Robustness Verification of an AUV-based Plankton Classifier

Abdelrahman Sayed Sayed[1], **Pierre-Jean Meyer**[1], **Asgeir J. Sørensen**[2], **Mohamed Ghazel**[1]

[1]COSYS-ESTAS, Université Gustave Eiffel, France

[2]Department of Marine Technology, Norwegian University of Science and Technology, Norway

[1]{abdelrahman.ibrahim, pierre-jean.meyer, mohamed.ghazel}@univ-eiffel.fr [2]asgeir.sorensen@ntnu.no

Paper

Co-funded by the European Union

Motivation & Impact

- **Ocean health monitoring:** assessing **planktonic standing stocks** in the upper water column is critical to understand the impact of **climate change** on ocean processes.
- **AUV-based observation:** AUVs with in-situ optical imaging (**SilCam**) and **AI-based classifiers** enable high-resolution imaging *(Davies et al. 2017; Saad et al. 2020)*.
- **Problem:** in-situ imaging is degraded by **environmental noise**, turbidity and **gas bubbles** from the AUV's own motion, routinely triggering **misclassifications** that require time-consuming manual validation by marine biologists.
- **Robustness gap**: CNN classifiers are vulnerable to **adversarial perturbations** *(Szegedy et al. 2014)*, while neural ODE are naturally robust against such perturbations *(Carrara et al. 2019)*, **formally verifying** their continuous-time dynamics **remains an open challenge**.

What our verification brings to ocean monitoring

- ✓ **Formal robustness guarantees** beyond empirical accuracy for in-situ plankton classification under bounded noise.
- ✓ An **automated filter** that flags ambiguous data and misclassifications **reducing the manual post-processing workload**.
- ✓ Advancing **reliable autonomous sampling** towards trustworthy long-term monitoring of planktonic life.

Neural Ordinary Differential Equations

- **Neural ODE** *(Chen et al. 2018)* model a **continuous-depth** transformation of a hidden state $z(t) \in \mathbb{R}^n$ as the solution of an initial value problem:

$$\frac{dz(t)}{dt} = f\big(z(t), t, \theta\big), \qquad z(t_0) = z_0,$$

where the vector field f is a neural network with trainable weights θ. The output $z(t_f)$ is evaluated using a numerical ODE solver.

- **General neural ODE (GNODE):** in practice the ODE block is embedded between neural network layers, yielding the composition $\mathcal{N} = g_{\text{post}} \circ \Phi \circ g_{\text{pre}} : \mathbb{R}^m \to \mathbb{R}^c$, where g_{pre} extracts features from the input images, $\Phi : z_0 \mapsto z(t_f)$ is the flow of the ODE, and g_{post} maps the final state to the c class scores.
- **Our proposed GNODE**: f is a single hidden-layer network, independent of t:

$$\frac{dz(t)}{dt} = W_2\,\sigma\big(W_1 z(t) + b_1\big) + b_2,$$

with activation functions $\sigma \in \{\text{ReLU}, \text{Tanh}\}$.

Problem Definition

- **Plankton Classifier:** the classifier $\mathcal{N}$ is the GNODE, and it maps an input image of $m = 192$ pixel values to $c = 7$ plankton class scores, and labels the image with the highest-scoring class. We consider a nominal image $x^\star$ that is *correctly* classified, i.e. the score of its true class ℓ is the highest of the 7 class scores.
- **Perturbation strategy:** environmental noise modeled as a bounded L_∞ perturbation of magnitude $\varepsilon = \alpha/255$ applied to the pixel values of the input image, where each attacked pixel in a subset P of $k = |P|$ pixels can vary independently by up to $\pm\varepsilon$ around its nominal value, which yields the set $\mathcal{X}_0$ of all resulting noisy images

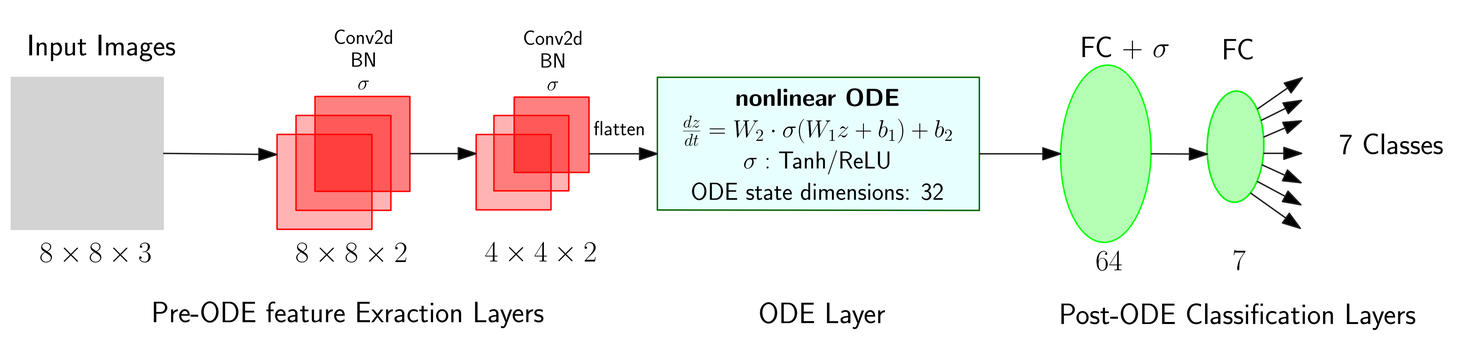
$$\mathcal{X}_0 = \Big\{x \in \mathbb{R}^m : \forall i \in P, |x_i - x_i^\star| \le \varepsilon;\ \forall i \notin P, x_i = x_i^\star\Big\},$$

with $k \in \{1, \ldots, 192\}$ attacked pixels, ranging from a single pixel up to the full image.

- **Robustness specification:** the correct label ℓ must remain dominant over the whole perturbation set. The robust output region is $\mathcal{S} = \big\{y \in \mathbb{R}^c : y_\ell > y_j\ \forall j \neq \ell\big\}$.
- **Verification goal:** certify that every reachable output keeps ℓ dominant, i.e. $\mathcal{N}(\mathcal{X}_0) \subseteq \mathcal{S}$.
- **Verdicts:** The verifier produces a `ROBUST` verdict when the output over-approximation of all reachable class scores stays within the robust region $\mathcal{S}$, `NOT ROBUST` when a counter-example violating the robustness specification is found, and `UNKNOWN` when no verdict is reached within the verification time limit.

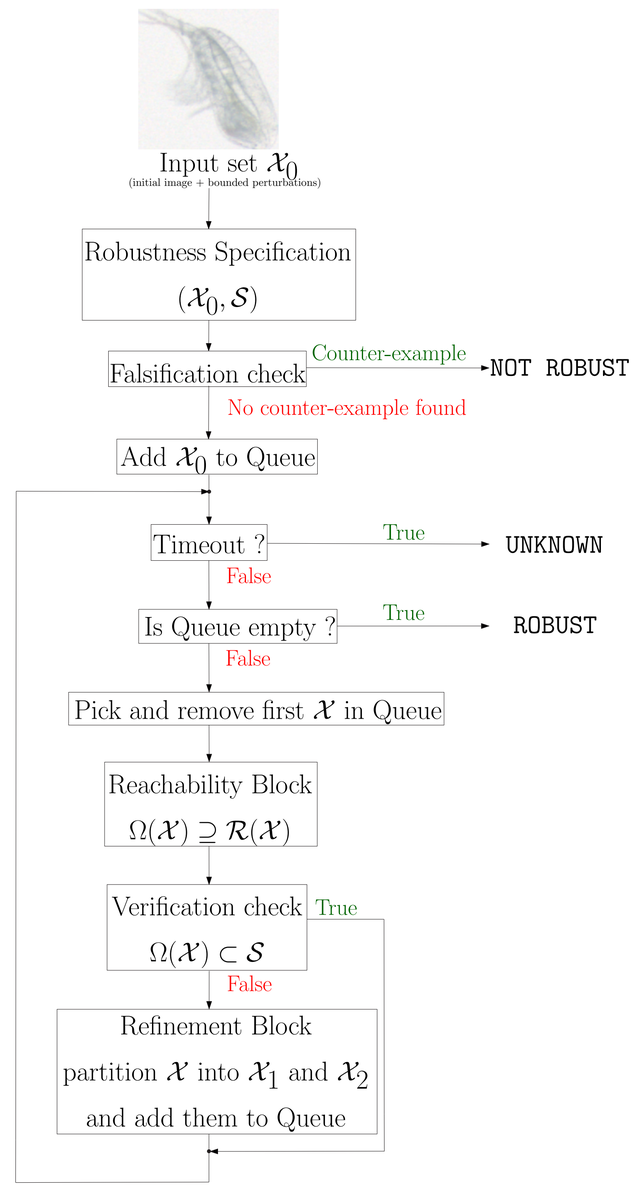
Classifier Architecture

- **Underwater plankton dataset:** in-situ **SilCam** images *(Davies et al. 2023)*, forming a dataset of 7738 images spanning 7 classes (bubble, copepod, diatom chain, fecal pellet, oil, oily gas, other).
- **Compact, verifiable architecture:** input images of 192 pixel values each ($8 \times 8 \times 3$) pass through two pre-ODE feature-extraction blocks (2D convolution + batch normalization + σ) down to a 32-dimensional vector, integrated by the **ODE block**, then classified by two fully-connected layers ($64 \to 7$ class scores).
- **Why compact?** small input and state dimensions yield **tighter over-approximations** and fewer dimensions to split during refinement ⇒ **lower verification cost**.



Underwater Image Verification Framework

- **Goal:** formally guarantee that a classified in-situ image remains robust against environmental or AUV-induced perturbations.
- **Phase 1 – Image falsification check:** evaluate the classified image against random samples of its noisy perturbations **before** any reachability analysis computation:
  - ✗ **Counter-example found** ⇒ terminate the verifier immediately with `NOT ROBUST` verdict for this image (at essentially zero cost).
  - ✓ **No violation** ⇒ proceed to the verification and refinement phase.
- **Phase 2 – Image verification and refinement:**
  1. **Reachability computation:** propagate the noisy input image through the classifier to compute a guaranteed **over-approximation** of all its possible output scores $\Omega(\mathcal{X}) \supseteq \mathcal{R}(\mathcal{X})$.
  2. **Verification check:** if $\Omega(\mathcal{X}) \subset \mathcal{S}$, i.e. correct class score dominant over the entire set, the current image is **verified**.
  3. **Refinement:** otherwise, the image is partitioned into smaller subsets by splitting the pixels that contribute most to the output uncertainty, and each subset is sent back to the reachability computation (step 1).
- **Termination:** every subset of the image is either verified ⇒ `ROBUST` or timeout ⇒ `UNKNOWN`.



Verifier architecture

Verification Results

Verification verdicts and runtimes obtained for both activation functions across different perturbation settings:

Table 1: Robustness verification results with ReLU activation

Pixels Attacked	Noise (L_∞)	Verdict	Time
1 / 192	0.1/255	ROBUST	128.1 s
1 / 192	0.5/255	ROBUST	104.2 s
10 / 192	0.1/255	ROBUST	104.8 s
192 / 192	0.01/255	ROBUST	108.9 s
192 / 192	5/255	NOT ROBUST	0 s
192 / 192	10/255	NOT ROBUST	0 s
192 / 192	1/255	Unknown	7200 s
192 / 192	0.05/255	Unknown	**7200 s**

Table 2: Robustness verification results with Tanh activation

Pixels Attacked	Noise (L_∞)	Verdict	Time
1 / 192	0.1/255	ROBUST	119.2 s
1 / 192	0.5/255	ROBUST	101.4 s
10 / 192	0.1/255	ROBUST	105.2 s
192 / 192	0.01/255	ROBUST	113.1 s
192 / 192	5/255	NOT ROBUST	0 s
192 / 192	10/255	NOT ROBUST	0 s
192 / 192	1/255	Unknown	7200 s
192 / 192	0.05/255	ROBUST	105.2 s