensive, requiring multiple forward passes at test time or training multiple models in ensemble methods. Lastly, many approaches lack robustness evaluation, as they are often assessed on clean datasets without considering adversarial attacks or noisy inputs

In contrast, the proposed SUPER-Net framework learns uncertainty during training and outputs simultaneously the predicted segmentation and its uncertainty map. A framework to learn the variance is proposed in [6], but derivations are limited to CNNs, rendering the approach unsuitable for the more complex end-to-end segmentation tasks. Building upon this work, we develop a Bayesian framework that propagates the first and the second moment of the variational posterior distribution across all layers of a segmentation DL model. At test time, the uncertainty in the predicted segmentation is produced by the network as the covariance matrix of the predictive distribution simultaneously alongside the segmentation without resorting to multiple runs.

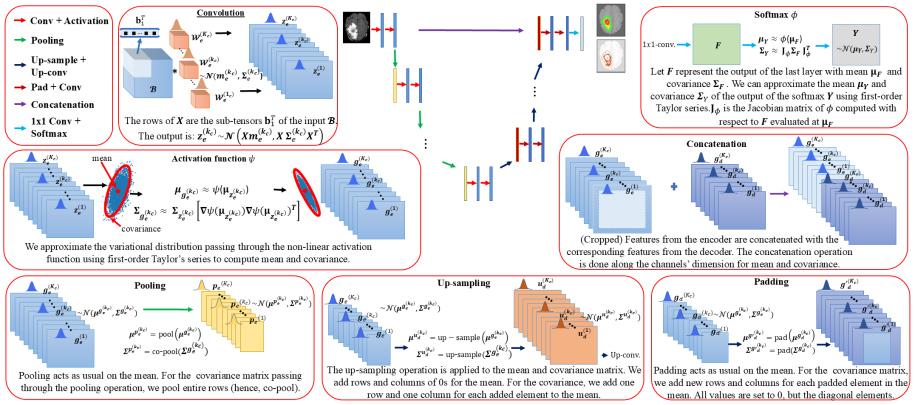


Figure 1: An illustration of the SUPER-Net model, where all mathematical operations are performed on random variables. The mean and covariance matrices are propagated through each operation. The output of SUPER-Net consists of the predicted segmented image and a covariance matrix, which is used to generate the associated uncertainty map.

3. SUPER-Net: Segmentation with Uncertainty Propagation in Encoder-decodeR Networks

We derive a Bayesian framework where the first two moments of the posterior distribution are learned simultaneously by (forward and backward) propagation through the network layers. For scalability and efficiency of the proposed approach, we adopt the Variational Inference (VI) technique, but rather than estimating the expected log-likelihood using expensive Monte Carlo sampling, we approximate its first two moments with a first-order Taylor-series expansion. In the sequel, we present our mathematical results. An illustration of the SUPER-Net model is presented in Figure 1.

3.1. Mathematical Notations

Scalars are represented by lower-case letters, e.g., x, x_i. Vectors are represented by bold lower-case letters, e.g., $\mathbf{y}$. All vectors are column vectors. y_i denotes the i^{th} element of vector $\mathbf{y}$. Matrices are represented by bold upper-case letters, e.g., $\mathbf{A}$. Tr$(\cdot)$ denotes the trace of a matrix, i.e., the sum of its diagonal elements. ^{T} denotes the transpose operator, and vec$(\cdot)$ denotes the vectorization operator. The Hadamard product, i.e., the element-wise product, is denoted with $\odot$, while $\times$ represents the matrix-matrix

or matrix-vector product. Tensors with three or more dimensions are represented by curly bold upper-case letters, e.g., $\boldsymbol{\mathcal{X}}$. If x is a random variable, $\mathbb{E}[x]$ denotes the expected value of x. We use $\boldsymbol{\mathcal{W}}_e^{(k_c)}$ to represent the k_c^{th} convolutional kernel of the c^{th} layer. K_c denotes the total number of kernels in layer c. The subscripts e and d represent the encoder and decoder path operations, respectively.

3.2. Bayesian Deep Learning and Variational Inference

In Bayesian statistics, the unknown parameters are fully characterized by their posterior distribution given the observations. In Bayesian DL, the network parameters Ω are endowed with a prior probability distribution $p(\Omega)$ and all information about the parameters is embedded in the posterior distribution $p(\Omega|\mathcal{D})$ given the (training) data $\mathcal{D} = \{\boldsymbol{\mathcal{X}}^i, \mathbf{y}^i\}_{i=1}^N$. Once the posterior is estimated, the predictive distribution, i.e., the distribution of the test data, can be derived as:

$$p(\mathbf{y}^*|\boldsymbol{\mathcal{X}}^*, \mathcal{D}) = \int p(\mathbf{y}^*|\boldsymbol{\mathcal{X}}^*, \Omega)\; p(\Omega|\mathcal{D})\; d\Omega, \tag{1}$$

where $\boldsymbol{\mathcal{X}}^*$ is the input, $\mathbf{y}^*$ is its corresponding predicted output and $p(\mathbf{y}^*|\boldsymbol{\mathcal{X}}^*, \Omega)$ is the likelihood.

Unfortunately, direct inference of the posterior is intractable due to the large parameter space and nonlinear nature of DL architectures. A popular approximation technique, known as VI, formulates the problem of posterior inference as an optimization problem [31]. The VI approach considers a simple family of distributions over the network parameters and attempts to find a distribution, called the *variational distribution* $q_{\boldsymbol{\theta}}(\Omega)$, within this family that is "close" to the *true* unknown posterior. The notion of distributional closeness is captured by the Kullback-Leibler (KL) divergence, and the optimization is performed with respect to the variational distribution parameters $\boldsymbol{\theta}$:

$$\mathbf{KL}\left(q_{\boldsymbol{\theta}}(\Omega)\middle\|p(\Omega|\mathcal{D})\right) = \int q_{\boldsymbol{\theta}}(\Omega) \log \frac{q_{\boldsymbol{\theta}}(\Omega)}{p(\Omega)p(\mathcal{D}|\Omega)} d\Omega. \tag{2}$$

By rearranging terms in (2), the well-known ELBO objective function is obtained [32]:

$$\mathcal{L}(\boldsymbol{\theta}) = -\,\mathbb{E}_{q_{\boldsymbol{\theta}}(\Omega)}\left[\log(p(\mathcal{D}|\Omega)\right] + \mathbf{KL}\left(q_{\boldsymbol{\theta}}(\Omega)\middle\|p(\Omega)\right). \tag{3}$$

Most Bayesian DL frameworks that use the VI approach sample one set of parameters $\boldsymbol{\theta}$ and perform a deterministic forward pass and backpropagation. The second

moment or the variance of the predictive distribution is obtained post-training using MC samples at inference time [33]. This practice is based on the assumption that the single set of sampled parameters $\boldsymbol{\theta}$ represents the variational distribution $q_{\boldsymbol{\theta}}(\Omega)$ with sufficient accuracy, which has no theoretical grounds [6].

3.3. Encoder Operations

We define a multivariate Gaussian distribution as a prior distribution for all convolution kernels. We assume that kernels are independent within each layer as well as across layers in both the encoder and decoder paths. The independence assumption results in a single additional parameter (variance) for each kernel, limiting the increase in the number of parameters due to the Bayesian formulation. Moreover, independent kernels help extract uncorrelated features and better explore the input space [6].

Convolution Between Input and Network Parameters: The convolution operation in the first layer is performed between the input data (initially assumed deterministic for simplicity) and the network parameters (random variables). We assume that network parameters $\mathcal{W}_e^{(k_1)}$ follow a Gaussian distribution, i.e., $\text{vec}(\mathcal{W}_e^{(k_1)}) \sim \mathcal{N}\left(\mathbf{m}_e^{(k_1)}, \boldsymbol{\Sigma}_e^{(k_1)}\right)$. We write the convolution as a matrix-vector multiplication, where $\mathbf{X}$ denotes the matrix having rows equal to the vectorized sub-tensors of the input $\boldsymbol{\mathcal{X}}$. Then, the convolution operation is expressed as $\mathbf{z}_e^{(k_1)} = \mathbf{X} \times \text{vec}(\mathcal{W}_e^{(k_1)})$, for $k_1 = 1, \cdots, K_1$. Thus, the output of the first convolutional layer follows a Gaussian distribution where the mean and covariance are given by:

$$\mathbf{z}_e^{(k_1)} \sim \mathcal{N}\left(\mathbf{X}\mathbf{m}_e^{(k_1)},\ \mathbf{X}\boldsymbol{\Sigma}_e^{(k_1)}\mathbf{X}^T\right). \tag{4}$$

Convolution Between Two Random Variables: We consider a generic case of convolution between two random variables. Let $\boldsymbol{\mathcal{B}}$ be the incoming input to any convolution layer, except the first layer, i.e., $c \neq 1$. The convolution operation is expressed as a matrix-vector multiplication; however, in this case both the input and the kernels are random tensors. We form $\mathbf{B}$ by vectorizing the sub-tensors of the incoming input $\boldsymbol{\mathcal{B}}$, i.e., $\mathbf{B} = [\mathbf{b}_1^T, \mathbf{b}_2^T, \cdots, \mathbf{b}_J^T]^T$, where $\mathbf{b}_j^T$ represents j^{th} row of $\mathbf{B}$. Let $\boldsymbol{\mu}_{\mathbf{b}_j}$ and $\boldsymbol{\Sigma}_{\mathbf{b}_j}$ represent the mean and covariance of $\mathbf{b}_j$. Then, the output of the convolution is formulated as $\mathbf{z}_e^{(k_c)} = \mathbf{B} \times \text{vec}(\mathcal{W}_e^{(k_c)})$ with $\text{vec}(\mathcal{W}_e^{(k_c)}) \sim \mathcal{N}\left(\mathbf{m}_e^{(k_c)}, \boldsymbol{\Sigma}_e^{(k_c)}\right)$ for $k_c = 1, \cdots, K_c$. Given

that the input $\mathcal{B}$ (feature map) is independent from the subsequent layer kernels, we compute elements of the mean of $\mathbf{z}_e^{(k_c)}$ as the product of the two mean vectors, $\boldsymbol{\mu}_{\mathbf{b}_j}$ and $\mathbf{m}_e^{(k_c)}$, i.e.,

$$[\boldsymbol{\mu}_{\mathbf{z}_e^{(k_c)}}]_j = \boldsymbol{\mu}_{\mathbf{b}_j}^T \mathbf{m}_e^{(k_c)}, \quad j = 1, \cdots, J. \tag{5}$$

The elements of the covariance matrix $\boldsymbol{\Sigma}_{\mathbf{z}_e^{(k_c)}}$ are derived as:

$$\text{Non-diagonal elements } (i \neq j): \quad \boldsymbol{\mu}_{\mathbf{b}_i}^T \boldsymbol{\Sigma}_e^{(k_c)} \boldsymbol{\mu}_{\mathbf{b}_j}, \tag{6}$$

$$\text{Diagonal elements } (i = j): \quad \text{Tr}\left(\boldsymbol{\Sigma}_{\mathbf{b}_i} \boldsymbol{\Sigma}_e^{(k_c)}\right) + \boldsymbol{\mu}_{\mathbf{b}_i}^T \boldsymbol{\Sigma}_e^{(k_c)} \boldsymbol{\mu}_{\mathbf{b}_j} + \mathbf{m}_e^{(k_c)T} \boldsymbol{\Sigma}_{\mathbf{b}_j} \mathbf{m}_e^{(k_c)}. \tag{7}$$

Nonlinear Activation Function: Convolutional layers are commonly followed by an element-wise nonlinear activation function, e.g., Rectified Linear Unit (ReLU). Let ψ denote the activation function and $\mathbf{g}_e^{(k_c)}$ denote the output of the activation function, i.e., $\mathbf{g}_e^{(k_c)} = \psi[\mathbf{z}_e^{(k_c)}]$ for $k_c = 1, \cdots, K_c$. We use the first-order Taylor series approximation to derive the mean and covariance of the random variable $\mathbf{g}_e^{(k_c)}$, i.e.,

$$\boldsymbol{\mu}_{\mathbf{g}_e^{(k_c)}} \approx \psi(\boldsymbol{\mu}_{\mathbf{z}_e^{(k_c)}}), \quad \boldsymbol{\Sigma}_{\mathbf{g}_e^{(k_c)}} \approx \boldsymbol{\Sigma}_{\mathbf{z}_e^{(k_c)}} \odot \left[\nabla\psi(\boldsymbol{\mu}_{\mathbf{z}_e^{(k_c)}}) \nabla\psi(\boldsymbol{\mu}_{\mathbf{z}_e^{(k_c)}})^T\right], \tag{8}$$

where ∇ is the gradient with respect to $\mathbf{z}_e^{(k_c)}$.

Max-Pooling Operation: The max-pooling operation is often used to downsample the incoming feature map. We propagate the mean through the max-pooling layer using the classical operation of selecting the largest value from a patch in the feature map. The pooling for the covariance is achieved by only retaining the rows and columns (of the incoming covariance matrix) corresponding to the retained elements (pooled elements) of the mean vector. We write the mean and covariance as follows:

$$\boldsymbol{\mu}_{\mathbf{p}_e^{(k_c)}} = \text{pool}(\boldsymbol{\mu}_{\mathbf{g}_e^{(k_c)}}), \quad \boldsymbol{\Sigma}_{\mathbf{p}_e^{(k_c)}} = \text{co-pool}(\boldsymbol{\Sigma}_{\mathbf{g}_e^{(k_c)}}). \tag{9}$$

An encoder may consist of multiple layers of convolution operations, nonlinear activation functions, and max-pooling to get a low-dimensional representation of the input.

3.4. Decoder Operations

The operations in the decoder path start with the low-dimensional representation produced by the encoder. The decoder may also include multiple convolutional layers, which are performed following the mathematical relationships provided in Eqs. (5)-(7).

Up-sampling: The up-sampling is an essential part of the decoder path that increases the resolution of the input. Using $\mathbf{g}_d^{(k_c)}$ to represent the input to the up-sampling operation and $\mathbf{u}_d^{(k_c)}$ as the output, we have:

$$\mathbf{u}_d^{(k_c)} = \text{up-sample}\left(\mathbf{g}_d^{(k_c)}\right). \tag{10}$$

The mean of $\mathbf{u}_d^{(k_c)}$ is computed by inserting zeros between two consecutive elements of the input and padding with zeros. The covariance matrix is obtained by adding rows and columns of zeros at locations corresponding to the newly added zeros in the mean.

Up-convolution: The up-sampling operation may produce sparse feature maps with many zeros. Generally, a 2×2 convolution operation is performed to get a dense high-resolution output. The mean and covariance are computed using results presented in Eqs. (5)-(7).

Padding: The padding operation applied to the mean is the same as the classical zero-padding operation. For the covariance matrix, we add a new row and a new column for each element padded to the mean. The new elements added in the covariance matrix are all set to zero, and the variance (diagonal) elements are set to a user-defined small value with $\sigma_{pa} > 0$.

Concatenation: The features from the encoder side are generally concatenated with the corresponding features from the decoder to improve the localization of various objects in the input. The feature maps from the encoder path may need to be resized or cropped before they can be concatenated with the decoder features due to the differences in size.

Let $\mathcal{G}_e^c$ be the c^{th} encoder feature map, and $\mathbf{g}_e^{(k_c)}$ the k_c^{th} slice from such map with mean and covariance $\boldsymbol{\mu}_{\mathbf{g}_e^{(k_c)}}$ and $\boldsymbol{\Sigma}_{\mathbf{g}_e^{(k_c)}}$, respectively. The cropped feature map is denoted with $\mathcal{G}_e^{*c}$ where k_c^{th} slice is $\mathbf{g}_e^{*(k_c)}$. For $k_c = 1, \ldots, K_c$, $\boldsymbol{\mu}_{\mathbf{g}_e^{*(k_c)}} = \text{crop}(\boldsymbol{\mu}_{\mathbf{g}_e^{(k_c)}})$ while $\boldsymbol{\Sigma}_{\mathbf{g}_e^{*(k_c)}}$ is obtained by removing the rows and columns from $\boldsymbol{\Sigma}_{\mathbf{g}_e^{(k_c)}}$ corresponding to the cropped elements of $\boldsymbol{\mu}_{\mathbf{g}_e^{(k_c)}}$.

The output of the concatenation operation is a feature map $\mathcal{G}_d^{*c} = \{\mathcal{G}_d^c, \mathcal{G}_e^{*c}\}$, where $\mathcal{G}_d^c$ is the c^{th} decoder feature map. The concatenation operation is done along the dimension that represents channels in the feature maps (generally the third dimension).

Softmax Function: Pixel-level segmentation can be considered as a dense classifica-

tion problem where we assign a label to each pixel. Hence, for a multi-class problem, a softmax function ϕ is applied to the output of the last layer. Let $\mathbf{F}$ represent the output of the last layer with mean $\boldsymbol{\mu}_{\mathbf{F}}$ and covariance $\boldsymbol{\Sigma}_{\mathbf{F}}$, and $\mathbf{Y}$ denote the output of the network after the softmax operation. We can approximate the mean $\boldsymbol{\mu}_{\mathbf{Y}}$ and covariance $\boldsymbol{\Sigma}_{\mathbf{Y}}$ using first-order Taylor series, that is:

$$\boldsymbol{\mu}_{\mathbf{Y}} \approx \phi(\boldsymbol{\mu}_{\mathbf{F}}), \quad \boldsymbol{\Sigma}_{\mathbf{Y}} \approx \boldsymbol{J}_{\phi} \boldsymbol{\Sigma}_{\mathbf{F}} \boldsymbol{J}_{\phi}^{T}, \tag{11}$$

where $\boldsymbol{J}_{\phi}$ is the Jacobian matrix of ϕ computed with respect to $\mathbf{F}$ evaluated at $\boldsymbol{\mu}_{\mathbf{F}}$.

The mathematical results presented above for various operations can be used to build any type of deep NN in addition to the proposed encoder-decoder-based networks.

4. Experimental Methods

4.1. Datasets

We use three different medical benchmark segmentation datasets, including lung CT [34], hippocampus MRIs [35] and brain tumor MRIs [36], and one clinical dataset [37]. Our experiments use only the publicly available annotated data from the respective datasets, i.e., unlabeled data is not used for training or validation. The datasets are divided into training, validation and testing bins with approximately 80% selected for training, 10% for validation and 10% for testing.

4.1.1. Lungs Dataset

The dataset includes 20 CT scans from the chest region, available at zenodo.org [34]. This heterogeneous dataset consists of both COVID-19 and non-COVID-19 patients. The data annotations include left lung, right lung and infections (if found). We consider a binary segmentation task for this dataset, i.e., delineating the boundaries of the lungs in the given CT images. We assign a label of 0 to the background and 1 to lung tissue. The pre-processing steps include: 1) windowing the Hounsfield units range between -1250 and 250; 2) normalizing all pixel values between 0 and 1; 3) deleting empty slices, i.e., slices that include only the label 0 corresponding to the background to minimize class imbalance; and 4) cropping all images to a single size, i.e., 512×512 pixels.

4.1.2. Hippocampus Dataset

The Hippocampus data is available as part of the Medical Segmentation Decathlon [35]. The dataset consists of 394 single-modality MRI scans. The segmentation task requires the precise delineation of two adjacent structures, i.e., anterior (label 1) and posterior (label 2). The pre-processing steps include: 1) normalizing the data to reduce the image bias (which is a characteristic of MRI data); 2) deleting empty slices, i.e., those that include only the background; and 3) padding images to have the same input size of 64×64 pixels.

4.1.3. Brain Tumor Segmentation (BraTS) Dataset

The Brain Tumor Segmentation (BraTS) dataset is available as part of the MICCAI BraTS Challenge. The dataset includes about 300 multi-modal (T1, T1c, T2, and FLAIR) MRI scans from 274 brain tumor patients (some patients have multiple MRI scans) [36]. The dataset is divided into two main types of tumors: low-grade gliomas (LGG) and high-grade gliomas (HGG). We focus on the more challenging HGG dataset in our experiments. The pre-processing steps include: 1) normalizing data to reduce the image bias; 2) deleting images that do not include any tumor structure; and 3) cropping each image to the size of 240×240 pixels. The input data size for each sample in the dataset is $240 \times 240 \times 4$ pixels, where the last number represents the four modalities, i.e., T1, T1c, T2, and FLAIR. All four networks (U-Net, Bayes U-Net, and SUPER U-Net) are trained to segment 5 different labels in the HGG MRIs, i.e., normal tissue (label 0), necrosis (label 1), edema (label 2), non-enhancing tumor (label 3), and enhancing tumor (label 4). In most clinical applications, generally, three tumor regions are considered for evaluating the results of segmentation: whole tumor (labels 1, 2, 3 and 4), tumor core (labels 1, 3 and 4), and enhancing tumor region (label 4) [36].

4.1.4. Clinical Dataset

We acquired a real-world, anonymized, IRB-approved brain tumor dataset from the O'Neal Comprehensive Cancer Center at the University of Alabama at Birmingham (UAB) School of Medicine. This dataset will be made available upon request. The imaging dataset includes 627 fluid-attenuated inversion recovery (FLAIR) sequences,

Table 1: Architecture Details for Different Datasets

Dataset	Encoder Blocks	Decoder Blocks	Encoder Filter	Decoder Kernels
Lungs	3	2	16, 32, 64	32, 16
Hippocampus	3	2	32, 64, 128	64, 32
BraTS	5	4	64, 128, 256, 512, 1024	512, 256, 128, 64
Clinical	5	4	16, 32, 64, 128, 256	128, 64, 32, 16

including 24 images each on average, from patients diagnosed with World Health Organization grade 2 gliomas, seen at the neuro-oncology clinics at the University of Alabama at Birmingham [37]. The tumor masks were manually annotated by an expert physician. The pre-processing steps include: 1) normalizing data to reduce the image bias; 2) deleting images that do not include any tumor structure; and 3) cropping each image to the size of 240×240 pixels.

4.2. Segmentation Network Architectures

We apply the proposed SUPER-Net framework to the U-Net architecture; for simplicity, we refer to it as SUPER U-Net. We compare SUPER U-Net with three state-of-the-art segmentation networks, a deterministic U-Net [9], a Bayes U-Net obtained using MC dropout [20], and an Ensemble U-Net [38].

4.2.1. U-Net - The Baseline Segmentation Architecture

Among all architectures proposed for medical image segmentation, U-Net is the most widely used [9]. U-Net is built using the encoder-decoder structure with a contracting path that is almost identical to the expanding path. The contracting path may consist of multiple encoder blocks, which, in turn, may include various convolution layers, max-pooling, and nonlinear activations. The expanding path consists of multiple decoder blocks, which are made of multiple layers of convolution, activation functions, up-convolution, up-sampling and padding. Additionally, there are connections between the encoder and decoder blocks that concatenate feature maps from the encoder with the corresponding feature maps of the decoder. Finally, a 1×1 convolution and SoftMax are applied to the decoded feature maps before calculating the cross-entropy loss function.

In the original U-Net architecture [9], the border pixels are lost due to un-padded convolution operations and the missing regions are extrapolated by mirroring. Such processing may yield erroneous results for some medical image segmentation datasets. Hence, in our setting, we apply the padding operation to increase the size of the feature maps and reconstruct the full image at the output of the network. We include the padding operation twice in each decoder block on the expanding path. The first padding operation is performed before the concatenation, and the second is performed before the second convolution in each decoder block. In our experiments, we refer to this U-Net architecture as the deterministic segmentation network.

In Table 1, we report the specifics of the architectures for all datasets. The kernel size is set to 3 for all datasets. For clinical data, we use convolutions with padding set to *same*, and apply batch normalization on both the encoder and decoder.

4.2.2. Bayes U-Net

Bayes U-Net is built using the MC dropout technique following the implementation of [20]. The dropout is used only in the central blocks with the probability of dropping a neuron set to $p = 0.5$. Bayes U-Net uses cross-entropy loss function. At the inference time, we use $N = 20$ MC samples and the uncertainty is measured in terms of predictive entropy (PE) [18].

4.2.3. Ensemble U-Net

Ensemble U-Net is built using an ensemble of U-Net models. We trained 5 networks with different initializations and used the entire training set for each model. The number of networks is chosen following the results in [24]. Ensemble U-Net uses the cross-entropy loss function. At inference time, each input is fed to the five models and the outputs are used to estimate uncertainty using PE.

4.2.4. SUPER U-Net

SUPER U-Net uses the mathematical operations presented in Sections 3.3 and 3.4 to propagate the first two moments of the variational distribution through the U-Net architecture. The output of SUPER U-Net consists of a segmentation map and an uncertainty map. The former is given by the mean of the predictive distribution, while the

Table 2: Training Hyperparameters for Different Datasets

Dataset	**Optimizer**	**Learning Rate**	**Batch Size**	**Epochs**	σ_{pa} (SUPER U-Net)
Lungs	Adam	0.001	10	50	0.1
Hippocampus	Adam	0.001	20	100	0.02
BraTS	Adam	0.001	20	100	0.1
Clinical	Adam	0.001	10	100	0.01

latter is generated by the covariance of the predictive distribution. We use a Gaussian variational distribution and employ the ELBO loss function defined in Eq. (3). We optimize the ELBO loss function with respect to the variational parameters, i.e., the mean and covariance of the variational distribution. To reduce the computational complexity, we propagate diagonal covariance matrices.

4.3. Other Experimental Settings

We report the specific hyperparameters used for each dataset in Table 2, including the optimizer, learning rate, batch size, number of training epochs, and σ_{pa} for the padding in the SUPER U-Net model. The selection was determined through empirical evaluation. We explored different values for each hyperparameter and selected those that provided stable training, faster convergence, and improved segmentation performance across all datasets. The values of σ_{pa} in SUPER U-Net were tuned to balance the trade-off between predictive uncertainty and segmentation accuracy. The batch size was determined based on the size of the data and the available hardware constraints. All simulations were performed using Python with the TensorFlow library on an NVIDIA RTX A6000 GPU.

We report the Dice Similarity Coefficient (DSC) as the metric to compare the performance of all four networks. We conduct a detailed robustness analysis of the performance of all four networks using two types of noise, i.e., Gaussian and adversarial. We compare the performance of all four networks under various levels of Gaussian noise added to the test data of all three datasets. We measure the noise level using the signal-to-noise ratio (SNR) in the units of decibels (dB). For the adversarial noise, we use the Fast Gradient Sign Method (FSGM) to generate untargeted attacks [39], and we use

Table 3: DSC for Lungs Dataset - performance comparison under additive Gaussian noise.

	U-Net	**Bayes U-Net**	**Ensemble U-Net**	**SUPER U-Net**
Noise Free	**.83**	**.83**	**.83**	**.83**
Gaussian noise added to the entire image				
SNR ≈ 35 dB	.82	.82	.82	**.83**
SNR ≈ 3 dB	.16	.19	.11	**.21**
Gaussian noise added to lung pixels only				
SNR ≈ 31 dB	.82	**.83**	**.83**	**.83**
SNR ≈ 14 dB	.63	.63	.65	**.79**

the Projected Gradient Descent (PGD) method to generate targeted adversarial attacks [40]. The attacks are generated with a maximum number of iterations set to 20 and a step size of 1. We select a *source* class and a *target* class to generate targeted attacks. The adversarial attack algorithm will try to fool the trained network into predicting pixels belonging to the *source* class as the pixels of the *target* class.

5. Results and Discussion

We report our results in four parts. First, we present the performance analysis (measured using DSC) of the four networks (U-Net, Bayes U-Net, Ensemble U-Net, and SUPER U-Net) under various levels of Gaussian noise added to the benchmark test datasets. Next, we analyze the same four networks under various levels of targeted and untargeted adversarial attacks. We report the results of the clinical data. Finally, we present an analysis of the uncertainty maps and the predictive variance generated by the proposed SUPER U-Net at inference time. For reference, we report DSC values for U-Net, Bayes U-Net, Ensemble U-Net and SUPER U-Net for noise-free test data in tables 3, 4, and 5.

5.1. Evaluation Under Gaussian Noise

Table 3, and Figs. 2 and 3 show DSC values for U-Net, Bayes U-Net, Ensemble U-Net and SUPER U-Net under different levels of Gaussian noise. For each dataset, we report results for two cases, i.e., noise added to the entire input image or only to the structures that the networks are trying to segment, e.g., tumors in the BraTS dataset.

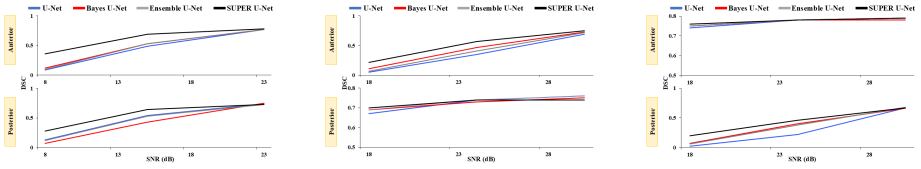
Table 4: DSC for Hippocampus Dataset - Noise Free.

	Anterior				**Posterior**			
	U-Net	**Bayes U-Net**	**Ensemble U-Net**	**SUPER U-Net**	**U-Net**	**Bayes U-Net**	**Ensemble U-Net**	**SUPER U-Net**
Noise Free	**.79**	**.79**	**.79**	**.79**	.76	.76	**.77**	.74

Table 5: DSC for BraTS Dataset - Noise Free

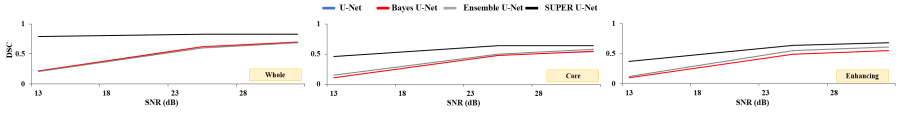
	Whole				***Core***				***Enhancing***			
	U-Net	**Bayes U-Net**	**Ensemble U-Net**	**SUPER U-net**	**U-Net**	**Bayes U-Net**	**Ensemble U-Net**	**SUPER U-net**	**U-Net**	**Bayes U-Net**	**Ensemble U-Net**	**SUPER U-net**
Noise Free	.77	.77	.76	**.83**	.58	.58	.60	**.64**	.57	.57	.63	**.69**

In Table 3, we compare the performance of the four models for the noise-free test data and for two levels of Gaussian noise added to the entire image and the lung pixels only. Fig. 2, reports the performance of the four models when Gaussian noise is applied to the Hippocampus test data. We consider 3 scenarios: noise added to the entire image, the Anterior pixels only, and the Posterior pixels only. We show the results for the BraTS test data in Fig. 3. We plot DSCs vs. SNR for the three tumor regions. Each subplot compares the performance of the four networks for multiple levels of Gaussian noise added to the tumor pixels only (Fig. 3a) and the entire image (Fig. 3b). The proposed SUPER U-Net generally demonstrates more robust behavior as compared to other models especially at low SNR values, i.e., high levels of noise.

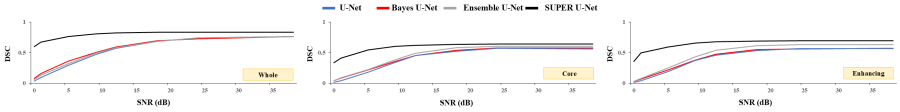


(a) Noise added to entire image (b) Noise added to Anterior pixels (c) Noise added to Posterior pixels

Figure 2: Performance of the four networks, i.e., U-Net (blue), Bayes U-Net (red), Ensemble U-Net (gray), and SUPER U-Net (black), under various levels of Gaussian noise added to the (a) entire image, (b) Anterior pixels only, and (c) Posterior pixels only of the Hippocampus test data. We plot Dice Similarity Coefficient (DSC) versus Signal to Noise Ratios (SNRs) for the Anterior and Posterior hippocampus.



(a) Gaussian noise added to the tumor pixels only



(b) Gaussian noise added to all pixels

Figure 3: Performance of the four networks, i.e., U-Net (blue), Bayes U-Net (red), Ensemble U-Net (gray) and SUPER U-Net (black), under various levels of Gaussian noise added to (a) the tumor pixels only and (b) all pixels of the BraTS test data. The three sub-plots show the Dice Similarity Coefficient (DSC) values for a range of Signal to Noise Ratios (SNRs) for three different tumor regions: whole tumor, core, and enhancing.

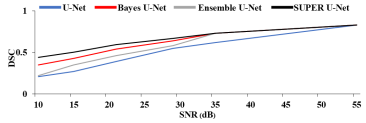
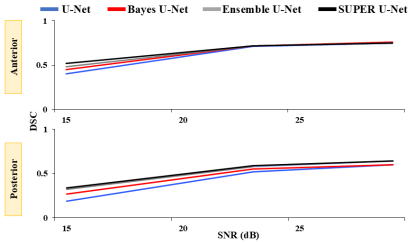


Figure 4: Performance of four networks, i.e., U-Net (blue), Bayes U-Net (red), Ensemble U-Net (gray) and SUPER U-Net (black), under various levels of untargeted attacks to the Lungs test data. We display Dice Similarity Coefficient (DSC) values for a range of Signal to Noise Ratio (SNR).

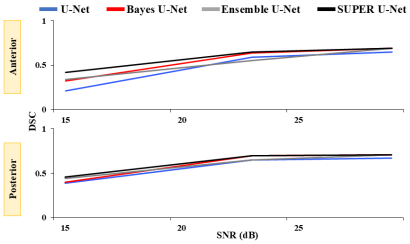
5.2. Evaluation Under Adversarial Attacks

We assess the robustness of all four networks against targeted and untargeted adversarial attacks. We show the results in Figures 4, 5, and 6. We plot the DSC vs. SNR for the four approaches.

In Fig. 4, we show DSC values for a range of untargeted adversarial attacks generated using FGSM against the lung test dataset. In Fig. 5, we consider various levels of targeted attacks applied to (a) the Anterior pixels only and (b) the Posterior pixels only of the Hippocampus test data. For both attack types, we report the performance for the two structures of interest, i.e., anterior and posterior hippocampus. On the other hand, Fig. 6 presents both targeted and untargeted adversarial attacks applied to the BraTS test data. The three subplots compare the performance of the four networks on the three structures of interest: whole tumor, core and enhancing tumor. We observe that SUPER U-Net shows better performance (i.e., high DSC values) as compared to



(a) Targeted attacks - source: label 1, target: label 2



(b) Targeted attacks - source: label 2, target: label 1

Figure 5: Performance of the four networks, i.e., U-Net (blue), Bayes U-Net (red), Ensemble U-Net (gray), and SUPER U-Net (black), under various levels of adversarial attacks applied to the Hippocampus test data. We show targeted adversarial attacks with (a) source: label 1, target: label 2, (b) viceversa. The two subplots show the Dice Similarity Coefficient (DSC) values for the Anterior and Posterior hippocampus measured using Signal to Noise Ratios (SNRs).

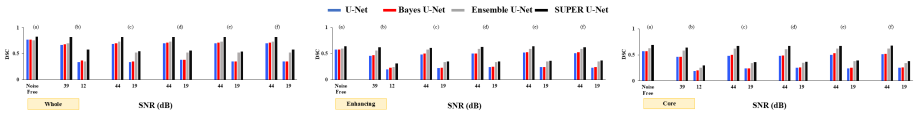


Figure 6: Performance of the four networks, i.e., U-Net (blue), Bayes U-Net (red), Ensemble U-Net (gray) and SUPER U-Net (black) under various levels of adversarial attacks applied to the BraTS test data. The three sub-plots show the Dice Similarity Coefficient (DSC) values for a range of Signal-to-Noise Ratios (SNRs) for three different tumor regions: whole, core, and enhancing tumor. We show (a) noise-free case, (b) untargeted attacks generated using FGSM, (c) Targeted adversarial attacks with source: label 3, target: label 1, (d) Targeted adversarial attacks with source: label 1, target: label 3, (e) Targeted adversarial attacks with source: label 3, target: label 2, and (f) Targeted adversarial attacks with source: label 2, target: label 3.

the other three networks, especially for stronger attacks (i.e., low values of SNR).

5.3. Evaluation of the Clinical Data

We show that SUPER U-Net can scale to real-world datasets. SUPER U-Net is able to achieve 86% DSC on held-on test data. Figure 9 shows sample scans from the UAB clinical data (first column) along with ground-truth segmentation (second column) and SUPER U-Net's segmentation and associated uncertainty maps (third and fourth columns, respectively). The representative images show that SUPER U-Net is uncertain when a tumor region is missed (scans 1 and 2), as well as for unusually low signal pixels within the tumor (scan 1). A typical tumor is associated with a high

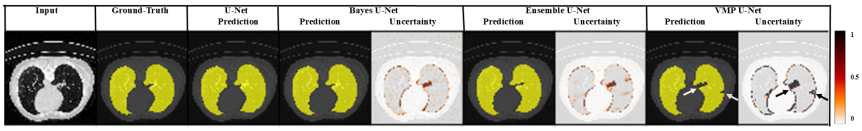


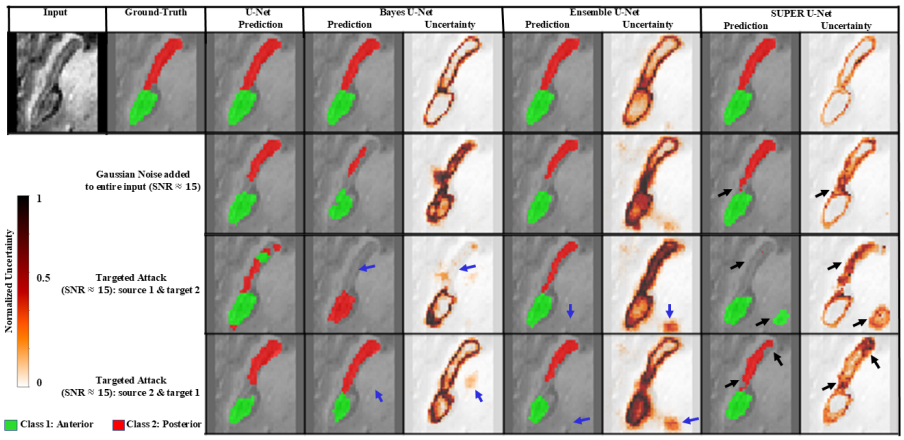
Figure 7: Segmentation of the Lungs test data. We show (left to right) the CT input image, the ground-truth segmentation, noise-free models' segmentation predictions (U-Net, Bayes U-Net, Ensemble U-Net, and SUPER U-Net). The uncertainty map of each Bayesian model is shown next to the corresponding prediction. The white arrows point to regions incorrectly classified by the network. We note that the corresponding pixels in the uncertainty maps reflect the low confidence by responding with higher variance values.

FLAIR signal; in the first scan, the central part of the tumor is associated with a low signal, which is atypical (see right arrow in Fig. 6, row 1, column 4). In a sense, the model attracts the physician's attention to these regions in the image so that they can confirm whether these are part of the tumor or not. In the second scan, the tumor is totally missed, but the model exhibits high uncertainty in the missed region. The last scan has no tumor, and SUPER-Net correctly predicts true negative cases and associates a very low uncertainty (predictive variance ≈ 0) or equivalently a high confidence in these predictions.

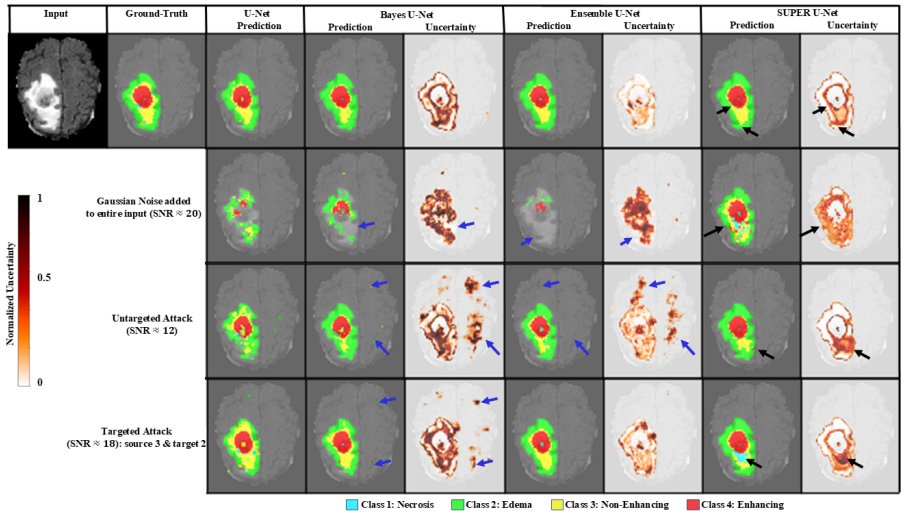
5.4. Uncertainty Maps and Predictive Variance – Quantitative Analysis

5.4.1. Uncertainty Maps

The output of SUPER U-Net consists of the pair: segmentation map (prediction) and uncertainty map (obtained from the predictive covariance). For the other approaches, uncertainty is evaluated through multiple forward passes. In Fig.7, we present a representative case for the Lungs dataset. In Fig. 8 we present representative cases selected from the hippocampus (8a) and BraTS (8b) test data. We show the input modality (only FLAIR for the BraTS data), the ground-truth label, and predictions with associated uncertainty maps. The first row presents the noise-free case, the second row reports the predictions and uncertainty maps for the Gaussian noise case, and the third and fourth rows show two examples of adversarial attacks. We normalized the predictive variance of SUPER U-Net for better visual comparison to the uncertainty maps of Bayes U-Net and Ensemble U-Net. We point to regions (pixels) incorrectly classified



(a) Hippocampus test data



(b) BraTS test data

Figure 8: Segmentation of the (a) hippocampus and (b) BraTS test data. The first row shows (left to right) the input image, the ground-truth segmentation, and noise-free models' segmentation predictions (U-Net, Bayes U-Net, Ensemble U-Net, and SUPER U-Net). The uncertainty map of each Bayesian model is shown next to the corresponding prediction. Rows $2, 3, 4$ display the segmented predictions along with their uncertainty maps (when applicable) for additive Gaussian noise and two adversarial attacks, respectively. The black arrows point to regions incorrectly classified by the network. Observe that the corresponding pixels in the uncertainty maps reflect low confidence or higher variance values. The blue arrows refer to inconsistent uncertainty estimates: low confidence is associated with incorrect predictions or high uncertainty for correctly classified regions.

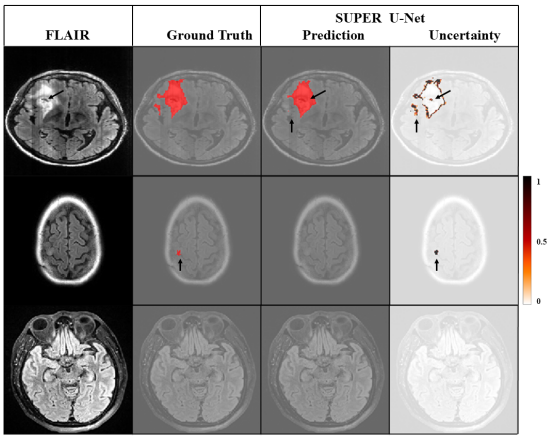


Figure 9: Sample scans from the clinical data. Images show (left to right) the Flair input image, the ground-truth segmentation, SUPER U-Net prediction and uncertainty map overlaid on the input scan. The black arrows point to regions incorrectly classified by the network or unusually low (atypical) signals in the FLAIR.

by our network with the black arrows, and we point to the corresponding locations in the uncertainty maps. It is evident from the figure that SUPER U-Net associates high uncertainty with incorrect predictions and pixels belonging to targeted regions.

5.4.2. Predictive Variance – Quantitative Analysis

We investigate the response of the derived second moment (variance/uncertainty) and relate it to the model's performance (DSC). We calculate the average predictive variance from uncertainty maps and plot these values against various levels of Gaussian noise in Fig. 10, and adversarial attacks in Fig. 11, for hippocampus and BraTS datasets. It is more instructive and insightful if sub-plots in both figures are interpreted from right to left, i.e., decreasing SNR or equivalently increasing noise in the test data. We note that the predictive variance increases monotonically with increasing noise (i.e., decreasing SNR) for all three sub-figures in Fig. 10 and all four sub-figures in Fig. 11. This behavior, i.e., increasing uncertainty with increasing noise, demonstrates that the network is aware of higher noise in the input. A useful and meaningful uncertainty estimate should convey a lower confidence/ higher uncertainty for low-accuracy segmented images [41]. Table 7 reports SUPER U-Net average predictive variance for

Table 6: DSC for test sample from BraTS Data - performance comparison before and after removal of uncertain pixels.

	Whole			***Core***			***Enhancing***		
	Bayes U-Net	**Ensemble U-Net**	**SUPER U-net**	**Bayes U-Net**	**Ensemble U-Net**	**SUPER U-net**	**Bayes U-Net**	**Ensemble U-Net**	**SUPER U-net**
Noise Free									
Original	.96	.96	.97	.85	.87	.89	.97	.96	.98
Uncertain Pixels Removed	.99 ↑	.99 ↑	1 ↑	.91 ↑	.88 ↑	.90 ↑	1 ↑	.98 ↑	1 ↑
Gaussian Noise added to entire input (SNR ≈ 20)									
Original	.56	.14	.97	.40	.16	.86	.58	.3	.86
Uncertain Pixels Removed	.11↓	0 ↓	.98 ↑	.40	.10↓	.99 ↑	.61 ↑	.21 ↓	1 ↑
Untargeted adversarial attacks (SNR ≈ 12)									
Original	.90	.91	.94	.74	.81	.78	.89	.91	.95
Uncertain Pixels Removed	.88 ↓	.79 ↓	.95 ↑	.75 ↑	.89 ↑	.81 ↑	1 ↑	.90 ↓	1 ↑
Targeted adversarial attacks (SNR ≈ 18): source 3, target 2									
Original	.93	.98	.99	.89	.90	.89	.95	.96	.97
Uncertain Pixels Removed	.90 ↓	.98	.99	.96 ↑	.96 ↑	.95 ↑	1 ↑	.98 ↑	1 ↑

Table 7: SUPER U-Net Predictive Variance for BraTS Dataset.

	Whole	**Core**	**Enhancing**
Correct	.007	.008	.008
Incorrect	.289	.307	.336

correctly classified and misclassified pixels on noise-free BraTS test set. Observe that the incorrect pixels are associated to high variance or less confident predictions.

Following the quantitative uncertainty evaluation task in the BRATS challenge [41], we compute the percentage change in DSCs when *uncertain* pixels are removed, and DSC is computed only using the remaining pixels. To define *uncertain* pixels, for each model, we set as a threshold the average predictive uncertainty for correctly classified pixels for the noise-free case. All pixels with an uncertainty value above this threshold are marked as uncertain and removed from the computation of the DSC. We report the change in DSCs in table 6. The sample scan corresponds to that provided qualitatively in Fig. 8b. Our approach consistently produces higher (↑) DSCs after removing uncertain pixels, i.e., unlike other approaches, our predictive variance (uncertainty) is above the threshold only for incorrectly classified pixels. Such information is valuable for detecting when the network may fail and its predictions may become untrustworthy.

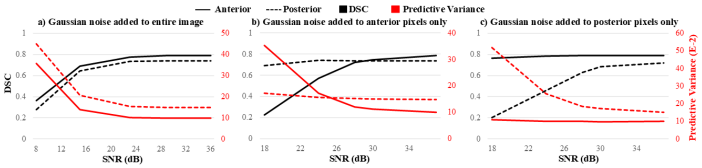


Figure 10: Accuracy, measured by Dice Similarity Coefficient (DSC) and plotted in black, and average predictive variance of SUPER U-Net, plotted in red, under various levels of Gaussian noise added in the test data for hippocampus dataset. SNR denotes the signal-to-noise ratio. (a) Noise is added to the entire input. (b) Noise is added to the anterior pixels only. (c) Noise is added to the posterior pixels only.

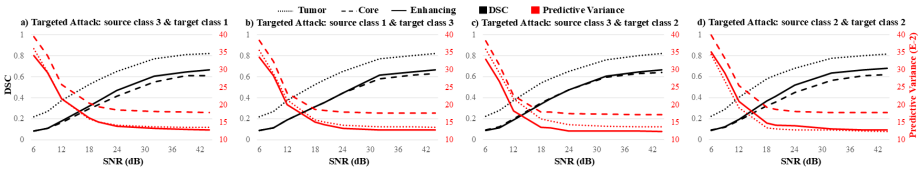


Figure 11: Accuracy, measured by Dice Similarity Coefficient (DSCs) and plotted in black, and average predictive variance of SUPER U-Net, plotted in red, under various levels of adversarial attacks applied to the test data for BraTS dataset. SNR denotes the signal-to-noise ratio. Test data is corrupted with targeted attacks: (a) source class 3 and target class 1, (b) source class 1 and target class 3, (c) source class 3 and target class 2, (d) source class 2 and target class 3.

5.5. Discussion

Our extensive analysis shows that SUPER U-Net has superior robustness to noise and adversarial attacks compared to state-of-the-art uncertainty quantification approaches. In the noise-free case, SUPER U-Net performance is equivalent to the state-of-the-art models. However, as the noise level increases (Gaussian or adversarial), the task or the dataset becomes more complicated, e.g., BraTS data (multiple segmentation labels and multiple modalities), SUPER U-Net outperforms both the deterministic U-Net and other Bayesian approaches, i.e., Bayes U-Net and Ensemble U-Net. The superior performance and robustness to noise, especially at high levels of noise and complex tasks/data, can be attributed to the intrinsic learning of the uncertainty during training through propagation of the covariance information (in addition to the mean). This is in contrast to post-hoc estimation of uncertainty using MC runs through the network or

models averaging at the inference time.

5.5.1. The Significance of Uncertainty Information

The reliability of the segmentation predictions can be assessed using the uncertainty maps. This is evident from the comparison with the point-estimate approach, i.e., U-Net (figures 8a, 8b). SUPER U-Net associates higher uncertainty with incorrect predictions and pixels or regions targeted by adversarial attacks (marked with black arrows). When the predicted segmentation is (almost) identical to the ground truth, the model is confident in its segmentation predictions and exhibits uncertainty only at the boundary between the structure of interest and the background. When the network's segmentation predictions are incorrect, or the input is perturbed by noise or adversarially attacked, SUPER U-Net associates higher uncertainty values with the predictions (rows 2 to 4). On the other hand, inspecting the maps generated using other probabilistic approaches, we see that their uncertainty maps do not convey the same level of insight into the trustworthiness of the predictions. Ideally, the model's uncertainty should increase (darker red shades) only for regions of incorrect classification and/or for regions with noise or artifacts. However, as shown with the blue arrows in Figs. 8a and 8b, both Bayes U-Net and Ensemble U-Net generate inconsistent uncertainty estimates: they associate low uncertainty (high confidence) to incorrect predictions and/or high uncertainty (low confidence) to correctly classified regions.

5.5.2. Clinical Impact

Coordinates of the tumor volume are used to determine the clinical target volume of the radiation therapy treatment. Inaccuracy and variation in defining critical volumes will affect everything downstream: treatment planning, dose-volume histogram analysis, and contour-based visual guidance used in image-guided radiation therapy. Studies have shown that under-coverage of radiation dose to the tumor target could compromise treatment outcomes [42]. Ultimately, both researchers and practitioners agree that radiotherapy is only as good as the accuracy with which the target is.

DL algorithms for segmentation have the potential to address the number one impediment to reliable use of imaging for guiding treatment planning of cancer, i.e., ac-

curate and objective delineation of tumors from healthy tissue and organs. However, reliability and lack of consistency of DL predictions hinder the safe deployment of such models in the clinic [43]. For example, a medical diagnostic system for detecting brain tumors from magnetic resonance scans may encounter a new tumor shape/structure (due to a different scan procedure) or an adversarial attack designed to fake a tumor for benefiting from medical bills [4]. Uncertainty information can limit the harmful consequences in such scenarios. Model confidence or uncertainty is critical when integrating the models in systems that make decisions that affect human life, either directly or indirectly. Ideally, the model should recognize the perturbed data and return an output (segmented image plus uncertainty map) that also conveys a high level of uncertainty.

SUPER U-Net simultaneously delivers the segmented image (prediction) along with the corresponding uncertainty map, which reflects the network's own confidence in the prediction of every pixel. The uncertainty information generated by the framework can provide critical guidance, particularly in cases where segmentation predictions are ambiguous. This capability will ensure that ML models are not merely passive tools but active collaborators in clinical workflows. For example, in figure 9, the SUPER U-Net prompts the physicians to false negative, i.e., missed tumor region and an unusual low SNR region inside the tumor structure. The uncertainty map can prompt the physician to pay particular attention to regions of low confidence rather than reviewing the model's prediction as a whole.

The uncertainty proposed with this work will help users develop trust in ML models as well as drive informed human-AI interaction. For example, ML systems that do not involve a physician-in-the-loop can flag the scan and request human intervention when uncertainty is above a set threshold. This research has the potential to improve the accuracy of tumor monitoring, optimize radiation therapy planning, and inspire the adoption of trustworthy ML.

5.5.3. Computational Complexity, Trade-offs, and Limitations

We report the average inference time of all four networks in Table 8. SUPER U-Net requires almost twice the time to process a single image at inference compared to a deterministic U-Net. This increase is due to the propagation of the covariance informa-

Table 8: Inference time per image

	U-Net	**Bayes U-Net**	**Ensemble U-Net**	**SUPER U-Nets**
Time (min)	0.81	0.82 N_1^*	0.81 N_2^*	1.92

*N_1 and N_2 denote the number of runs at inference time and ensemble networks, respectively.

tion, which involves additional operations. Other approaches that deliver uncertainty, i.e., Bayes U-Net and Ensemble U-Net, take the same time as that of a deterministic U-Net for one pass (or one model). However, these approaches necessitate multiple passes to calculate the variance of the prediction. For example, we used $N = 20$ for Bayes U-Net, leading to 16.4 ms for each image, almost 8 times more than SUPER U-Net. Additionally, it is worth mentioning that ensemble approaches require an extended training time and additional storage as several models are trained and saved.

The computational complexity of the proposed SUPER U-Net framework is influenced by the propagation of the first two moments of the variational pdf. Given the tensor normal distribution, SUPER U-Net requires an additional trainable parameter per convolutional kernel [6]. Hence, the total number of trainable parameters remains nearly the same as that of a deterministic model. For instance, consider the Lungs dataset with the number of kernels as reported in Table 1, and size 3×3. For a deterministic model, we have $n_1 = 1440$ total parameters, corresponding to a storage requirement of approximately 5.625 KB. In the case of the proposed method, we have one additional parameter per kernel, resulting in a total of $n_2 = 1600$ parameters, corresponding to approximately 6.25 KB of storage. This slight increase in storage highlights that the main computational burden arises from performing separate operations on the mean and variance vectors rather than from storing the additional parameters.

While these drawbacks present challenges for real-time or resource-constrained scenarios, they are counterbalanced by several significant advantages. First, SUPER U-Net demonstrates superior robustness to noise and adversarial attacks. Additionally, the model's uncertainty provides a valuable tool to assess the reliability of the predictions. Finally, prior work showed the ability of Bayesian models to discover redundant kernels that can be pruned without affecting accuracy, hence reducing the storage re-

quirements [44].

Certain limitations must be considered when evaluating the applicability of SUPER U-Net. One key challenge concerns its performance on small and imbalanced datasets. While Bayesian methods, including early works such as [33] and [20], have demonstrated improved generalization in low-data regimes, dataset imbalance can still influence uncertainty estimates. In cases where specific structures appear infrequently in training, the reliability of uncertainty predictions remains unexplored in this study.

Another limitation relates to the sensitivity of SUPER U-Net to prior assumptions and hyperparameter choices. The model adopts a Gaussian prior over network weights, a common assumption in Bayesian models. However, this prior may not always be optimal for complex medical imaging tasks, where the underlying data distribution exhibits non-Gaussian properties. While the first-order Taylor expansion provides an efficient approximation, propagating additional moments could enhance robustness. Moreover, hyperparameters such as the prior variance and the KL regularization term significantly influence model behavior, and suboptimal tuning could lead to failure in learning, overconfident predictions, or excessive uncertainty. Future work could explore these directions to assess SUPER U-Net's behavior in rare disease segmentation and other data-scarce applications while exploring alternative priors.

6. Conclusion

This study introduced SUPER-Net, a novel Bayesian DL framework that effectively quantifies uncertainty in medical image segmentation tasks using encoder-decoder architectures. One of the strengths of SUPER-Net is its ability to produce pixel-wise uncertainty maps alongside segmentation outputs in real-time without relying on expensive post-hoc sampling techniques like Monte Carlo. This inherent capability of uncertainty quantification enhances the trustworthiness and reliability of the model's predictions and makes it more robust in the face of noisy and adversarial inputs, as demonstrated across multiple medical imaging datasets. SUPER-Net's ability to propagate uncertainty through nonlinear layers via a Taylor series approximation is a significant step forward in making DL models more interpretable and suitable for real-world

clinical applications. Researchers and practitioners in medical imaging can benefit from SUPER-Net by utilizing its uncertainty maps to improve the reliability of automated segmentation, especially in high-risk clinical settings. This framework can be adapted for different architectures and imaging modalities, thus serving as a valuable tool for applications requiring high levels of confidence in predictions. For future work, we plan to address the current limitations by exploring more complex posterior distributions beyond the Gaussian assumption to better capture uncertainties in heterogeneous data. Additionally, integrating SUPER-Net with active learning strategies could help refine segmentation models by focusing on areas of high uncertainty, thus improving training efficiency. Furthermore, the potential of SUPER-Net in other domains, such as object detection and image registration, will be investigated to expand its applicability beyond segmentation tasks.

References

[1] X. Liu, L. Faes, A. U. Kale, S. K. Wagner, D. J. Fu, A. Bruynseels, T. Mahendiran, G. Moraes, M. Shamdas, C. Kern, et al., A comparison of deep learning performance against health-care professionals in detecting diseases from medical imaging: a systematic review and meta-analysis, The lancet digital health 1 (6) (2019) e271–e297.

[2] B. Biggio, F. Roli, Wild patterns: Ten years after the rise of adversarial machine learning, Pattern Recognition 84 (2018) 317–331.

[3] I. J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, in: International Conference on Learning Representations (ICLR), 2015.

[4] S. G. Finlayson, J. D. Bowers, J. Ito, J. L. Zittrain, A. L. Beam, I. S. Kohane, Adversarial attacks on medical machine learning, Science 363 (6433) (2019) 1287–1289.

[5] E. Goan, C. Fookes, Bayesian neural networks: An introduction and survey, in: Case Studies in Applied Bayesian Data Science, Springer, 2020, pp. 45–87.

[6] D. Dera, G. Rasool, N. Bouaynaya, Extended Variational Inference for Propagating Uncertainty in Convolutional Neural Networks, in: 2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP), IEEE, 2019, pp. 1–6.

[7] A. Doucet, A. M. Johansen, A tutorial on particle filtering and smoothing: Fifteen years later, Handbook of Nonlinear Filtering 12 (2009) 656–704.

[8] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440.

[9] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer, 2015, pp. 234–241.

[10] G. Cao, Z. Sun, C. Wang, H. Geng, H. Fu, Z. Yin, M. Pan, Rasnet: Renal automatic segmentation using an improved u-net with multi-scale perception and attention unit, Pattern Recognition (2024) 110336.

[11] D. E. Cahall, G. Rasool, N. C. Bouaynaya, H. M. Fathallah-Shaykh, Inception modules enhance brain tumor segmentation, Frontiers in computational neuroscience 13 (2019) 44.

[12] M. U. Rehman, S. Cho, J. H. Kim, K. T. Chong, Bu-net: Brain tumor segmentation using modified u-net architecture, Electronics 9 (12) (2020) 2203.

[13] Q. Yan, S. Liu, S. Xu, C. Dong, Z. Li, J. Q. Shi, Y. Zhang, D. Dai, 3d medical image segmentation using parallel transformers, Pattern Recognition 138 (2023) 109432.

[14] H. Thisanke, C. Deshan, K. Chamith, S. Seneviratne, R. Vidanaarachchi, D. Herath, Semantic segmentation using vision transformers: A survey, Engineering Applications of Artificial Intelligence 126 (2023) 106669.

[15] H. Xiao, L. Li, Q. Liu, X. Zhu, Q. Zhang, Transformers in medical image segmentation: A review, Biomedical Signal Processing and Control 84 (2023) 104791.

[16] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al., Segment anything, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 4015–4026.

[17] Y. Huang, X. Yang, L. Liu, H. Zhou, A. Chang, X. Zhou, R. Chen, J. Yu, J. Chen, C. Chen, et al., Segment anything model for medical images?, Medical Image Analysis 92 (2024) 103061.

[18] Y. Gal, Z. Ghahramani, Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning, in: International Conference on Machine Learning, 2016, pp. 1050–1059.

[19] B. Lakshminarayanan, A. Pritzel, C. Blundell, Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles, in: Advances in Neural Information Processing Systems, 2017, pp. 6402–6413.

[20] A. Kendall, V. Badrinarayanan, R. Cipolla, Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding, in: Proceedings of the British Machine Vision Conference (BMVC), 2017.

[21] A. Jungo, R. McKinley, R. Meier, U. Knecht, L. Vera, J. Pérez-Beteta, D. Molina-García, V. M. Pérez-García, R. Wiest, M. Reyes, Towards Uncertainty-Assisted Brain Tumor Segmentation and Survival Prediction, in: International MICCAI Brainlesion Workshop, Springer, 2017, pp. 474–485.

[22] A. G. Roy, S. Conjeti, N. Navab, C. Wachinger, A. D. N. Initiative, et al., Bayesian quicknat: Model uncertainty in deep whole-brain segmentation for structure-wise quality control, NeuroImage 195 (2019) 11–22.

[23] A. Largent, J. De Asis-Cruz, K. Kapse, S. D. Barnett, J. Murnick, S. Basu, N. Andersen, S. Norman, N. Andescavage, C. Limperopoulos, Automatic brain

segmentation in preterm infants with post-hemorrhagic hydrocephalus using 3d bayesian u-net, Human brain mapping 43 (6) (2022) 1895–1916.

[24] A. J. Larrazabal, C. Martínez, J. Dolz, E. Ferrante, Orthogonal ensemble networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2021, pp. 594–603.

[25] K. Kamnitsas, W. Bai, E. Ferrante, S. McDonagh, M. Sinclair, N. Pawlowski, M. Rajchl, M. Lee, B. Kainz, D. Rueckert, et al., Ensembles of multiple models and architectures for robust brain tumour segmentation, in: International MICCAI brainlesion workshop, Springer, 2017, pp. 450–462.

[26] T. Buddenkotte, L. E. Sanchez, M. Crispin-Ortuzar, R. Woitek, C. McCague, J. D. Brenton, O. Öktem, E. Sala, L. Rundo, Calibrating ensembles for scalable uncertainty quantification in deep learning-based medical image segmentation, Computers in Biology and Medicine 163 (2023) 107096.

[27] B. Ghoshal, A. Tucker, B. Sanghera, W. Lup Wong, Estimating uncertainty in deep learning for reporting confidence to clinicians in medical image segmentation and diseases detection, Computational Intelligence 37 (2) (2021) 701–734.

[28] C. F. Baumgartner, K. C. Tezcan, K. Chaitanya, A. M. Hötker, U. J. Muehlematter, K. Schawkat, A. S. Becker, O. Donati, E. Konukoglu, Phiseg: Capturing uncertainty in medical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2019, pp. 119–127.

[29] H. Li, Y. Nan, J. Del Ser, G. Yang, Region-based evidential deep learning to quantify uncertainty and improve robustness of brain tumor segmentation, Neural Computing and Applications 35 (30) (2023) 22071–22085.

[30] X. Guo, X. Lin, X. Yang, L. Yu, K.-T. Cheng, Z. Yan, Uctnet: Uncertainty-guided cnn-transformer hybrid networks for medical image segmentation, Pattern Recognition 152 (2024) 110491.

[31] G. E. Hinton, D. Van Camp, Keeping the neural networks simple by minimizing the description length of the weights, in: Proceedings of the sixth annual conference on Computational learning theory, 1993, pp. 5–13.

[32] A. Graves, Practical Variational Inference for Neural Networks, in: Advances in neural information processing systems, 2011, pp. 2348–2356.

[33] C. Blundell, J. Cornebise, K. Kavukcuoglu, D. Wierstra, Weight uncertainty in neural networks, in: Proceedings of the 32nd International Conference on Machine Learning (ICML), 2015, pp. 1613–1622.

[34] M. Jun, G. Cheng, W. Yixin, A. Xingle, G. Jiantao, Y. Ziqi, Z. Minqing, L. Xin, D. Xueyuan, C. Shucheng, W. Hao, M. Sen, Y. Xiaoyu, N. Ziwei, L. Chen, T. Lu, Z. Yuntao, Z. Qiongjie, D. Guoqiang, H. Jian, COVID-19 CT Lung and Infection Segmentation Dataset (Apr. 2020). doi:10.5281/zenodo.3757476.

[35] M. Antonelli, A. Reinke, S. Bakas, K. Farahani, A. Kopp-Schneider, B. A. Landman, G. Litjens, B. Menze, O. Ronneberger, R. M. Summers, et al., The medical segmentation decathlon, Nature Communications 13 (1) (2022) 4128.

[36] B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, et al., The multimodal brain tumor image segmentation benchmark (brats), IEEE transactions on medical imaging 34 (10) (2014) 1993–2024.

[37] H. M. Fathallah-Shaykh, A. DeAtkine, E. Coffee, E. Khayat, A. K. Bag, X. Han, P. P. Warren, M. Bredel, J. Fiveash, J. Markert, N. Bouaynaya, L. B. Nabors, Diagnosing growth in low-grade gliomas with and without longitudinal volume measurements: A retrospective observational study, PLoS Med 16 (5) (2019) e1002810.

[38] K. Hoebel, V. Andrearczyk, A. Beers, J. Patel, K. Chang, A. Depeursinge, H. Müller, J. Kalpathy-Cramer, An exploration of uncertainty information for segmentation quality assessment, in: Medical Imaging 2020: Image Processing, Vol. 11313, International Society for Optics and Photonics, 2020, p. 113131K.

[39] Y. Liu, X. Chen, C. Liu, D. Song, Delving into transferable adversarial examples and black-box attacks, in: International Conference on Learning Representations (ICLR), 2017.

[40] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu, Towards deep learning models resistant to adversarial attacks, in: International Conference on Learning Representations (ICLR), 2018.

[41] R. Mehta, A. Filos, Y. Gal, T. Arbel, Uncertainty evaluation metric for brain tumour segmentation, in: Medical Imaging with Deep Learning (MIDL), 2020.

[42] Z. Chang, Will ai improve tumor delineation accuracy for radiation therapy?, Radiology 291 (3) (2019) 687–688.

[43] G. Carannante, N. C. Bouaynaya, Bayesian deep learning detection of anomalies and failure: Application to medical images, in: 2023 IEEE 33rd International Workshop on Machine Learning for Signal Processing (MLSP), IEEE, 2023, pp. 1–6.

[44] G. Carannante, D. Dera, G. Rasool, N. C. Bouaynaya, Self-compression in bayesian neural networks, in: 2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP), IEEE, 2020, pp. 1–6.