2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

CVAE-SM: A Conditional Variational Autoencoder with Style Modulation for Efficient Uncertainty Quantification

Amin Ullah†, Taiqing Yan†, Li Fuxin

Abstract— **Deep learning has brought transformative advancements to object segmentation, especially in marine robotics contexts such as waste management and subaquatic infrastructure oversight. However, a central challenge persists: calibrating the prediction confidence of the model to ensure robust and reliable outcomes, especially within the demanding underwater environment. Existing solutions for estimating uncertainty are often computationally intensive and have largely centered around Bayesian neural networks or ensemble methods. In this paper, we present a Conditional Variational Autoencoder-based framework (CVAE-SM), which is capable of generating diverse latent codes for improved uncertainty quantification in image segmentation. Our method, enhanced by a style modulator, merges content features, and latent codes more effectively, leading to refined prediction of uncertainty levels. We further introduce a dataset of perturbed underwater images to benchmark uncertainty quantification in this domain. The proposed model not only surpasses peers in segmentation metrics but also matches ensemble models in uncertainty predictions, all while being 2.5 times faster.**

I. INTRODUCTION

Deep learning-based object segmentation has demonstrated impressive performance in numerous practical applications, ranging from self-driving vehicles to medical diagnostics. Notwithstanding, a significant drawback of many deep learning models is their generation of point estimates, leading to a lack of insight into the model's certainty regarding its predictions [1], [2]. In many situations, understanding the confidence level of the model, specifically its capacity to discern between systematic decision-making and random guessing, becomes not just beneficial but vital. This is particularly applicable in demanding contexts such as underwater object analyses where underwater cameras on sea robots are utilized in marine waste management and the maintenance of subaquatic infrastructures [3].

In machine learning models, two main forms of uncertainty are typically explored: aleatoric and epistemic [4], [5]. Aleatoric uncertainty usually stems from inherent ambiguities in the labeling process, for example, when multiple individuals cannot agree on the labeling of an image. This was exemplified in a study on CIFAR-10-H by [6], where multiple participants were asked to label low-resolution images to measure human uncertainty. This type of uncertainty is irreducible, persisting even when infinite data is available, and can be addressed primarily through additional features derived from human input. Conversely, epistemic uncertainty arises due to the inability of a model to generalize effectively to test data. This can be mitigated with larger datasets or improved models. In this paper, our analysis focuses on epistemic uncertainty within the context of underwater object segmentation using data collected from sea robots.

The prevailing hypothesis suggests that an accurate approximation of epistemic uncertainty would enable deep learning models to better understand their limitations when dealing with unseen test data distributions. Despite its significance, quantifying uncertainty remains a largely unresolved issue in segmentation tasks. Researchers in the machine learning community have addressed this challenge with Bayesian neural networks, where model parameters are learned as a distribution rather than fixed points. For example, [7] employed Monte Carlo Dropout, a technique that randomly nullifies certain model parameters during multiple forward passes in the inference stage. Perhaps the most well-known approach is the ensemble method [8]. It involves training the same model with different initial parameters to capture the stochastic nature of the learning process. Each of these methods produces a distribution of outputs from a single input, which can then be used to calculate uncertainty. However, these techniques tend to be computationally intensive due to the multiple forward passes required to derive the output distributions.

To alleviate the computational requirements, this paper introduces a Conditional Variational Autoencoder (CVAE) based framework. The main advantage of utilizing CVAE for this task is its inherent capability to learn a conditional density model over diverse segmentation masks. It allows us to generate multiple predicted segmentation masks, enabling a refined representation of segmentation uncertainty. CVAE [9], as an extension of Variational Autoencoder (VAE) [10], has found its place in many applications beyond segmentation and shown its versatility and capability in computer vision tasks. From superpixel-wise variational autoencoders for image background modeling [11] to generative segmentation models using U-Net [12], the various applications of CVAE attest to its adaptability. Moreover, some studies, like [13], have ventured into tasks like forecasting uncertain trajectories, others, such as [14], have leveraged it for saliency detection. Our CVAE-SM conditionally trains the latent distributions through reconstruction using both prior and posterior encoders. Unlike other approaches, our method iterates solely through the decoder to generate output distributions, bypassing the need to engage the entire network. We propose a novel style modulator that learns

†A. Ullah, T. Yan contributed equally to this work

A. Ullah, T. Yan, and L. Fuxin are with the Collaborative Robotics and Intelligent Systems (CoRIS) Institute, Oregon State University, OR, United States. ullaham@oregonstate.edu

Code and Dataset: https://github.com/aminullah6264/CVAE-SE

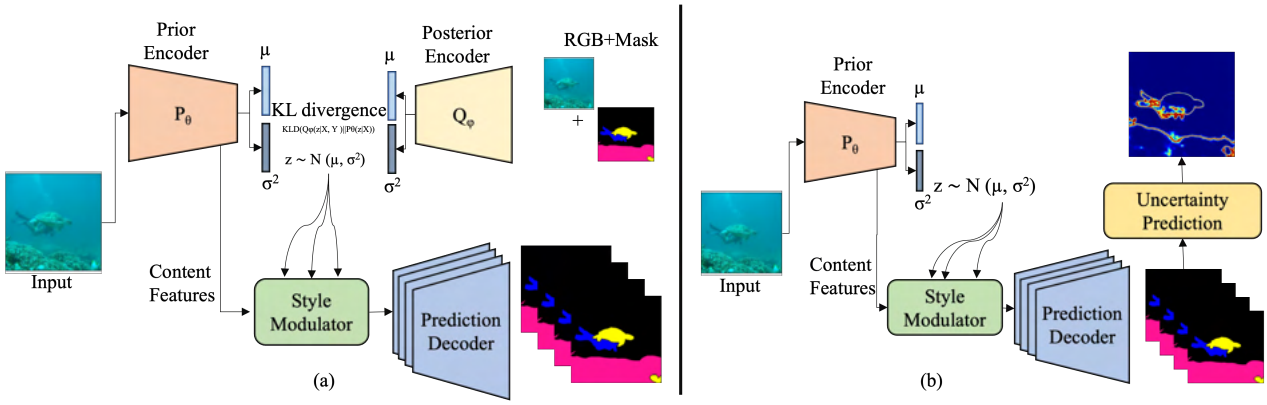

Fig. 1. Illustration of the proposed framework for uncertainty quantification in underwater object segmentation. The framework is comprised of four primary modules: the Prior Encoder and Posterior Encoder generate conditional latent codes, the Style Modulator adjusts latent codes and content features, and the Decoder generates final segmentation maps. During testing, we sample latent codes from a normal distribution guided by the mean and standard deviation of the Prior encoder. This approach yields a diverse array of segmentation maps, facilitating robust uncertainty quantification.

stochastic features from the conditional latent space to bring more diversity to the decoded predictions. Finally, we utilize entropy over the predicted distribution to calculate pixel-level uncertainty in segmentation. The following are our main contributions:

- We propose a Conditional Variational Autoencoder (CVAE) based framework that generates diverse latent codes, allowing effective stochastic inference for image segmentation and uncertainty quantification.
- Generating segmentation maps directly from latent codes do not yield optimal results. To overcome this, we implemented a style modulator that facilitates a more effective fusion of content features and latent codes. This, in turn, generates diverse outputs, thereby improving uncertainty quantification.
- We created a new benchmark for uncertainty quantification in underwater images by applying four different levels of relevant perturbations to the underwater images.
- The proposed model has achieved better IoU and F1 scores in the segmentation task. The model matches the performance of ensemble models in uncertainty quantification while being 2.5 times faster.

II. PROPOSED FRAMEWORK

This section describes our probabilistic model designed for uncertainty quantification in underwater object segmentation. The foundation of this model is a conditional variational autoencoder that focuses on learning a broad distribution of segmentation maps rather than confining itself to a singular prediction. Define the training dataset as $\xi = \{(X_i, Y_i)\}_{i=1}^{N}$, with X_i as the RGB input and Y_i as the corresponding ground truth segmentation map. The complete structure of the model during the training and testing phases is depicted in Figures 1(a) and 1(b).

The architecture of our network is divided into four primary components: *Prior Encoder:* This translates the RGB input X_i into a set of low-dimensional latent variables $z_i \in \mathbb{R}^K$, where K represents the latent space dimensionality. *Posterior Encoder:* This operates in a manner analogous to the Prior Encoder but processes both X_i and Y_i to produce the latent variables. *Style Modulator:* Refines the stochastic latent codes z_i and deterministic content features obtained from the Prior Encoder. *Prediction Decoder:* Generates segmentation maps utilizing multiple sampled latent codes.

During the testing stage, an additional module is implemented: *The Uncertainty Quantification Module:* This module calculates entropy over the stochastic segmentation maps to predict uncertainty.

A. Stochastic Segmentation Mapping via CVAE

In our computational framework, the CVAE serves as a critical component, designed to transform the prior distribution into a conditioned Gaussian distribution. Specifically, the characteristics of this Gaussian distribution are contingent on the input data X, thereby enabling a more refined and adaptive encoding process.

First, the content features F are extracted from the layer preceding the generation of μ and σ in the prior encoder. These features F capture essential content-based information of the input data X, providing context for the stochastic segmentation process. Subsequently, the latent variable z is sampled from the Gaussian distribution $P_\theta(z|X)$, which is conditioned on the input variable X. This conditional distribution effectively captures the underlying structural dependencies between the input data and the latent space. Subsequently, the posterior distribution of z is denoted as $Q_\phi(z|X, Y)$, which describes how well the latent variables z can explain the observed data Y when conditioned on X.

The loss function corresponding to the CVAE, denoted as L_{CVAE}, is mathematically formulated as:

$$L_{\text{CVAE}} = \mathbb{E}_{(z,F)\sim P_\theta(z,F|X)}[-\log P_w(Y|X; z, F)] + D_{\text{KL}}(Q_\phi(z|X; Y)||P_\theta(z|X)); \quad (1)$$

In equation (1), the first term computes the expectation of the negative log-likelihood of generating the observed data Y, when the model is conditioned on X and sampled from z. Essentially, it quantifies how well the model can reconstruct Y from the latent variable z and the input X.

The second term, $D_{\text{KL}}(Q_\phi(z|X;Y)||P_\theta(z|X))$, is the Kullback-Leibler Divergence, serving as a regularization term. It quantifies the dissimilarity between the prior distribution $P_\theta(z|X)$ and the posterior distribution $Q_\phi(z|X,Y)$. This term acts as a constraint that aims to minimize the encoding error, compelling the posterior distribution to closely approximate the prior. Without the Q encoder, the network becomes unstable in training. We opt for D_{KL} due to its effectiveness in quantifying the divergence between two distributions [15].

By balancing these two terms, the CVAE is capable of representing the log-likelihood $P(Y|X)$ while maintaining a regularization encoding error $D_{\text{KL}}(Q_\phi(z|X;Y)||P_\theta(z|X))$.

Adhering to conventional practices for CVAEs, as outlined in the paper by Sohn et al. [9], our network is configured to generate multiple segmentation maps that predict underwater object uncertainty. Through this configuration, we aim to offer a robust, data-driven approach for uncertainty quantification in underwater object detection tasks.

B. Prior and Posterior Encoders

The prior encoder $P_\theta(z|X)$ is responsible for mapping an input RGB image X into a low-dimensional latent feature space. Utilizing an architecture similar to the prior encoder and given the ground truth segmentation masks Y, we define $Q_\phi(z|X,Y)$ as the posterior encoder. Both the prior P_θ and posterior Q_ϕ encoders utilize a pre-trained ImageNet backbone to convert the input RGB image X into content features and a latent Gaussian variable $z \sim \mathcal{N}(\mu, \text{diag}(\sigma^2))$.

During the training phase, we sample multiple latent vectors from the prior encoder, which are then fed into the style modulator to generate a varied ensemble of latent vectors suitable for the decoder.

C. Style Modulator

In StyleGAN2 [16], the idea behind the modulation and demodulation operations is to give the generator more control over how styles are infused into generated images. This enables the network to create images that are not only high-quality but also highly diverse, as it can tweak different styles at different layers to get very specific kinds of images. Inspired by this approach we applied style modulation on latent codes as a style vector z and content features F to bring more diversity to the decoded predictions. Instead of merely concatenating latent codes and content features before passing them to the decoder, we employ a modulation step. The modulation operation scales each input feature map of the convolution based on the given style vector z. In essence, this scaling can also be realized by modifying the convolution weights as:

$$w'_{ijk} = (Az)_i \cdot w_{ijk} \tag{2}$$

where A is an affine transformation, w and w' are the original and modulated weights, respectively. j and k enumerate the feature maps and spatial dimension of the convolution, respectively.

Post modulation and scaling, the demodulation step is applied, which aims to restore output activations to a unit standard deviation. this is attained by scaling each output feature map which can be incorporated directly into the convolution weights:

$$w''_{ijk} = \frac{w\prime_{ijk}}{\sqrt{\sum_{i,k}(w\prime_{ijk})^2}} \tag{3}$$

Ultimately, by utilizing Equations (1), (2) and (3), the entire style block can be represented as a single convolution layer with weights adjusted based on Az. Finally, the w''_{ijk} are convolved over the features F to obtain the stochastic features for the decoder. This allows us to fine-tune the interactions between the latent and content spaces, ensuring that each brings its own influence on the decoded output pixel probabilities. Through this, we manage to preserve a more diverse set of decoded predictions in our model.

D. Prediction Decoder

The prior encoder generates both deterministic content features and stochastic latent codes derived from its parameters (μ, σ). Rather than solely relying on deterministic content features, our proposed method modulates these alongside the stochastic latent codes, producing stochastic features, subsequently fed into the prediction decoder. In our experiments, we employed the Fully Convolutional Network (FCN) [17] as our decoder. This decoder processes the stochastic features to yield predictions of the size of the segmentation map, with logits channels corresponding to the total number of classes within the dataset. Throughout both training and testing phases, given multiple stochastic features, we can derive a variety of predictions. This is achieved by sampling latent codes from the prior encoder, specifically $z \sim \mathcal{N}(\mu, \text{diag}(\sigma^2))$, multiple times. This procedure yields a rich collection of segmentation maps, effectively granting us deep ensemble-level performance. Notably, this is attained without the need for multiple passes through the entire network. Instead, we iterate over just a few layers of the decoder multiple times, ensuring efficiency.

E. Uncertainty quantification

In traditional segmentation tasks where softmax is applied across the class dimension, entropy can be computed directly over the class predictions to quantify uncertainty. This entropy value provides an indication of the confidence of the model in assigning a pixel to a specific class. In contrast, our proposed methodology utilize a sigmoid function on each individual pixel. This adaptation necessitates an alternative mechanism for estimating uncertainty.

Assume we are given a tensor of predicted logits with dimensions $[K, C, H, W]$, where K is the number of ensemble models and C the number of classes. We first invoke the sigmoid function on all the logits, then entropy is calculated for each element separately using the equation $H(p) = -p \cdot \log_2(p) - (1-p) \cdot \log_2(1-p)$. For an entropy tensor E, the pixel-level uncertainty is first extracted by considering the peak entropy across the class dimension. This

is mathematically represented as:

$$U'_{e,i,j} = \max_{c}(E_{e,c,i,j}) \tag{4}$$

where $e \in \{1, \ldots, K\}$ represents the model id, $c \in \{1, \ldots, C\}$ is a category, and i and j iterate over the image height and width, respectively.

The final step in our methodology then focuses on acquiring the maximum entropy value across the ensemble dimension for each pixel, thereby capturing the highest uncertainty from all ensemble models:

$$U_{i,j} = \max_{e}(U'_{e,i,j}) \tag{5}$$

The uncertainty for each pixel is derived from the ensemble model that exhibits the highest entropy, ensuring a comprehensive representation of the model uncertainty quantification. The rationale behind this maximum value selection is rooted in our "one versus many" classification strategy. Given this framework, a pixel can display a spectrum of entropy scores across categories and models. Simply averaging these values might inadvertently suppress pronounced uncertainties, as the high and low entropy values could potentially cancel each other out. By considering the maximum entropy value, we ensure a holistic representation of uncertainties. This approach guarantees that even if a single model/class presents significant uncertainty for a pixel, it is duly recognized and represented, offering a deeper insight into the models' confidence throughout the spatial expanse of the image.

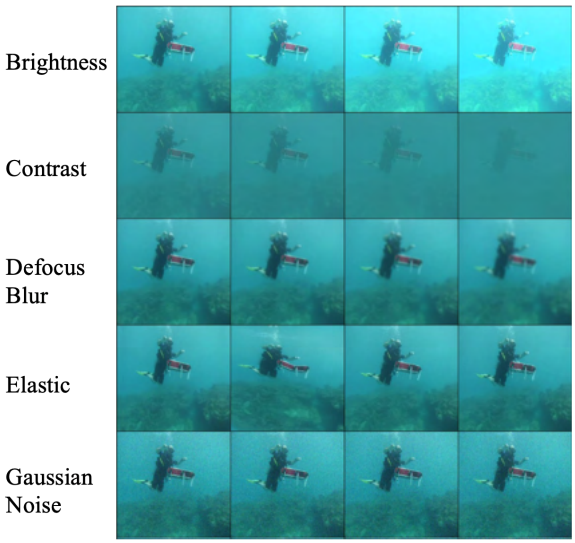

Fig. 2. Example images from the dataset, demonstrating different levels of perturbations applied to underwater images for uncertainty quantification.

III. EXPERIMENTS

A. Implementation Details

We implemented the proposed approach using PyTorch and employed the FCN model with a ResNet50 backbone, pre-trained on a subset of the COCO train2017 dataset. For the posterior encoder, where the input becomes an 8-channel tensor resulting from the concatenation of X and Y, we clone the weights of the original 3 channels of the first layer and utilize them as initializations for the extended channels before fine-tuning. We trained ten distinct models as a baseline ensemble model. For MC dropout, ten forward passes from a single model determined the outputs. While our method samples only four segmentation maps from the decoder during training, ten segmentation maps are sampled during testing. Models are trained with a batch size of six images per GPU, distributed across four GPUs for each task, utilizing a learning rate of 1e-4. We trained for 150 epochs, with scheduled learning rate reductions at the 80th and 130th epochs.

B. Evaluation Metrics

We utilize the commonly use IoU and F1 to measure segmentation accuracy. IoU evaluates the intersection-over-union overlap between the prediction and ground truth segmentation maps, while F1 is a geometric average of the precision and recall. For measuring uncertainty, we utilized the Brier score and Expected Calibration Error (ECE) [18]. The Brier score computes the squared error between a predicted probability vector and the one-hot encoded true response. On the other hand, ECE assesses the alignment between a model's predicted probabilities and actual outcomes. To calculate ECE, predicted probabilities are bucketed into intervals. For each bucket, there's a comparison between the average predicted probability and the actual accuracy of the predictions within that interval. ECE is then determined by averaging the discrepancies between these two quantities across all buckets. A lower ECE indicates that the model's predicted probabilities are better calibrated to actual outcomes.

C. Datasets and Perturbations for Uncertainty Evaluation

Our proposed method is evaluated using the Semantic Segmentation of Underwater Imagery (SUIM) dataset [19], a comprehensive resource that has five distinct underwater objects classes: robots, fish, human divers, reefs, shipwrecks, and general background. The dataset is well-structured, consisting of 1,525 RGB images for training, and provides a separate set of 110 test images that are used exclusively for the benchmark evaluation of semantic segmentation algorithms. Notably, the images in the SUIM dataset have been carefully curated to include a broad spectrum of natural underwater environments as well as various scenarios that involve human-robot collaborations.

To enrich our uncertainty evaluation, we employed a variety of image perturbations to the test set, inspired by the benchmark study [20]. These perturbations are visually demonstrated in Figure 2 and are applied across four different intensity levels on the SUIM test dataset, leaving the ground truths unchanged. The suite of perturbations includes Brightness variations influenced by different light conditions, Contrast changes affected by lighting and object color, Gaussian Noise often present in low-light conditions, and Shot Noise, also known as Poisson noise, which is electronic noise influenced by the inherent discrete nature of

TABLE I

A DETAILED COMPARISON OF IOU, F1, ECE, AND BRIER SCORES ACROSS DIFFERENT METHODS FOR FOUR PERTURBATION LEVELS APPLIED ON THE TEST SET OF SUIM DATASET. THE HIGHEST AND MOST SIMILAR SCORES IN EACH COLUMN ARE HIGHLIGHTED IN **BOLD** WHILE THE SECOND HIGHEST IS IN *italic*.

Method	Ori	Lv1	Lv2	Lv3	Lv4
		IoU			
Vanilla	0.735	0.617	0.549	0.498	0.397
Dropout	0.732	0.615	0.547	0.496	0.395
Ensemble	*0.757*	*0.634*	*0.565*	**0.515**	**0.411**
CVAE-SM	**0.764**	**0.645**	**0.571**	*0.508*	**0.410**
		F1			
Vanilla	0.701	0.542	0.474	0.438	0.342
Dropout	0.698	0.540	0.472	0.436	0.340
Ensemble	*0.734*	*0.564*	**0.491**	**0.457**	**0.353**
CVAE-SM	**0.740**	**0.578**	**0.493**	**0.445**	**0.350**
		ECE			
Vanilla	0.079	0.099	0.113	0.121	0.143
Dropout	*0.077*	0.097	*0.111*	*0.118*	*0.140*
Ensemble	0.078	**0.091**	**0.102**	**0.112**	**0.137**
CVAE-SM	**0.075**	*0.095*	0.122	0.168	0.171
		Brier Score			
Vanilla	0.021	0.029	0.034	0.038	0.046
Dropout	0.021	0.029	0.034	0.037	0.046
Ensemble	**0.018**	**0.024**	**0.029**	**0.033**	**0.041**
CVAE-SM	*0.019*	*0.026*	0.032	*0.035*	*0.045*

light. Additional perturbation types like Impulse Noise, often caused by bit errors, Defocus, Motion and Zoom Blurs, are also applied. We also applied Elastic Transformations that deform the image, Pixelation due to upscaling, and JPEG compression artifacts. This extensive set of perturbations allows us to perform a comprehensive evaluation of the model performance, thereby providing valuable insights into its robustness and adaptability under a variety of challenging real-world underwater conditions.

Additionally, we employed the Underwater Image Enhancement Benchmark Dataset [21] for out-of-distribution (OOD) uncertainty quantification. This dataset comprises 950 raw underwater images, with 890 of them possessing corresponding reference images. The remaining 60 images, which lack satisfactory reference counterparts, are designated as OOD test. Since this dataset does not include pixel-level segmentation ground truth labels, our analysis is limited to reporting predictive entropy comparisons between the raw and reference images.

D. Comparison with Other Approaches

The results presented in Table I provide a quantitative measure of performance for various methods against the proposed CVAE-SM. In terms of IoU, CVAE-SM exhibits superior performance compared to the other methods across all levels except Lv3. Similarly, Analyzing the F1 scores reveals that the CVAE-SM to improve over baselines in this aspect. To evaluate uncertainty, ECE measures the reliability of probabilistic predictions. A lower ECE indicates better reliability. In the initial levels (Ori to Lv2), CVAE-SM demonstrates a commendable performance, showing the lowest ECE for original data compared to the others. However, in Lv3 and Lv4, there is a noticeable surge in the ECE for CVAE-SM to 0.16 and 0.17. Lastly, the Brier Score assesses the accuracy of probabilistic predictions, with a lower score indicating better performance. The ensemble method seems to marginally outperform CVAE-SM in most levels, but the difference remains minimal. For instance, at Ori, Ensemble records a score of 0.018, just a slight edge over CVAE-SM at 0.019. This trend continues throughout the levels, indicating that while CVAE-SM demonstrates significant potential, there remains room for refining its uncertainty estimation.

TABLE II

PREDICTIVE ENTROPY COMPARISONS OF DIFFERENT METHODS ON RAW AND REFERENCE DATA OF OUT-OF-DISTRIBUTION UNDERWATER IMAGE ENHANCEMENT DATASET.

Method	Raw Data	Reference Data
Vanilla	0.23	0.25
Dropout	0.31	0.32
Ensemble	0.52	0.53
CVAE-SM	0.28	0.29

Next, we assess the performance of various methods for uncertainty quantification using the out-of-distribution underwater image enhancement dataset (Table II). We believe that on this OOD dataset uncertainty should be higher because the images have never been seen during training. However, the vanilla approach produces the lowest entropy values, suggesting high (over)confidence in its predictions. The Ensemble method showcases the highest predictive entropy values for both raw and reference data, indicating greater uncertainty in its outputs. Our proposed CVAE-SM method demonstrates an improvement over the Vanilla method, registering entropy values of 0.28 for raw data and 0.29 for reference data. Intriguingly, the performance of CVAE-SM is almost on par with the Dropout approach, which records entropy values of 0.31 and 0.32 for raw and reference data, respectively.

Table III shows the runtime of different approaches. Our speed comparison was conducted on an NVIDIA DGX system, dedicated 16 CPU cores and a single Tesla V100 32GB GPU for this task. The numbers are derived from averaging multiple runs under identical conditions, ensuring a fair and consistent benchmark. Primarily, our proposed CVAE-SM method offers a distinct efficiency advantage over both the Dropout and Ensemble methods. Clocking in at a mere 40 milliseconds, CVAE-SM operates at speeds over three times faster than its Dropout and Ensemble counterparts, which both require 140 milliseconds. The Ensemble method might outperform in certain levels, as evidenced in previous analyses, yet efficiency becomes a focal point in real-time or near-real-time applications. This is where the CVAE-SM combination of performance and rapid processing becomes invaluable. In scenarios like underwater robotics demanding prompt processing, CVAE-SM efficiency presents a discernible advantage and establishes it as an excellent choice for a variety of applications that prioritize prompt and trustworthy uncertainty quantification.

E. Visual Results Comparison

Figure 3 shows both the segmentation outcomes and the pixel-level uncertainty for an image with its original form and with four increasing levels of motion blur applied.

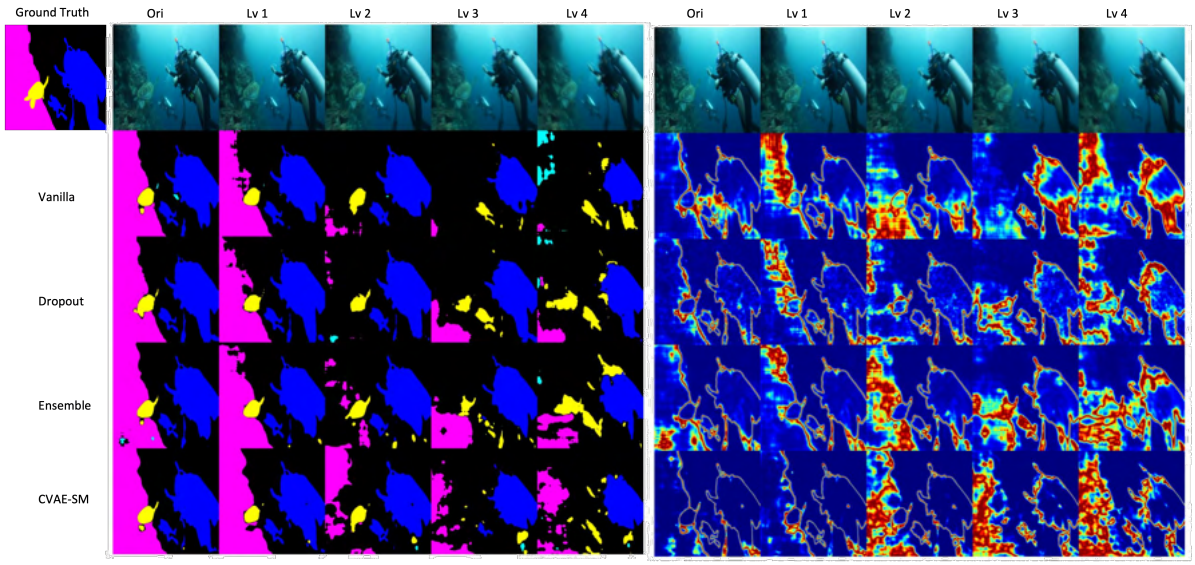

Fig. 3. Qualitative comparison of semantic segmentation prediction and uncertainty quantification on different levels of motion blur perturbation.

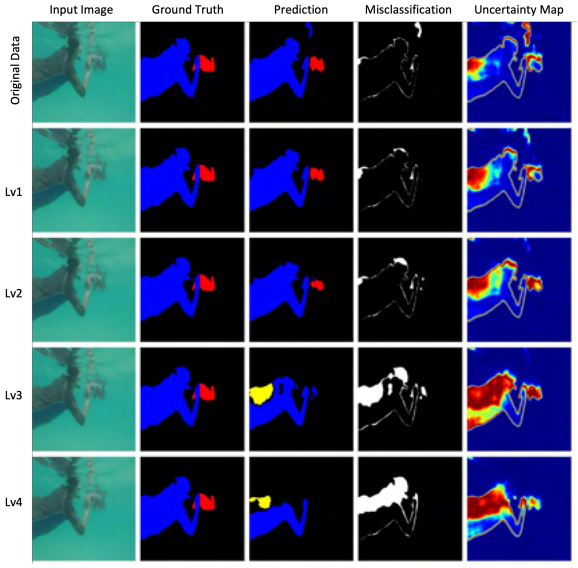

Fig. 4. Uncertainty quantification and miss-classification by CVAE-SM across zoom blur levels.

TABLE III

PROCESSING TIMES COMPARISON WITH OTHER UNCERTAINTY QUANTIFICATION METHODS.

Method	**Processing time (milliseconds)**
Vanilla	30
Dropout	140
Ensemble	140
CVAE-SM	40

In the context of segmentation, accurate pixels correspond to those matching the colors found in the ground truth segmentations. In the image on the right side, areas with high heatmap values indicate heightened uncertainty levels. As the perturbation level amplifies, the segmentation performance of all models gradually declines. Ideally, misclassified pixels should have a high uncertainty value as shown in Figure 4 column 4 and column 5, while correctly classified ones should exhibit a lower value. This correlation is effectively captured in the results of our proposed CVAE-SM model (as seen in the last row). For other methods, one can see that the left side segment of coral reef (depicted in pink color) was missed in most variants at perturbation levels higher than Lv 2, however, most of the approaches other than CVAE-SM attributed little uncertainty in this area. The CVAE-SM model not only surpasses the performance of both vanilla and dropout approaches but also places uncertainty at more appropriate locations compared with other uncertainty quantification approaches.

IV. CONCLUSIONS

In this paper, we presented the CVAE-SM framework, a robust solution for stochastic inference in image segmentation and uncertainty quantification. By employing a novel style modulator, we successfully fused deterministic content features with stochastic latent codes, amplifying both output diversity and uncertainty quantification. We also established a novel benchmark for underwater image uncertainty, introducing different perturbation levels to gauge model performance. Impressively, our model not only demonstrated superior segmentation accuracy through IoU and F1 scores but also rivaled ensemble models in uncertainty quantification, meanwhile running at a 2.5 times faster speed. We believe this research holds significant promise for real-world applications, particularly in underwater robotics applications where accuracy and efficiency are paramount.

ACKNOWLEDGMENT

This work is supported in part by ONR/NAVSEA contract N00024-10-D-6318 and ARO Cooperative Agreement Number W911NF-22-2-0149. The views in this paper are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office, Office of Naval Research, or the U.S. Government.

REFERENCES

[1] J. Gawlikowski, C. R. N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher *et al.*, "A survey of uncertainty in deep neural networks," *Artificial Intelligence Review*, pp. 1–77, 2023.

[2] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. Dillon, B. Lakshminarayanan, and J. Snoek, "Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift," *Advances in neural information processing systems*, vol. 32, 2019.

[3] J. Li, K. A. Skinner, R. M. Eustice, and M. Johnson-Roberson, "Watergan: Unsupervised generative network to enable real-time color correction of monocular underwater images," *IEEE Robotics and Automation letters*, vol. 3, no. 1, pp. 387–394, 2017.

[4] A. F. Psaros, X. Meng, Z. Zou, L. Guo, and G. E. Karniadakis, "Uncertainty quantification in scientific machine learning: Methods, metrics, and comparisons," *Journal of Computational Physics*, vol. 477, p. 111902, 2023.

[5] N. Tagasovska and D. Lopez-Paz, "Single-model uncertainties for deep learning," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[6] R. M. Battleday, J. C. Peterson, and T. L. Griffiths, "Improving machine classification using human uncertainty measurements," 2018.

[7] J. Mukhoti and Y. Gal, "Evaluating bayesian deep learning methods for semantic segmentation," *arXiv preprint arXiv:1811.12709*, 2018.

[8] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," *Advances in neural information processing systems*, vol. 30, 2017.

[9] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," *Advances in neural information processing systems*, vol. 28, 2015.

[10] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[11] B. Li, Z. Sun, and Y. Guo, "Supervae: Superpixelwise variational autoencoder for salient object detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 8569–8576.

[12] S. Kohl, B. Romera-Paredes, C. Meyer, J. De Fauw, J. R. Ledsam, K. Maier-Hein, S. Eslami, D. Jimenez Rezende, and O. Ronneberger, "A probabilistic u-net for segmentation of ambiguous images," *Advances in neural information processing systems*, vol. 31, 2018.

[13] J. Walker, C. Doersch, A. Gupta, and M. Hebert, "An uncertain future: Forecasting from static images using variational autoencoders," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*. Springer, 2016, pp. 835–851.

[14] J. Zhang, D.-P. Fan, Y. Dai, S. Anwar, F. S. Saleh, T. Zhang, and N. Barnes, "Uc-net: Uncertainty inspired rgb-d saliency detection via conditional variational autoencoders," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8582–8591.

[15] X. Yang, X. Yang, J. Yang, Q. Ming, W. Wang, Q. Tian, and J. Yan, "Learning high-precision bounding box for rotated object detection via kullback-leibler divergence," *Advances in Neural Information Processing Systems*, vol. 34, pp. 18 381–18 394, 2021.

[16] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8110–8119.

[17] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[18] M. P. Naeini, G. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using bayesian binning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 29, no. 1, 2015.

[19] M. J. Islam, C. Edge, Y. Xiao, P. Luo, M. Mehtaz, C. Morse, S. S. Enan, and J. Sattar, "Semantic segmentation of underwater imagery: Dataset and benchmark," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 1769–1776.

[20] D. Hendrycks and T. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," *arXiv preprint arXiv:1903.12261*, 2019.

[21] C. Li, C. Guo, W. Ren, R. Cong, J. Hou, S. Kwong, and D. Tao, "An underwater image enhancement benchmark dataset and beyond," *IEEE Transactions on Image Processing*, vol. 29, pp. 4376–4389, 2019.