Enhancing the Dependability of Autonomous Surface Vehicles through Robustness Benchmarking of Real-time Object Detection Models

Yunjia Wang[a,e,*], Zihao Zhang[b], Kaizheng Wang[a,e], Holger Caesar[d], Jeroen Boydens[a], Davy Pissoort[c,e] and Mathias Verbeke[a,e]

[a]Department of Computer Science, KU Leuven, Bruges, 8200, Belgium
[b]School of Engineering and Applied Science, Columbia University, New York, 10027, USA
[c]Department of Electrical Engineering, KU Leuven, Bruges, 8200, Belgium
[d]Department of Cognitive Robotics, TU Delft, Delft, 2628 CD, Netherlands
[e]Flanders Make@KU Leuven, Leuven, Belgium

ARTICLE INFO

Keywords:
corruption robustness
real-time object detection
autonomous surfaces vehicles
artificial intelligence dependability
robustness benchmark

ABSTRACT

The Autonomous Surface Vehicle (ASV) market is expected to double by 2030, rapidly transforming maritime logistics through faster deliveries, lower costs, reduced risks from human error, and the potential to save human lives. ASVs depend on robust object detection models to ensure safe navigation. However, existing models are often susceptible to natural corruptions such as blur, noise, adverse weather, and occlusions—risks to perception robustness further intensified by the lack of domain-specific robustness benchmarks. To fill this gap, we propose the first waterborne-focused robustness benchmark, incorporating 25 synthetic corruptions (15 adapted from ImageNet-C plus 10 novel ones for ASVs) across five severity levels. We also incorporate mixed corruptions to capture real-world complexity. Building on three public waterborne datasets (SeaShips, SMD, SSAVE), we create SeaShips-C, SMD-C, and SSAVE-C, each augmented with our corruption suite. A comprehensive robustness evaluation is conducted on multiple sizes of YOLOv8, SSD, NanoDet-Plus, and RT-DETR, revealing critical vulnerabilities: e.g., YOLOv8n's mAP^{50} drops by 43.0% under contrast corruption on SeaShips-C, reaching a 59.5% decline when combined with raindrops. Larger variants (e.g., YOLOv8x) exhibit greater robustness, offering insights for safer deployments. Aligned with ISO/IEC TR 5469 and IEC 61508, our benchmark supports pre-deployment verification. By identifying risk-prone conditions, practitioners can apply targeted mitigation strategies, such as data augmentation and human oversight. To promote further research and support industrial practice, we provide open access to all benchmark datasets and code—which can also serve as a data augmentation resource to enhance model training.

1. Introduction

Maritime shipping accounts for around 90% of international cargo transportation, serving as the "lifeblood of the world economy" (Rødseth (2017)). With the rapid advancement of artificial intelligence and automation technologies, the development of Autonomous Surface Vehicles (ASVs) has increasingly gained attention in the last decade, as they offer the potential to revolutionize waterborne transportation and logistics. ASVs promise significant benefits in this domain, including faster deliveries, lower operation cost, enhanced environmental sustainability, and reduced human error risks. According to Rothblum (2000), 96% of maritime accidents are related to human errors.

The concept of an ASV was defined in Waterborne Technology Platform (2011), sparking a wave of research and development initiatives (Adams (2014); Bertram (2016); Komianos (2018); Negenborn et al. (2023)). Projects such as MUNIN have concluded that, in general, there are no major obstacles to the realization of fully autonomous vessels,

*Corresponding author
yunjia.wang@kuleuven.be (Y. Wang); zz2763@columbia.edu (Z. Zhang); kaizheng.wang@kuleuven.be (K. Wang); h.caesar@tudelft.nl (H. Caesar); jeroen.boydens@kuleuven.be (J. Boydens); davy.pissoort@kuleuven.be (D. Pissoort); mathias.verbeke@kuleuven.be (M. Verbeke)
ORCID(s): 0009-0006-5539-7051 (Y. Wang); 0009-0008-4003-0702 (Z. Zhang); 0000-0001-8268-962X (K. Wang); 0000-0001-5099-6297 (H. Caesar); 0000-0002-7902-8537 (J. Boydens); 0000-0002-5077-4237 (D. Pissoort); 0000-0001-8297-6071 (M. Verbeke)

although a few constraints need to be addressed (Rødseth (2017)). Among the most important success factors, robust perception systems—particularly visual perception—stand out as essential.

ASVs depend on accurate and efficient vision-based object detection as a fundamental component for safe navigation and intelligent real-time decision-making. While sensors such as LiDAR, radar, and sonar enhance situational awareness, visual cameras excel in object classification and identification due to the rich information provided by high-resolution images. Object detection serves as the foundation for key ASV tasks, including object tracking and path planning. Given the real-time nature of ASVs, which require high inference speed and low computational cost, they rely on real-time object detection models. These include Convolutional Neural Network (CNN)-based models such as You Only Look Once (YOLO) (Redmon et al. (2016)), Single Shot Multibox Detector (SSD) (Liu et al. (2016)), and NanoDet-Plus (RangiLyu (2021)), as well as recent transformer-based models like the Real-Time Detection Transformer (RT-DETR) (Zhao et al. (2024)).

Despite rapid advancements and increased adoption of vision-based approaches in ASVs, ensuring the robustness and dependability of these systems remains a significant challenge, particularly since deep learning models can be severely affected by various corruptions (Rødseth & Burmeister (2015); Ahmed et al. (2023); Dodge & Karam (2016); Geirhos et al. (2018a,b)). Failures in waterborne object detection can lead directly to severe maritime accidents, such as collisions (Rødseth & Burmeister (2015)). Consequently, addressing the robustness limitations of vision-based deep learning models is imperative. To ensure safety, safety standards like ISO/IEC TR 5469 (ISO and IEC (2024)), IEC 61508 (IEC (2010)), and safety assurance frameworks including the Assurance of Machine Learning in Autonomous Systems (AMLAS) (Hawkins et al. (2021)), explicitly require a model verification phase before operational deployment. In this phase, the model must demonstrate compliance with predefined safety requirements, including dependable performance under degraded or adverse conditions. Robustness benchmarks, which offer controlled, repeatable, and scalable test scenarios to stress-test model performance against various corruptions, are therefore critical to systematically evaluate and mitigate the risks posed by model performance degradation. By identifying potential failure modes early, these benchmarks facilitate risk assessment, guide targeted improvements, and assist the compliance of safety standards for ASVs.

Current research has focused on naturally occurring data corruption robustness (Hendrycks & Dietterich (2019); Liu et al. (2024)), using benchmarks like ImageNet-C and ImageNet-P for image classification models against common corruptions and perturbations (Hendrycks & Dietterich (2019); Mu & Gilmer (2019); Barbu et al. (2019); Hendrycks et al. (2021)). Researchers have extended these efforts to autonomous driving, using benchmarks to evaluate object detection robustness (Michaelis et al. (2019)). However, these benchmarks target general image classification or autonomous driving, leaving waterborne-specific factors under-explored. Compared to road-based autonomous vehicles, ASVs operate in uniquely challenging environments characterized by factors including strong sun glare intensified by water reflection, the presence of adhered droplets on camera lenses from rain or splashing water, and complex weather conditions (e.g., fog combined with rain or frost). These unique factors, along with dynamic waterborne environments, highlight the urgent need for highly robust, vision-based object detection to ensure safe navigation in dynamic waterborne settings.

In our earlier work, we took a step toward filling this gap by assessing how real-time CNN-based object detectors perform under ImageNet-C corruptions adapted for maritime images (Wang et al. (2024)). Still, these corruptions do not capture the distinctive challenges of waterborne environments where ASVs operate, nor do they consider mixed corruptions that more accurately reflect real-world complexity. In fact, existing benchmarks rarely evaluate how well models tolerate simultaneous corruptions, an omission that can underestimate the practical vulnerability of vision-based perception systems.

Given the high stakes of ASV's safety, there is a clear need for domain-specific robustness benchmarks and verification datasets that complement safety standards and safety assurance frameworks. Beyond post-hoc verification, robustness benchmarks themselves can serve as effective data augmentation tools, improving model robustness during training, while not being used for verification meanwhile. This dual role—verification and augmentation—aligns directly with industry demands for both robust model development and a rigorous verification phase prior to deployment in safety-critical environments.

In this paper, we address these gaps by proposing the first maritime-focused robustness benchmark, which comprises three key contributions:

- **Development of a novel robustness benchmark:** We address a key concern in utilizing vision-based object detection for ASVs by introducing a comprehensive robustness benchmark tailored for visual object detection in

ASVs. It enhances the ImageNet-C benchmark with 10 novel corruption types at 5 severity levels. This results in a benchmark featuring 25 types of synthesized waterborne corruption types, including mixed corruptions—the first effort of its kind—reflecting real-world complexity often omitted in standard benchmarks. This benchmark can be integrated into the model verification phase as specified by safety standards (IEC/ISO TR 5469, and IEC 61508) and safety assurance frameworks (AMLAS). The results of the benchmark are directly actionable for industrial practice.

- **Creation of enhanced benchmark datasets:** To facilitate the robustness testing of object detection models for ASVs, we introduce three new datasets, SeaShips-C, SMD-C, and SSAVE-C. These datasets are based on existing public waterborne datasets but extensively augmented with our comprehensive corruption suite. They provide a shared testbed to rigorously verify and compare object detection models for ASVs.

- **Comprehensive evaluation of state-of-the-art models:** We conduct an extensive evaluation that incorporates additional experiments on novel corruptions and recent models (i.e. the RT-DETR family), compared with Wang et al. (2024), assessing models' preserved performance under various corruption severities. The results reveal specific vulnerabilities in investigated object detection models across various datasets, emphasizing areas where robustness improvements are needed. These insights can guide future research toward more robust detection systems for ASVs.

The remainder of the paper is organized as follows. Section 2 presents the related work. Section 3 elaborates on the developed robustness benchmark. Subsequently, Section 4 provides a comprehensive evaluation using this benchmark, presenting results and a discussion. Section 5 presents the conclusions and offers several suggestions for future research. Lastly, appendices with supplementary tables and figures are provided.

2. Related Work

2.1. Benchmarking corruption robustness

The deficiency in robustness has been widely demonstrated for deep learning models, particularly CNNs. Dodge & Karam (2016) have shown that Deep Neural Network (DNN) models for image classification are susceptible to quality distortions. According to Geirhos et al. (2018b), the performance of CNNs on object recognition deteriorates more rapidly than human performance under various types of image degradations. ImageNet-trained CNNs have been discovered to be strongly biased towards recognising textures rather than shapes, which contrasts human behavior (Geirhos et al. (2018a)). Generally, two types of robustness are investigated in literature: natural corruption robustness caused by natural distribution shifts and adversarial robustness caused by adversarial attacks. For corruption robustness, benchmarks such as ImageNet-C and ImageNet-P have been proposed to evaluate the robustness of image classification models against common corruption and perturbations by Hendrycks & Dietterich (2019). Other works have also contributed to the benchmarking and evaluation of corruption robustness, such as MNIST-C proposed by Mu & Gilmer (2019), ObjectNet by Barbu et al. (2019), and DeepFashion Remixed by Hendrycks et al. (2021). As most works focus on classification robustness, there is also a benchmark for object detection in autonomous driving being proposed by Michaelis et al. (2019) based on the corruption types in ImageNet-C (Hendrycks & Dietterich (2019)). Hendrycks et al. (2021) and Arsenos et al. (2024) investigate the transferability of improvements in corruption robustness benchmarks to real-world distribution shifts, demonstrating their practical relevance in diverse scenarios.

2.2. Towards enhanced corruption robustness

Various works focus on improving performance on data with natural distribution shifts in different ways. One approach is to restore the image by removing the corruption. Numerous studies have developed methods for denoising (Goyal et al. (2020)), deblurring (Zhang et al. (2022)), rain removal (Wang et al. (2020)), fog removal (Liang et al. (2021)), etc. However, a limitation of these algorithms is that they currently target specific types of corruption, while real-world scenarios involve diverse corruption types.

Another approach is data augmentation, where various techniques are applied to improve model robustness. Although models trained or pretrained on a specific corruption type can surpass human performance on that particular corruption type, this does not imply generalization of robustness to other corruption types (Geirhos et al. (2018b)). However, in other works, some data augmentation techniques like style transfer (Geirhos et al. (2018a)) and AugMix

(Hendrycks et al. (2019)) have been found to improve corruption robustness across multiple types of corruption. Nevertheless, robustness gains are typically not uniform across different corruption types (Yin et al. (2019)).

Other techniques, such as self-attention, pretraining, and larger models, have also been studied to improve corruption robustness, mainly for classification. In specific image classification robustness benchmarks, ResNet-152 has demonstrated superior robustness compared to its smaller counterpart, ResNet-50 (Hendrycks et al. (2021)).

2.3. Situational awareness of ASVs

Both non-vision-based and vision-based object/obstacle detection approaches have been extensively studied to ensure the safe navigation of ASVs. For non-vision-based approaches, technologies such as LiDAR, radar, sonar, the Global Navigation Satellite System (GNSS), and the Automatic Identification System (AIS) can be employed to provide navigation awareness for ASVs (Lyu et al. (2022)). LiDAR is effective for detecting objects and obstacles within a range of approximately 1 to 200 m with good precision, although its range and accuracy can diminish in adverse weather conditions (Villa et al. (2020)). In contrast, radar offers more robust performance across various weather scenarios, capable of detecting objects at greater distances (from 40 m to 72 nautical miles), particularly those with a high Radar Cross Section (RCS), which indicates the extent to which a target reflects radar signals back to the source. For current large merchant ships, radar is the most widely used equipment for detection (Lyu et al. (2022)). However, radar may not perform as well in densely populated areas or for detecting objects with a low RCS (Bloisi et al. (2016, 2015); Szpak & Tapamo (2011); Wen et al. (2021)). Sonar is mainly utilized for identifying underwater objects that exhibit distinct acoustic characteristics in Autonomous Underwater Vehicles (AUVs) (Hayes & Gough (2009)). It also shows potential for obstacle detection in ASVs according to Heidarsson & Sukhatme (2011). GNSS is used for positioning the ASV itself (Zhang et al. (2018)) and is often integrated with AIS and the Electronic Navigational Chart (ENC) to facilitate path planning. AIS, a waterborne communications system, allows vessels to share key navigational data, including identity, speed, position, course, heading, and navigational status. However, it is important to note that smaller boats, such as private yachts and fishing vessels, may lack AIS, thus limiting nearby vessels from detecting them with the AIS signal (Hesselbarth et al. (2020)).

Vision-based technologies, utilizing images and video data captured by cameras, offer distinct advantages over non-vision-based methods in ASVs and, therefore, gain increasing attention and interest in this field. These include intuitiveness and account for a rich set of data characteristics such as shape, color, and texture, which are critical for object classification, scenario analysis, and decision-making processes (Prasad et al. (2017); Bai et al. (2016); Huang et al. (2021)). Moreover, vision-based systems can detect objects that may be missed by radar or LiDAR technologies (Lyu et al. (2022)). The integration of vision-based approaches, particularly through deep learning models like YOLO, is widely acknowledged for its significant practical value in enhancing ASV navigation (Lyu et al. (2022)). Notable applications include the use of YOLOv3 for object detection across various maritime scenes (Chen et al. (2020)), a saliency-aware CNN framework for real-time, high-accuracy ship detection (Shao et al. (2019)), and a positioning system based on the YOLOv3 model for auto-docking, which has been successfully tested on a small ASV (Volden et al. (2022)). While vision-based approaches for object detection in ASVs offer significant advantages, they also face considerable challenges due to natural disturbances such as adverse weather conditions, occlusions, and varied lighting conditions, including sun glare, brightness, and contrast (Shao et al. (2018); Guo et al. (2023); Cheng et al. (2021); Wang et al. (2024)). To enhance the robustness of object detection systems, vision-based technologies can be effectively integrated with non-vision-based methods. For instance, combining image data with LiDAR or radar data has improved detection accuracy and dependability (Stanislas & Dunbabin (2018); Cheng et al. (2021); Nobis et al. (2019)). Moreover, verifying vision-based approaches in various challenging navigation scenarios enhances the trustworthiness and effectiveness of these combined systems by allowing for an informed assessment of risks associated with vision-based technologies.

2.4. Safety verification in the context of ISO/IEC standards and safety assurance frameworks

Ensuring the safety of ML components within safety-critical systems, including ASVs, requires rigorous verification processes that align with established safety standards. IEC 61508 (IEC (2010)) outlines a comprehensive functional safety lifecycle, including safety verification and validation phases following system realization, for electrical, electronic and programmable electronics safety-related systems. It is acknowledged in ISO/IEC TR 5469 (ISO and IEC (2024)) as a foundational lifecycle that can be tailored and adapted to address the specific challenges of AI systems, such as robustness to corruption. ISO/IEC 5338 (ISO and IEC (2023)) further defines the lifecycle for AI systems as a structured series of processes, explicitly including verification as a core process. In parallel, the AMLAS

framework (Hawkins et al. (2021)) provides a structured approach for building safety cases around ML components. Within AMLAS, the model verification stage is a critical stage, requiring the verification of ML system behavior against defined safety requirements.

While formal verification methods aim to mathematically guarantee system correctness, they often prove infeasible for complex perception systems (Huang et al. (2020)). Test-based verification, by contrast, offers a more scalable and practical alternative for assessing risk under controlled conditions. Our benchmark, by exposing real-time object detection models to a suite of corrupted inputs with varying severity, contributes to this risk-driven verification step. The results offer quantitative evidence that can support risk assessment, guide safety case development, and inform threshold-based acceptance criteria. Furthermore, it facilitates the scoping of safety goals, and informs human-in-the-loop oversight mechanisms by identifying prominent risk factors, which are practices that are emphasized in the AI Act, a regulation laying down harmonized rules on AI systems proposed by European Commission (2021), and supported by the above standards and frameworks.

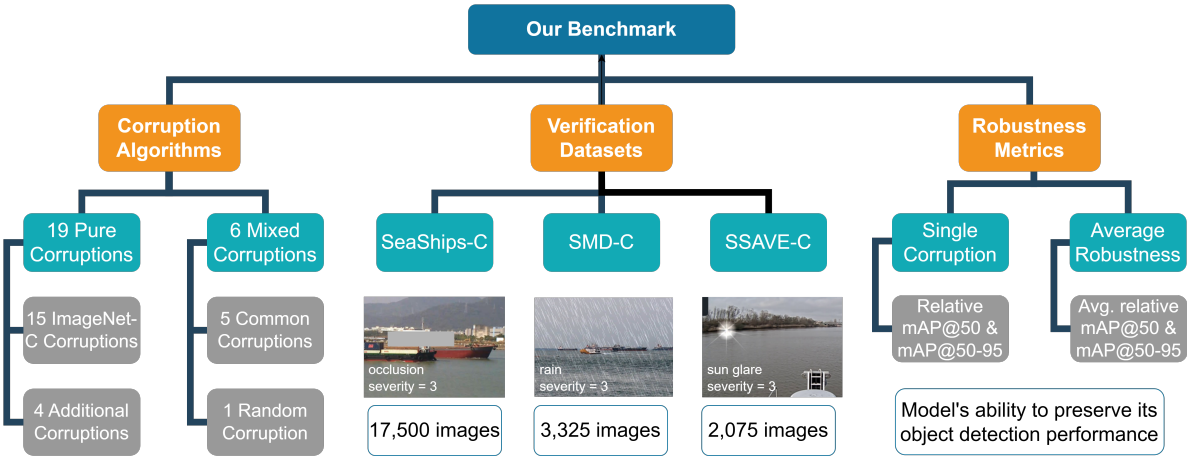


Figure 1: Visualization of the proposed benchmark. This benchmark comprises three key components: corruption algorithms, benchmark datasets, and robustness metrics.

3. Methodology

To assess the robustness of real-time object detection approaches in the context of ASVs, we propose a benchmark comprising three key components: corruption algorithms, benchmark datasets for robustness verification, and robustness evaluation metrics, as illustrated in Figure 1. The corruption algorithms simulate various realistic waterborne conditions to generate the benchmark datasets for robustness verification. These datasets are then used to evaluate the performance of object detection models using specific robustness metrics. In this section, we detail the methodology for generating corruptions, describe the creation of the robustness benchmark datasets, and outline the evaluation metrics utilized.

3.1. Corruption

Corruption refers to distortions or degradations in image data caused by noise, weather conditions, lighting changes, occlusions, or sensor malfunctions, which can adversely impact the performance of vision-based models. In existing literature, there are two distinct technical routes to introduce corruption in datasets: (1) using corrupted images collected from real-world scenarios or (2) using synthesized corruption. Each has its unique strengths and limitations. For synthesized corruption-based benchmarks, creating a test dataset with various severity levels of abundant corrupted images is comparatively easier, more cost-effective, and more practical than collecting corrupted images from a wide range of real-world scenarios. Additionally, this approach allows for the reuse of object detection labels, making it a reasonable choice for verifying known unknowns. There might be a pertinent concern regarding the extent to which offline evaluations conducted on corruption robustness benchmarks accurately mirror a model's robustness in real-world scenarios. However, improvements in artificial robustness benchmarks are already found to be transferable to real-world distribution shifts in various works (Hendrycks et al. (2021); Arsenos et al. (2024)).

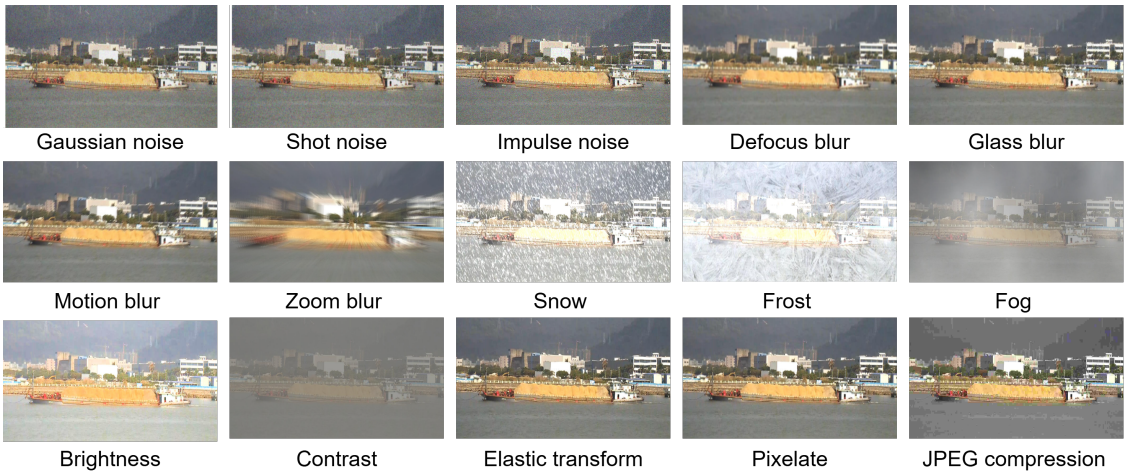


Figure 2: An example image from the SeaShips dataset under the 15 ImageNet-C corruption types at severity 3.

In this work, we introduce a robustness benchmark that encompasses 25 synthesized corruption types. This benchmark extends beyond the 15 corruption types identified in ImageNet-C by incorporating 10 specific types for ASVs. The additional corruptions include 4 pure corruptions typical in ASV environments (rain, raindrops and water droplets adhered to camera lens or bridge windows, sun glare, and occlusion), five common mixed corruptions (rain with fog, frost with fog, raindrops with contrast, Gaussian noise with contrast, sun glare with motion blur), and one random mixed corruption generated by randomly selecting two out of the 19 defined pure corruption types.

Each corruption is implemented across five severity levels, ranging from negligible (level 1) to severe (level 5). This five-level scheme is consistent with the standardized structure established by ImageNet-C, and widely adopted across other robustness benchmarks such as CIFAR-10-C (Hendrycks & Dietterich (2019)), Cityscapes-C (Michaelis et al. (2019)), KITTI-C, nuScenes-C (Dong et al. (2023)) and ModelNet-C (Ren et al. (2022)). These benchmarks, including ours, define severity levels through parameterized adjustments that progressively degrade image quality or increase task difficulty. For example, in ImageNet-C, Gaussian noise severity is quantified via increasing standard deviations, and motion blur is defined by growing kernel sizes. Such quantitative definitions ensure that severity levels are both measurable and controllable. For the 15 corruption types adopted directly from ImageNet-C, we retain the original parameter settings at each severity level, ensuring alignment and comparability with the broader robustness literature. For the newly introduced corruption types, we define severity levels based on empirical calibration, perceptual gradation, and operational relevance in real-world maritime conditions. Parameters such as rain streak intensity, droplet density, occlusion area, sun glare radius are systematically varied to create five distinct levels that are perceptually distinguishable and practically meaningful in ASV contexts, and monotonically increasing in difficulty.

Notably, our corruption methodology operates entirely from the data side, independent of model architecture or complexity, and scales linearly with the size of the evaluation dataset. While the full benchmark includes 25 corruption types applied across five severity levels (125 variations per image), the corruption generation process is parallelizable, enabling efficient execution. In scenarios where storage or compute becomes a bottleneck, corruptions can be generated on-the-fly during evaluation, or the benchmark can be applied selectively to representative slices of the test dataset. The corruption generation pipeline is designed to be scalable, controllable, and adaptable to varying practical needs.

The rest of Section 3.1 provides a detailed overview of the 25 corruption types. Section 3.1.1 summarizes the 15 existing corruption types from ImageNet-C. The four newly added pure corruption types are then described: rain (Section 3.1.2), raindrops and water droplets adhered to camera lenses or bridge windows (Section 3.1.3), sun glare (Section 3.1.4), and occlusion (Section 3.1.5), with severity level quantification details presented in Appendix A. Lastly, the five common mixed corruption types and one random mixed corruption type are collectively discussed in Section 3.1.6.

3.1.1. 15 corruption types in ImageNet-C

The 15 corruption types in ImageNet-C can further be classified into four categories, related to lighting conditions (Gaussian noise, shot noise, brightness, and contrast), related to adverse weather conditions (glass blur, snow, frost, fog), related to the camera movement (defocus blur, motion blur, and zoom blur), and related to digital processing or errors that occur during digital processing (impulse noise, elastic transform, pixelation, and JPEG compression), making them also common to ASVs. Figure 2 depicts an example from the SeaShips dataset corrupted with the 15 corruption types.

3.1.2. Rain

Rain significantly affects the visibility of the surrounding environment for ASVs (Lyu et al. (2022)), making it a crucial corruption type to consider in robustness benchmarks. One challenge is the scarcity of image data collected under adverse weather conditions for ASVs (Volden et al. (2022)), such as heavy rain. To address this, various studies have developed methods to synthesize rain, primarily motivated by the need for autonomous driving verification and improvement (Wei et al. (2021); Halder et al. (2019); Joaedl (2019)). Despite extensive research in autonomous driving, verifying synthesized rain effects in ASVs has been comparatively neglected. To bridge this gap, we adapt the methodology from Joaedl (2019), to incorporate rain corruption into our benchmark. We introduce a rain mask to the original images, composed of slender rectangles that simulate rain streaks. These streaks' slope, length, width, density, blur, transparency, and intensity are adjusted to mirror real-world conditions. Furthermore, we establish five different severity levels of rain corruption with variations in the number, length, and intensity of rain streaks, aligning with the standards set in ImageNet-C to ensure a consistent and comprehensive test benchmark across various corruption types. This corruption type is visualized together with other newly-added corruption types in Figure 3.

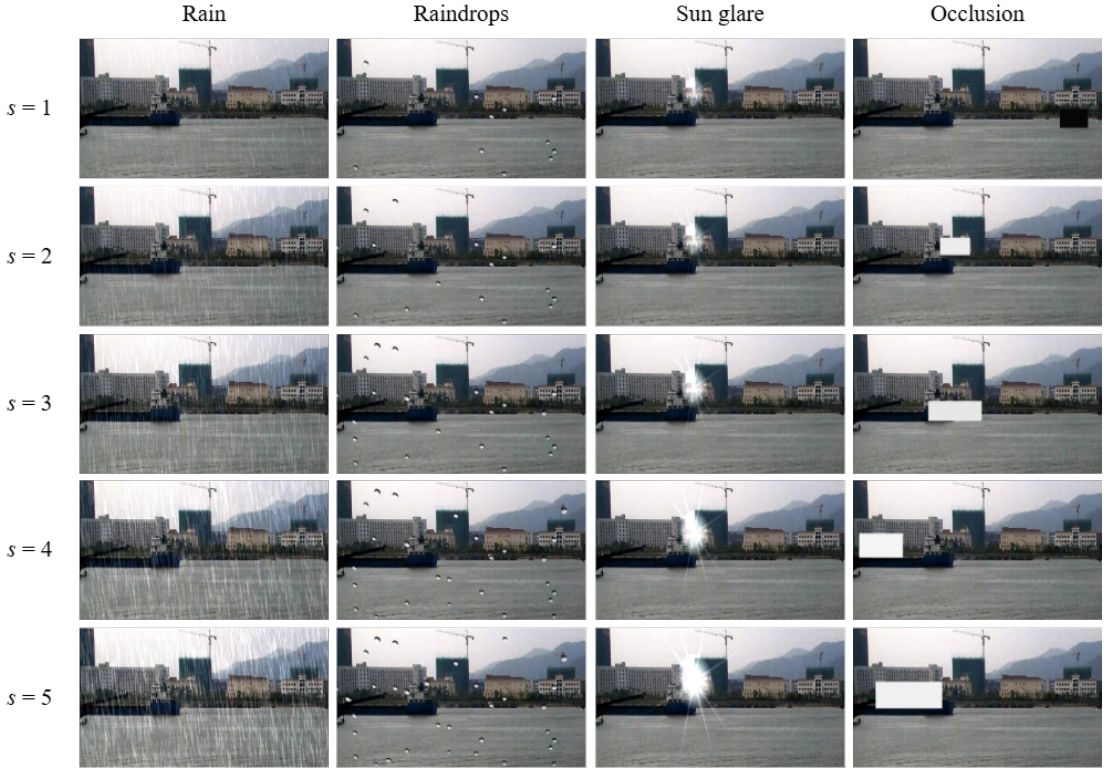


Figure 3: An image from SeaShips under 5 severity levels s of the newly-added pure corruption types.

3.1.3. Raindrops and water droplets adhered to camera lens or bridge windows

Adhered raindrops and water droplets on camera lenses, windshields, or bridge windows present another significant type of corruption for ASVs, distinct from rain streaks introduced in Section 3.1.2. Although both types of corruption

can occur during rainy conditions, raindrops possess unique characteristics and a wider variety, necessitating their separate consideration in vision-based deep learning models.

To synthesize raindrops corruption, we adapt a physics-based method proposed by Hao et al. (2019) that considers water dynamics, geometry, and photometry. In this method, a virtual imaging system is modeled, with raindrops adhering to a transparent interface located along the optical axis, with a background plane located further behind. Raindrops are randomly distributed on the glass surface and modeled as spherical caps. For each pixel in the synthetic image, the corresponding light ray is traced: if no raindrop is intersected, the background pixel value is retained. If the ray passes through a raindrop, its path is refracted according to water-air boundary conditions, and the pixel is assigned the value from the ray's intersection with the background plane. In cases where total internal reflection occurs (commonly at droplet boundaries), pixels are set to black, simulating the "dark band" effect observed in real-world imagery. To further enhance realism, a disk blur kernel (typically 7–20 pixels in diameter) is applied to raindrop regions, mimicking the soft appearance of raindrops captured by cameras due to limited depth of field. Building upon this framework, we implement a Python-based variant to synthesize raindrop corruptions at five severity levels, determined by varying the visible density of raindrops. We further refine the procedure to generate all five severities from a single clean image in batch, enabling stable raindrop positioning across levels and reducing computational overhead. These optimizations cut processing time by 90%, facilitating efficient deployment in industrial settings.

3.1.4. Sun glare

Sun glare is a common phenomenon in the daily navigation of ASVs, where intense sunlight near the horizon can severely obstruct a camera's view. This corruption poses significant challenges for vision-based object detection models. While sun glare also commonly impairs cameras in other applications, such as autonomous driving, it is even more pronounced in ASVs due to the reflective water surfaces that intensify its effects. Despite its prominence in ASV environments, sun glare has long been neglected in test verification processes for these systems, a gap in the research that is critical to address.

Limited studies have explored the generation and removal of sun glare in land transportation. For instance, Chen et al. (2021) developed a synthesized image dataset of sun-glared license plates for electronic toll collection systems by manually adding glare masks to clean images using Photoshop. However, this method is unsuitable for our robustness benchmark, which requires automatically synthesizing corrupted images from random clean ones.

Inspired by Wu & Pradalier (2019), which treats sun glare as local illumination changes, we develop a method to synthesize sun glare by adjusting the local illumination in an image. To simulate the sun glare corruption effect, we implement a multi-component masking strategy that approximates both the central glow and radiating spikes commonly observed in real-world glare. First, we determine the glare center by selecting a point along the upper boundary of the target object's bounding box. This placement mimics the typical positioning of the sun relative to objects in natural scenes. Around this center, we construct a set of concentric circular masks, with radii increasing in fixed steps of 3 pixels, up to an outermost radius determined by the severity level. To simulate the brightening associated with central glare, each circular mask is assigned an increasing brightness level toward the center. The cumulative overlay of these masks forms a stepped radial gradient that approximates a smooth, centralized glow.

To enhance realism with directional light artifacts, we augment the circular glow using four slim, diamond-shaped masks. These masks are centered on the glare point and randomly rotated to simulate the angular variability seen in natural sun glare patterns. Within each diamond mask, we apply a brightness gradient similar to the radial one used in the circular masks, increasing pixel intensities from the outer edges inward. This combination of concentric radial glow and angular spikes allows our method to effectively mimic the characteristic structure of sun glare, while enabling controlled variation through the severity levels. A comparison of our method with the two existing methods is visualized in Figure 9 in Appendix B.

3.1.5. Occlusion

Occlusion is also a frequent challenge in ASVs' daily navigation scenarios. Inevitably, various elements such as ships, vessels, obstacles, navigation marks, and waves, may obscure objects from the camera's perspective on an ASV. To effectively address this, object detection models must be capable of handling occlusions. In our approach, we synthesize occlusion by randomly masking a square region of a clean image. To determine the severity levels of occlusion, we consider the area of the masked region and its positional relationship to the object. Additionally, the area of the masked region is proportionate to the object it intends to occlude; thus, a small area is not used to obscure a large object, nor is a very large area used to obscure a small object.

3.1.6. Mixed corruption types

In addition to newly introduced pure corruption types, our study also incorporates mixed corruptions composed of two distinct pure types. This enhancement reflects the complex and diverse scenarios encountered in real-world navigation. For instance, phenomena such as rain and fog often occur simultaneously in autonomous waterborne operations.

Since certain mixed corruption types are particularly common in real-world conditions, their verification requires special attention. We include five common mixed corruption types specifically relevant to waterborne applications, thereby enhancing the benchmark's practical focus and depth. Importantly, as our benchmark serves as a standardized tool, it remains flexible, allowing users to generate any desired mixed corruption combinations from the pure types. This flexibility enables users to extend or replace the five predefined common mixed corruptions, facilitating customized testing based on specific operational needs. The five common mixed corruptions are defined as:



(a) Rain+fog (b) Frost+fog (c) Raindrops+contrast

(d) Gaussian noise+contrast (e) Sun glare+motion blur

Figure 4: Five common mixed corruption types added on an image from SeaShips, at severity level 3.

- Rain with Fog: Frequently co-occurring in waterborne settings, presenting significant visual challenges.
- Frost with Fog: Typical in cold weather conditions, these conditions need to be verified for year-round operations in waterborne contexts.
- Raindrops with Contrast: This phenomenon occurs when raindrops adhere to the camera lens, windshield, or bridge windows of ASVs, distorting and scattering light. It is often accompanied by reduced contrast in overcast or stormy conditions.
- Gaussian Noise with Contrast: Both commonly appear in low-light conditions, making their combined effect a critical test case.
- Sun Glare with Motion Blur: Common during navigation in sunny conditions.

Each mixed corruption type is defined across five severity levels (visualized in Figure 4), corresponding to the severity of the based pure corruptions. For example, the first severity level of a mixed corruption is synthesized using the first severity level of both contributing pure corruption types. This systematic approach extends to other severity levels, ensuring a consistent and scaled approach to severity across the benchmark.

Given the 19 identified pure corruption types, there exists a vast space of potential mixed corruption combinations. Comprehensive safety verification benefits from exploring this diversity. In addition to the 5 common mixed corruptions, we introduce a method for generating a random mixed corruption type to ensure broad representation within our benchmark. For each image, this method randomly selects two of the 19 pure corruption types and applies them at matching severity levels. These two selected types are then fixed across all five severity levels for this image, ensuring uniformity and intuitive testing conditions.

By incorporating pure, specific common mixed and random mixed corruption scenarios, our benchmark aims to comprehensively assess an ASV's visual detection systems under a broad spectrum of operational challenges.

3.2. Benchmark datasets

To enhance dataset diversity, we base our research on three datasets (SeaShips, SMD, and SSAVE) with various camera settings (onboard, onshore, and on a drone) and distinct waterborne environments (maritime and inland waterway). To assess robustness, we apply the 25 corruption types from the proposed benchmark to these clean test datasets, resulting in three corrupted datasets that can be used for test-based model verification: SeaShips-C, SMD-C, and SSAVE-C. This subsection further elaborates on the characteristics of SeaShips, SMD, and SSAVE datasets, as well as the three corrupted verification datasets derived from them.

3.2.1. SeaShips, SMD, and SSAVE

Seaships is a large-scale and well-annotated maritime dataset specifically created for ship object detection (Shao et al. (2018)). The online released version of SeaShips includes 7000 images in total, consisting of six common ship types: ore carrier, passenger ship, container ship, bulk cargo carrier, general cargo ship, and fishing boat. This dataset is obtained by cameras deployed in different locations onshore. The dataset also includes ship images of various sizes, under different lighting conditions and from different viewpoints.

Singapore Maritime Dataset (SMD) is another widely used maritime object detection and object tracking dataset (Prasad et al. (2017)). This dataset provides 51 Visual-Optical (VIS) and 30 Near Infrared (NIR) videos taken from Singapore waters with 10 classes of annotations. In this paper, we only consider the VIS videos. Of these 51 VIS videos, 40 were captured with onshore cameras, while 11 were recorded using onboard cameras. Different illumination conditions, such as fog, daylight, and dark/twilight, are also covered in this dataset. Due to severe class imbalance, where several classes contain very few instances, we frame the object detection task as a two-class problem: object vs. background. This simplification is consistent with common practice in prior work, as noted in Moosbauer et al. (2019).

Shared Situational Awareness between Vessels (SSAVE) is a small-scale dataset that contains 827 representative images gathered in various Belgian waterways (Lahouli et al. (2022)). Unlike SeaShips and SMD, SSAVE is an inland waterway dataset with specific backgrounds and objects, such as various types of markers on the water surface. The images in SSAVE are captured either onboard from a navigating barge or from a camera mounted on a drone.

The dataset splits for all three datasets follow the configuration established in Wang et al. (2024).

3.2.2. SeaShips-C, SMD-C, and SSAVE-C

SeaShips-C, SMD-C, and SSAVE-C are three robustness benchmark datasets developed from their respective base datasets (SeaShips, SMD, and SSAVE) by incorporating the 25 corruption types specified in our robustness benchmark. SeaShips-C contains a total of 17,500 images, SMD-C comprises 3,325 images, and SSAVE-C includes 2,075 images. Each dataset is presented with an example image in Figure 1. It is important to clarify that these datasets are designated for robustness evaluation and model verification; they are not to be used for training purposes in the meanwhile.

The robustness benchmark datasets were constructed by applying the 25 corruption algorithms (illustrated in Figure 1) to the clean test datasets. Each benchmark dataset has an equal distribution of images for each of the 25 corruption types, ensuring balanced representation across all corruptions.

3.3. Evaluation metrics

Generally, the performance of object detection models is evaluated with two standardized performance metrics: mAP^{50} and mAP, where mAP^{50} denotes the mean Average Precision computed at 50% Intersection over Union (IoU) and mAP denotes the mean Average Precision averaged over 10 IoU values ranging from 0.50 to 0.95 with a step size of 0.05 (.50:.05:.95).

As a mathematical definition of robustness is still largely missing in the literature, similar to Michaelis et al. (2019), we consider the robustness here as a model's ability to preserve its model performance under natural corruption. We also assume that a model's ability to preserve its model performance under natural corruption may vary depending on the type of corruption. Based on this assumption, we propose two metrics to evaluate model robustness under a specific corruption type. These metrics are derived from the two standardized performance metrics, mAP^{50} and mAP.

Based on mAP^{50}, we define a robustness metric on a corruption type as the relative mean average precision at 50% IoU on the corruption type c:

$$\text{rmAP}^{50}_{c} = \frac{1}{S} \frac{\sum_{s=1}^{S} \text{mAP}^{50}_{s,c}}{\text{mAP}^{50}} \times 100\%, \tag{1}$$

where s denotes the level of severity of this corruption, S is the total number of severity levels (S equals 5 in this work), $\mathrm{mAP}^{50}_{s,c}$ stands for the mean Average Precision at IoU 50% obtained under corruption type c at severity level s, and mAP^{50} stands for the measure obtained on the clean test dataset without synthesized corruption. rmAP^{50}_{c} then is the preserved percentage of mAP^{50} performance under corruption type c averaged across all levels of severity.

Similarly, based on mAP, we define another robustness metric on a corruption type as the relative mean average precision which averages over IoUs between 50% and 95% on corruption type c:

$$\mathrm{rmAP}_c = \frac{1}{S}\frac{\sum_{s=1}^{S}\mathrm{mAP}_{s,c}}{\mathrm{mAP}} \times 100\%, \tag{2}$$

where $\mathrm{mAP}_{s,c}$ stands for the mean Average Precision averaged over IoUs under corruption type c at severity level s, and mAP stands for the measure on the clean test dataset without any synthesized corruption. Similar to rmAP^{50}_{c}, rmAP_c denotes the preserved percentage of mAP performance when subjected to corruption type c, averaged across all levels of severity.

Notably, for both rmAP^{50}_{c} and rmAP_c, higher values indicate greater model robustness against the specified corruption type, as they reflect the extent to which the model's original performance is maintained under corruption.

We also define two average robustness metrics to represent the robustness averaged over various corruption types based on rmAP^{50}_{c} and rmAP_c respectively, which are calculated as follows:

$$\mathrm{rmAP}^{50}_{\mathrm{avg}} = \frac{1}{C}\sum_{c=1}^{C}\mathrm{rmAP}^{50}_{c}, \tag{3}$$

$$\mathrm{rmAP}_{\mathrm{avg}} = \frac{1}{C}\sum_{c=1}^{C}\mathrm{rmAP}_{c}, \tag{4}$$

where C represents the total number of corruption types. $\mathrm{rmAP}^{50}_{\mathrm{avg}}$ and $\mathrm{rmAP}_{\mathrm{avg}}$ represent a model's average ability to preserve its performance over various defined corruption types. Similar to rmAP^{50}_{c} and rmAP_c, higher values for $\mathrm{rmAP}^{50}_{\mathrm{avg}}$ and $\mathrm{rmAP}_{\mathrm{avg}}$ indicate greater robustness, reflecting the model's overall resilience across multiple corruption scenarios.

4. Results and Discussion

The primary objective of this study is to introduce a comprehensive robustness benchmark tailored for verifying object detection models in the challenging waterborne environment encountered by ASVs. This benchmark includes 25 distinct corruption types across five severity levels, to rigorously evaluate model performance under various corruption scenarios.

To validate the efficacy and applicability of our robustness benchmark, this section addresses five key research questions (RQs), each corresponding to a dedicated subsection:

- **RQ1:** *How do models perform across various state-of-the-art real-time object detection architectures, and across multiple waterborne datasets?*

  Section 4.1 assesses and discusses the performance from the two perspectives.

- **RQ2:** *Which corruption types most significantly degrade model performance on waterborne environments, and which have a minimal impact?*

  Section 4.2 analyzes the impact of corruption type on robustness.

- **RQ3:** *How does the model size affect the robustness?*

  Section 4.3 investigates the correlation between model size and robustness on benchmark datasets.

- **RQ4:** *How does the presence of mixed corruption differ from pure corruption in terms of robustness?*

  Section 4.4 compares model performance under pure versus mixed corruption scenarios across different severity levels.

- **RQ5:** *How do the proposed robustness metrics correlate with real-world failure modes and risks?*

  Section 4.5 links benchmark metrics to real-world detection failures, emphasizing the benchmark's relevance to ASV safety.

Each subsection presents and analyzes experimental results and reinforces how these findings validate the robustness benchmark's effectiveness and utility as a standardized tool for model verification in ASVs.

4.1. Benchmark validation across different architectures and datasets

To validate our proposed robustness benchmark, we employed a two-stage experimental methodology to evaluate the performance of various object detection architectures across multiple waterborne datasets.

- Stage 1: Evaluation across different architectures on a single dataset. In the first stage, we assessed four state-of-the-art real-time object detection models, YOLOv8, SSD, NanoDet-Plus, and RT-DETR, on SeaShips-C which includes most images among the datasets considered. Each model was evaluated in different sizes and with varying network backbones to capture a broad range of architectural performances. This approach aimed to identify universal weaknesses and performance trends inherent to each architecture when subjected to the same set of corruption scenarios.

- Stage 2: Evaluation of a single model family across different datasets. The second stage extended the evaluation to YOLOv8 models across three benchmark datasets: SeaShips-C, SMD-C, and SSAVE-C, as introduced in Section 3. By maintaining a consistent model family while varying the datasets, this stage aimed to uncover dataset-specific robustness factors and assess how different data environments influence model performance under identical corruption conditions.

For the training process, each model was consistently trained on the clean datasets using default settings, including hyperparameters and data augmentation methods as specified in their respective sources. Specifically, YOLOv8, SSD, and NanoDet-Plus retain the exact training configurations established in our previous work (Wang et al. (2024)), which adopted default settings from their original references (Jocher et al. (2023); Li (2018); RangiLyu (2021)). The newly evaluated RT-DETR follows the default training settings provided by its official paper (Zhao et al. (2024)). For the evaluation process, models are evaluated on both the clean and the corrupted test datasets, including 25 types of corruption at all five severity levels. Every model undergoes a rigorous training and evaluation process repeated five times to ensure statistical reliability, with performance metrics averaged across runs to provide consolidated insights.

We present the results of Stage 1 and Stage 2 in Tables 1 and 2, respectively. In these tables, the robustness metrics—$\mathrm{rmAP}^{50}_{\mathrm{avg}}$ and $\mathrm{rmAP}_{\mathrm{avg}}$, which represent the models' average relative performance across all 25 corruptions—are the primary focus. To provide additional context, we also report the absolute object detection performance on the clean test datasets using mAP^{50} and mAP. Note that the robustness metrics are not directly comparable to the clean performance scores, as the former are calculated relative to the corresponding clean performance (see Equations (1)–(4)).

Detailed robustness scores for each corruption, $\mathrm{rmAP}^{50}c$, are provided in Tables 4 and 5 in Appendix C for both stages. Notably, $\mathrm{rmAP}_{\mathrm{avg}}$ and rmAP_c scores, yields findings highly analogous to those of $\mathrm{rmAP}^{50}_{\mathrm{avg}}$ and rmAP^{50}_c, as presented in Figure 5. Therefore, to maintain clarity and brevity, only the detailed rmAP^{50}_c scores are presented in Appendix C and subsequent figures.

From Table 1, we observe that on the clean SeaShips dataset, both CNN-based YOLOv8 models and transformer-based RT-DETR models deliver strong performance: RT-DETR achieves the highest mAP^{50}, while YOLOv8 generally attains higher mAP. In terms of robustness, both YOLOv8 and RT-DETR maintain higher relative performance under corruption, whereas SSD is notably vulnerable to certain corruptions such as adverse weather and occlusion (see Appendix C). Although SSD underperforms on the clean dataset, it demonstrates the highest inference speed. NanoDet-Plus, despite offering the shortest training time per epoch and being the most lightweight model, sacrifices robustness, highlighting a trade-off between efficiency and robustness. Furthermore, due to the highly customized implementation of each model, both the training time per epoch and inference speed comparisons presented in Table 1 may be biased.

For object detection performance across clean benchmark datasets, YOLOv8 models demonstrate high detection accuracy on the clean SeaShips and SMD datasets. In contrast, performance on the clean SSAVE dataset is comparatively lower. This can be attributed to the inherent challenges of SSAVE, which contains a diverse range of object classes—including ships, barges, cutters, various water surface markers, and other obstacles—and many

Table 1
Average robustness results ($rmAP^{50}_{avg}$ (%) and $rmAP_{avg}$ (%)), clean detection results (mAP^{50} and mAP), training time per epoch, and inference speed (frames per second, FPS) on the SeaShips-C/SeaShips dataset for YOLOv8, SSD, NanoDet-Plus, and RT-DETR models. The epoch time and inference speed were measured on NVIDIA Tesla P100 GPUs (one GPU per experiment).

Model Family	Model	$rmAP^{50}_{avg}$	$rmAP_{avg}$	mAP^{50}	mAP	epoch time (min:sec)	inference speed (FPS)
YOLOv8	n	79.1±0.7	71.8±0.8	99.3±0.1	84.9±0.4	1:16	153.0
	s	80.9±1.0	74.0±1.0	99.3±0.1	86.3±0.1	1:34	132.8
	m	83.7±1.0	77.5±1.1	99.2±0.1	86.2±0.2	3:18	94.7
	l	84.7±1.3	78.5±1.3	99.2±0.1	**86.6±0.3**	4:58	73.3
	x	**85.4±1.2**	**79.4±1.2**	99.3±0.0	86.4±0.4	8:13	55.7
SSD	-R18	73.7±1.4	65.4±1.4	95.2±0.7	68.3±0.8	2:23	**225.8**
	-R34	75.6±1.7	67.4±2.0	94.9±0.9	70.9±0.5	3:10	184.7
	-R50	76.2±1.7	67.8±1.7	95.9±1.3	70.6±0.8	3:22	136.7
NanoDet-Plus	-m	72.4±1.7	63.9±1.9	98.8±0.1	77.8±0.4	**0:58**	180.1
	-m-1.5x	73.8±1.2	65.7±1.2	99.2±0.1	79.1±0.4	1:13	143.7
RT-DETR	-R18	80.5±1.3	72.3±1.2	99.7±0.1	85.0±0.3	4:47	61.5
	-R34	82.4±1.3	75.2±1.2	99.6±0.1	85.1±0.1	7:03	50.0
	-R50	84.6±1.1	76.8±1.2	**99.8±0.1**	85.4±0.2	9:18	33.3

Table 2
Average robustness results ($rmAP^{50}_{avg}$ (%) and $rmAP_{avg}$ (%)) and clean detection results (mAP^{50} and mAP) of YOLOv8 models on the SMD-C/SMD and SSAVE-C/SSAVE datasets.

Model Family	Model	SMD-C/SMD				SSAVE-C/SSAVE			
		$rmAP^{50}_{avg}$	$rmAP_{avg}$	mAP^{50}	mAP	$rmAP^{50}_{avg}$	$rmAP_{avg}$	mAP^{50}	mAP
YOLOv8	n	71.3±1.1	59.6±1.0	98.5±0.1	83.1±0.3	74.6±1.8	69.3±1.7	79.9±1.2	57.9±1.7
	s	77.2±1.6	64.8±1.2	**99.1±0.2**	86.4±0.3	78.5±1.4	73.4±1.4	82.3±0.7	62.4±0.7
	m	79.7±1.5	67.7±1.3	99.0±0.2	87.7±0.2	82.9±1.4	77.2±1.4	82.6±0.7	64.4±0.5
	l	80.8±1.4	69.6±1.2	99.0±0.1	88.1±0.3	83.9±1.5	78.5±1.3	**83.1±0.9**	**64.8±0.7**
	x	**81.7±1.0**	**70.9±1.1**	99.0±0.1	**88.2±0.2**	**84.6±1.9**	**79.7±1.7**	82.8±0.4	64.6±1.0

small objects that are more challenging to detect. In term of robustness, SeaShips-C serves as an effective baseline due to its extensive and diverse image collection, which contains the largest number of images among the datasets. This abundance likely enhances model generalization and stability, resulting in higher average robustness scores. The smaller SMD-C and SSAVE-C datasets reveal vulnerabilities not observed with SeaShips-C, with SMD-C exhibiting much lower robustness scores. Notably, while average robustness scores on SMD-C are generally lower than those on SSAVE-C, an exception emerges for certain corruption types such as defocus blur, motion blur, and fog (see Appendix C). Because the SMD dataset includes real-world images affected by these specific corruptions, the trained model demonstrates higher robustness scores on these particular corruption types in SMD-C compared to SSAVE-C. This improvement likely arises from the models' prior exposure during training, enabling them to better recognize and adapt to patterns associated with these corruption types.

The experiments validate our robustness benchmark by demonstrating its capability to differentiate model robustness across various architectures and datasets.

4.2. Impact of different corruption types

Figure 5 illustrates that models demonstrate varying degrees of susceptibility to different types of corruption. While some corruption types result in *mild* performance degradation (less than 10%), others lead to *moderate* (10-30%) or even *severe* reductions in performance (greater than 30%), significantly impairing model dependability. The classification of performance drop into mild, moderate, and severe categories can further be associated with the risks posed by different

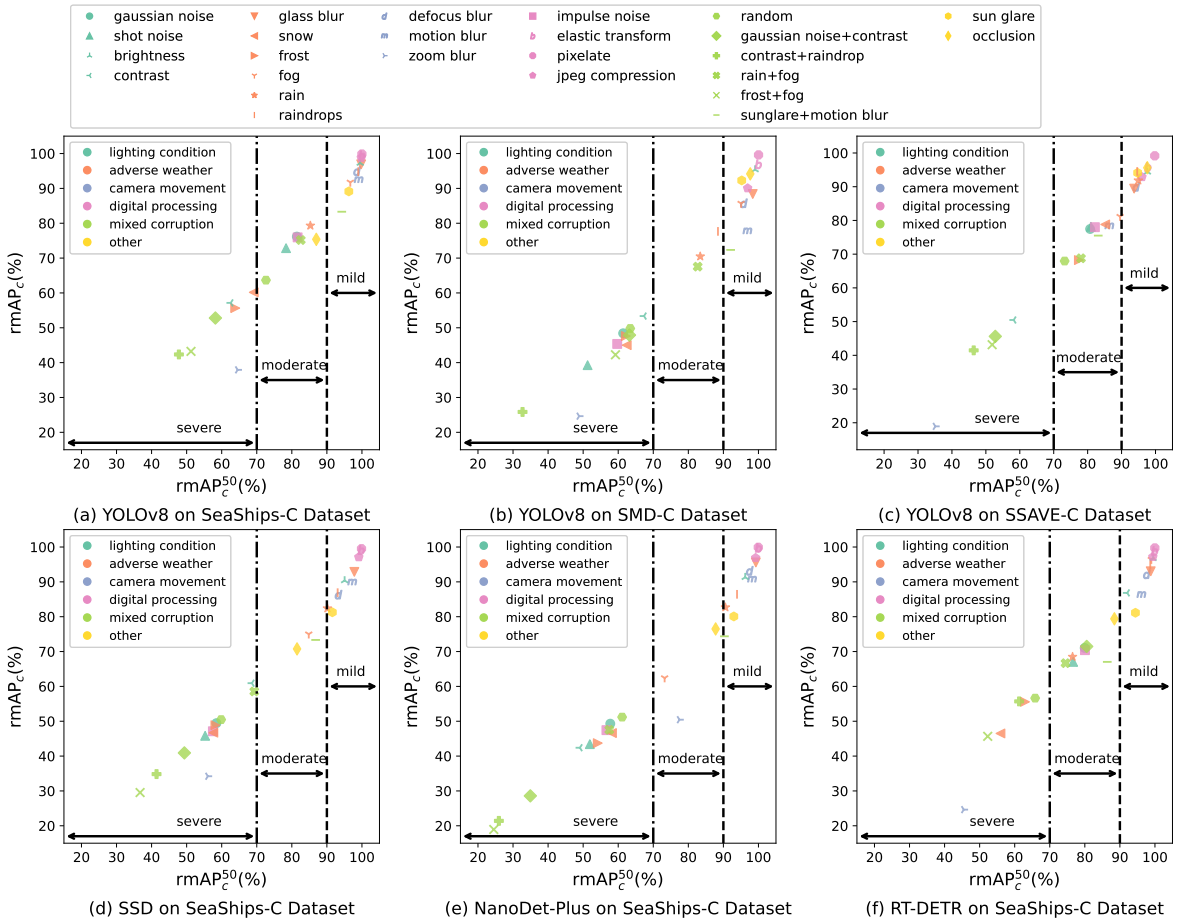
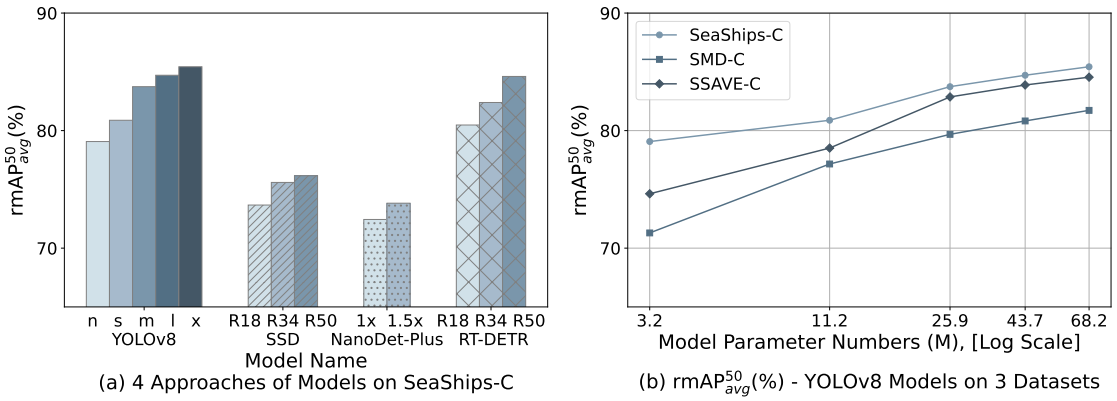


Figure 5: Visualization of the model robustness rmAP_c^{50} and rmAP_c on 25 corruption types. The results are averaged over different sizes of models in the same model family. The 25 corruption types (denoted by distinct symbols) are categorized into six corruption groups, denoted in different colors for better distinction.

corruption types. The frequent occurrence of moderate and, in particular, severe reductions underscores that current state-of-the-art real-time object detection models lack consistent robustness against certain corruption types.

To comprehensively understand the impact of different corruption types on object detection models, we categorize the 25 types of corruption into six distinct groups based on their underlying causes in real-world waterborne operations, as follows:

- Lighting condition: Gaussian noise, shot noise, brightness, and contrast. These corruption types are caused by unusual lighting conditions. For example, Gaussian noise can occur in low-light environments, while shot noise results from the discrete nature of light.
- Adverse weather: glass blur, snow, frost, fog, rain, and raindrops. These corruption types can be caused by adverse weather in the real-world. Specifically, glass blur may appear due to frosted glass windows.
- Camera movement: defocus blur, motion blur, and zoom blur. These corruption types are caused by the camera's movement, resulting in various forms of blur.



(a) 4 Approaches of Models on SeaShips-C

(b) rmAP$^{50}_{avg}$(%) - YOLOv8 Models on 3 Datasets

Figure 6: The average robustness of all 25 corruption types increases with the model size. (a) shows the results from the first stage, and (b) shows the results from the second stage.

- Digital processing: impulse noise, elastic transform, pixelate, and JPEG compression. Corruption occurs during digital image processing, affecting image quality and introducing distortions.
- Mixed corruption: Gaussian noise with contrast, contrast with raindrops, rain with fog, frost with fog, sun glare with motion blur, and random mixed corruption. These are combinations of two corruption types, creating complex corruption scenarios that mimic real-world challenges where multiple factors coexist.
- Other: Sun glare and occlusion. These corruption types exist in the real-world but do not neatly fit into the above categories.

The extent of susceptibility to specific corruption types generally follows a consistent pattern across various models and datasets. For instance, mild performance drops are consistently observed across models and datasets with corruption caused by digital processing (elastic transform, pixelation, JPEG compression), brightness adjustments, defocus blur, and glass blur. These corruptions introduce distortions that models can partially compensate for, resulting in manageable performance declines.

Conversely, severe performance drops are commonly seen with mixed corruption (contrast with raindrops, frost with fog, and Gaussian noise with contrast, and random), adverse weather (snow and frost), and lighting condition (Gaussian noise, shot noise, and contrast). These corruptions overwhelm the models' ability to detect and classify objects accurately. Specifically, the compounded effects of mixed corruption make it exceedingly difficult for models to maintain high detection accuracy, highlighting significant vulnerabilities in current model architectures.

Despite the identification of universal weaknesses, we also need to note the variability across corruption types for different models (e.g. SSD and NanoDet-Plus exhibit a higher number of severe-impact corruptions) and datasets (e.g., lighting condition-related corruptions in SMD-C and camera movement-related corruptions in SSAVE-C). These findings confirm that our benchmark serves as a standardized tool for evaluating and verifying the robustness of object detection models in autonomous waterborne operations. The varying impact of corruption types suggests that enhancing model robustness requires targeted approaches focusing on corruption types that cause moderate and severe performance degradation, such as data augmentation with specific corruption types, architectural enhancements, and preprocessing techniques that enhance the quality of input data.

4.3. Impact of model size on robustness

To explore the correlation between model size and robustness across various corruption types, we conduct a comparative analysis of average robustness among different model sizes. The averaged results of all 25 corruption types at two stages are presented in Figure 6. This figure shows a prevailing trend on our benchmark datasets: larger models, as adopted in all four examined methodologies, consistently demonstrate higher average robustness across the three different datasets. As shown in Tables 1 and 2, even when performance on clean datasets is comparable across different model sizes within the same architecture (e.g., among RT-DETR models or among YOLOv8 models on SeaShips), larger models still demonstrate superior robustness under corruption. These findings emphasize the need

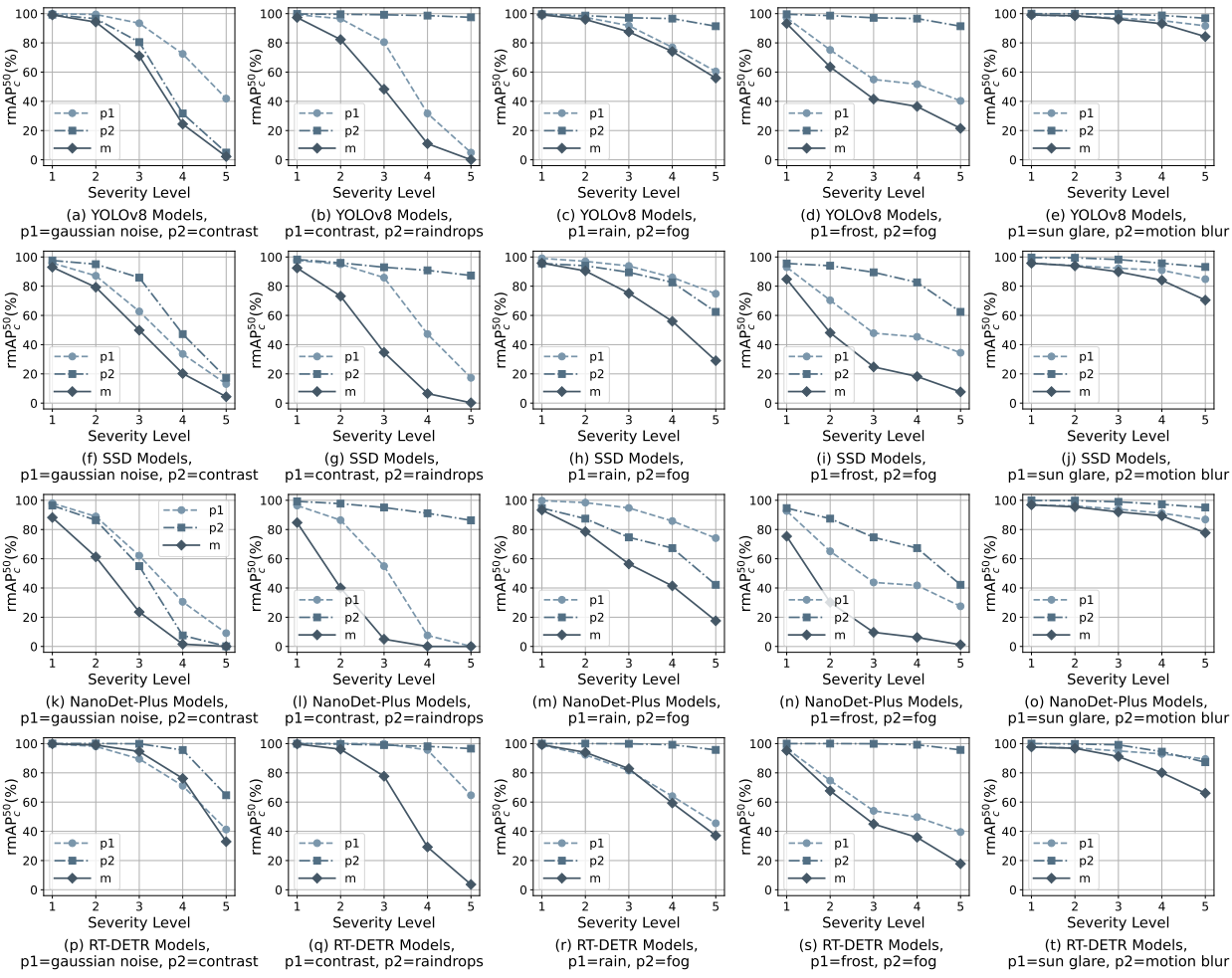


Figure 7: The rmAP^{50}_c scores of the five common mixed corruption types compared with their corresponding two pure corruption types on SeaShips-C. Within each column of sub-figures, $p1$ and $p2$ denote two distinct pure corruption types, and m represents the mixed corruption resulting from their combination. Note that the pairs of corruptions ($p1$, $p2$) differ across columns but remain consistent within the sub-figures of the same column. Each row of sub-figures corresponds to results from a specific model family, with scores averaged across model sizes within that family.

for a strategic approach in selecting, designing, and verifying object detection models for ASVs. It is crucial to balance robustness, model complexity, and resource constraints to enhance detection robustness.

Moreover, improvements in robustness due to larger model sizes are not uniformly observed across all corruption types or models. For example, larger YOLOv8 models show significant enhancements in robustness against corruptions such as snow, frost, fog, and contrast. In contrast, for YOLOv8, the benefits of increased model size are less pronounced when facing Gaussian noise and shot noise. Interestingly, for RT-DETR, larger models provide pronounced robustness improvements specifically against Gaussian noise and shot noise. This indicates that the impact of model size on robustness is both model-specific and dependent on the type of corruption.

4.4. Pure corruption vs. mixed corruption across severity levels

To further investigate the robustness benchmark regarding severity levels, we present the performance of the five YOLOv8 models under four newly added pure corruption types: rain, sun glare, occlusion, and raindrops (see Figure 10 in Appendix D). The rmAP^{50}_c scores for all models show a noticeable decline as the severity of newly-added pure corruption increases. This observation aligns with intuition, as the severity levels are subjectively defined by humans and are visually discernible. Furthermore, the larger models consistently outperform their smaller counterparts under

the newly-added pure corruption, especially when the corruption severity level is high. This again demonstrates the relatively higher robustness of larger models against severe corruption in waterborne environments.

To further explore the synthesized mixed corruption, we present the performance of YOLOv8, SSD, NanoDet-Plus, and RT-DETR on SeaShips-C under the five common mixed corruption types alongside the corresponding pure corruption types. The results are depicted in Figure 7, showing the rmAP_c^{50} scores for models across five severity levels. The results indicate that mixed corruptions generally lead to a more significant performance drop than their individual counterparts. Notably, the impact on performance is not merely additive or multiplicative. For instance, while raindrops corruption of severity level 3 alone causes a mild performance drop, its combination with contrast corruption leads to a significantly greater drop in performance than that caused by either the pure raindrops or contrast corruption alone. Correspondingly, mixed corruption types can often result in detection failures that are not observed under the individual pure corruption types. For instance, a YOLOv8 model, while capable of detecting objects under individual raindrops and contrast corruptions, fails under mixed raindrops with contrast corruption, see Figure 11 in Appendix D. These findings are further supported by the distinct Fourier spectrum patterns of mixed corruptions, as visualized in Figure 12 in Appendix D. They highlight the necessity of incorporating mixed corruption types into our benchmark to enhance detection robustness and dependability.

4.5. Benchmarking results and real-world alignment

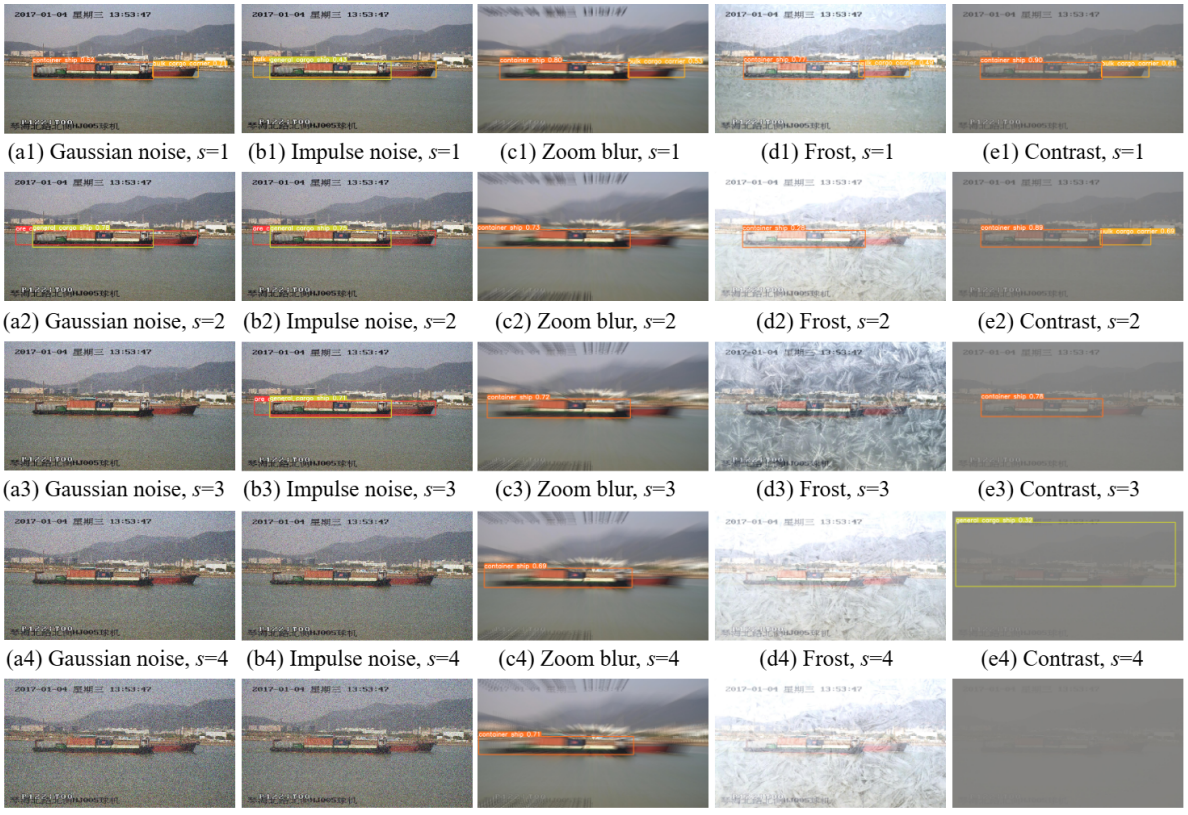


Figure 8: Detection failures using YOLOv8 models appear in a challenging scenario from SeaShips with overlapping ships, i.e., a container ship in the front and a bulk cargo carrier in the back, under various corruption types. The columns of the figure stand for corruption types, while the rows stand for the levels of severity s from 1 to 5. Detection failures such as missed detections (e.g. (a3)), misclassifications (e.g. (b1)), inaccurate bounding boxes (e.g. (e4)), and spurious detections (e.g. (b3)) where the container ship in the front is detected as two ships, a general cargo ship and an ore carrier) are included.

Our robustness benchmark evaluates the performance of object detection models under simulated corruptions and ensures that these evaluations translate meaningfully to real-world waterborne perception challenges. Given that visual

perception is critical for ASV's situational awareness, failures in perception systems could propagate to navigational risks according to Fan et al. (2020). Figure 8 presents typical failure cases observed under various corruption severities. Four main observed failure types and their relations to real-world navigational risks are as follows:

- Missed detections (false negatives): Missed detections are particularly prevalent among smaller objects, such as fishing boats, whose undetected presence could lead to collision risks.
- Spurious detections (false positives): Similarly, reductions in these metrics reflect an uptick in false positives. This can result in unnecessary maneuvers, disrupting the ASV's navigation and potentially leading to inefficient operations.
- Misclassifications: Corruption can lead to object misclassification, where detected objects are incorrectly identified. For instance, mistaking a ship for a buoy can have severe navigational consequences.
- Inaccurate bounding boxes: Inaccurate bounding boxes can misrepresent the size and position of detected objects, further complicating navigation and collision avoidance.

The occurrence of these failure modes correlated with drops in rmAP_c^{50} and rmAP_c scores, indicating a clear link between our benchmark metrics and actual detection failures caused by corruption. By quantifying the impact of various corruption types, our benchmark enables developers to pinpoint vulnerabilities early in the model development lifecycle, reducing the risk of critical failures once ASVs are operational. Consequently, these results support our benchmark's effectiveness as a standardized tool for robust, pre-deployment model verification in waterborne environments.

5. Conclusion and Discussion

In this paper, we introduced a novel robustness benchmark designed explicitly for ASVs, synthesizing a range of realistic waterborne corruptions such as adverse weather, blur, noise, occlusions, and sun glare across 25 corruption types and five severity levels. The benchmark's ability to simulate pure and mixed corruption scenarios provides a test verification framework that supports model verification and validation, a key phase in the functional safety lifecycle defined in IEC 61508 and reiterated in IEC/ISO TR 5469. Alongside, we created three enhanced datasets—SeaShips-C, SMD-C, and SSAVE-C—by augmenting established waterborne datasets with synthetic corruptions. These datasets, along with the corruption algorithms which can be utilized for either verification or data augmentation during training, providing valuable resources for both the research community and industrial applications.

Our extensive evaluation of state-of-the-art object detection models reveals significant vulnerabilities across models and datasets. The findings also highlight a strong correlation between model size and robustness, with larger models generally exhibiting higher resilience to corruption. Furthermore, our comparative analysis of pure versus mixed corruption demonstrates that mixed corruption leads to performance drops beyond the effects of individual corruption alone. Additionally, we established a correlation between our benchmark metrics and real-world detection failures, validating the benchmark's relevance to navigational safety and its practical implications for ASV operations. These results highlight the critical need to ensure the safety and robustness of ASVs in diverse waterborne conditions.

In designing our benchmark, we extended beyond the 15 common corruption types of ImageNet-C to include several corruption types that are particularly relevant to the ASV context, such as rain, raindrops and water droplets adhered to camera lens or bridge windows, sun glare, occlusion, and mixed corruption scenarios like rain with fog. While some of these corruptions have been considered in other domains, they are underrepresented in standard benchmarks (e.g., Cityscapes-C). Moreover, their manifestation and operational impact in waterborne environments can be distinct. For instance, sun glare is typically more prominent in ASVs due to the reflective properties of water surfaces.

Although our benchmark was tailored for ASVs, we believe these domain-specific corruption types have broader relevance to safety-critical perception systems in other autonomous platforms, such as drones and self-driving cars, which may encounter similar visual challenges. This raises opportunities to explore their generalization and incorporation into broader cross-domain benchmarking efforts.

To further advance this field, several key areas emerge for future research:

- Enhancing benchmark comprehensiveness: While our benchmark currently includes a wide range of corruption types, it is not exhaustive and relies on subjective judgments. Future research could benefit from incorporating additional corruption types to augment the robustness benchmark, utilizing data-driven methodologies. This approach would not only increase the benchmark's comprehensiveness but also ensure it remains relevant and adaptive to emerging challenges in waterborne environments.

- Extending the benchmark to multimodal object detection systems: Recognizing the limitations of relying solely on vision-based detection, there is a need to broaden our robustness benchmark to include multimodal sensor systems. This expansion is significant for evaluating the overall system robustness more effectively. Achieving this goal requires efforts from the community to collect and make more publicly available synchronized multimodal data tailored for ASVs. Such datasets are currently limited, and their development is critical for facilitating comprehensive system evaluations and promoting advancements in ASV technologies.

- Bridging the gap between synthetic and real-world corruptions: Our benchmark currently relies on synthesized corruption due to the scarcity of annotated real-world corrupted data in waterborne settings. To further improve practical applicability, future work could collect and integrate datasets with naturally occurring corruption observed in waterborne environments. Additionally, empirical studies are needed to validate the extent to which performance under synthetic corruption translates to real-world robustness, extending validation efforts from Hendrycks et al. (2021), which show that performance on synthetic corruptions (ImageNet-C) reliably predicts real-world robustness. This would enhance the benchmark's realism and its value for ASV deployment.

- Development of safety monitors for deep learning models used in ASVs: A safety monitor can observe a machine learning model or its environment to trigger interventions that ensure system safety. Developing such monitors is crucial for assessing and indicating the risks associated with deep learning models at runtime (Ferreira (2023)). By integrating safety monitoring, the system can not only detect objects reliably but also continuously evaluate and respond to potential risks in the detection process. Our benchmark be easily adapted to verify both object detection and monitoring components, enabling a comprehensive verification of system safety and robustness.

These initiatives aim to advance the robustness and dependability of ASVs, ensuring that they meet the operational safety standards in increasingly complex waterborne environments. By addressing these areas, contributions can be made to the dependable integration of artificial intelligence enabled systems in waterborne operations, ultimately leading to safer navigation and enhanced situational awareness for ASVs.

A. Appendix: Severity definition for rain, raindrops, sun glare, and occlusion

This appendix further elaborated on the severity levels as discussed in Sections 3.1.2 to 3.1.5.

A.1. Rain

The parameters of the rain streaks (tilt angles, streak length, number of streaks, streak intensity) for the 5 severity levels are shown in table 3. The ranges of the parameters are calibrated on the basis of the Rain1200 dataset in Zhang & Patel (2018).

A.2. Raindrops

A maximum raindrop mask containing 300 raindrops is generated for each original image. Each severity level corresponds to a subsample of this mask, with the number of raindrops n, $n \in \{60, 120, 180, 240, 300\}$.

A.3. Sun glare

Ranging from severity level 1 to 5, the center of the circular masks is fixed, while the outermost radius $r \in \{50, 60, 70, 80, 90\}$, corresponds to SeaShips images with a height of 1080 pixels, and is linearly scaled with image height for other resolutions. The accompanying diamond-shaped masks grow accordingly, with longer diagonal $l \in \{300, 360, 420, 480, 540\}$ and shorter diagonal $s \in \{10, 12, 14, 16, 18\}$, ensuring that both radial and directional glare components scale consistently with severity.

Table 3
Modeling details for rain.

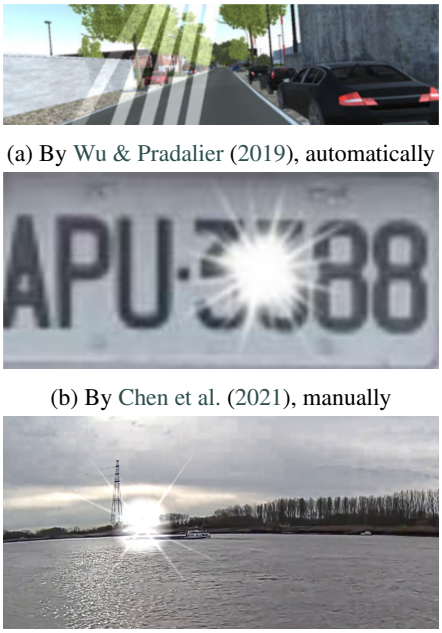
Severity Level	Tilt Angles (°)	Streak Length (psi)	Number of streaks	Streak Intensity
1		50	2000	50
2		55	2449	70
3	[-20, 20]	60	3000	90
4		65	3674	110
5		70	4500	130

A.4. Occlusion

The five severity levels of occlusion are determined by two key factors: the location of the occluded region relative to the object, and the ratio of the occluded area to the size of the object's bounding box. At severity level 1, the masked area does not touch the object itself but instead covers part of its surrounding environment. For levels 2 and 3, the occlusion begins to overlap with the object. At the highest severity levels, 4 and 5, the occlusion lies entirely within the object's bounding box, causing substantial visual obstruction. The occlusion area ratios for these five levels are set as $\alpha \in \{0.15, 0.15, 0.3, 0.3, 0.5\}$, where similar numerical values (e.g., 0.15 for both levels 1 and 2, and 0.3 for levels 3 and 4) correspond to different spatial contexts: outside the object, partial overlap, and full containment.

B. Appendix: A supplementary figure comparing our synthesized sun glare with existing methods

Figure 9 presents the comparison of sun glare corruption in existing literature and our method.



(a) By Wu & Pradalier (2019), automatically

(b) By Chen et al. (2021), manually

(c) Our method, automatically

Figure 9: Synthesized sun glare corruption in existing literature and our method.

C. Appendix: The tables of rmAP_c^{50} results for the two stages

Detailed rmAP_c^{50} results are presented in Table 4 and Table 5. Table 4 shows the rmAP_c^{50} scores for all four evaluated model families on the SeaShips-C dataset (Stage 1). Table 5 presents the results for YOLOv8 models on the

Table 4
rmAP_c^{50} (%) - Robustness of YOLOv8, SSD, NanoDet-Plus, and RT-DETR on the SeaShips-C dataset.

Approach	**YOLOv8**					**SSD-ResNet**			**NanoDet-Plus**		**RT-DETR**		
Model	**n**	**s**	**m**	**l**	**x**	**-18**	**-34**	**-50**	**-m**	**-m-1.5x**	**R18**	**R34**	**R50**
gaussian noise	76.1	79.2	84.4	83.8	83.7	58.1	59.4	58.1	56.5	58.9	73.0	79.8	**87.1**
shot noise	73.5	75.9	81.2	80.4	80.7	54.4	56.0	55.4	50.6	53.2	68.9	76.8	**84.3**
impulse noise	76.2	78.9	84.8	84.1	84.2	57.3	58.4	56.8	55.4	57.0	73.5	79.9	**86.7**
defocus blur	**98.9**	98.5	98.6	98.6	98.6	93.6	94.4	92.0	97.4	97.6	97.1	97.9	98.1
glass blur	**99.8**	**99.8**	**99.8**	**99.8**	99.7	97.7	98.4	97.3	99.3	99.2	98.7	98.8	98.9
motion blur	98.3	99.0	99.2	**99.4**	**99.4**	97.2	97.9	96.6	97.9	98.5	94.7	96.2	97.5
zoom blur	62.3	64.4	63.4	66.2	66.3	57.1	57.8	53.2	**77.8**	77.1	39.1	45.4	51.9
snow	59.3	64.0	73.4	74.1	**74.9**	53.7	57.9	61.3	52.4	64.0	57.4	52.0	58.4
frost	54.1	60.4	67.6	68.1	**69.1**	53.9	59.9	60.8	53.9	54.4	63.0	62.1	63.7
fog	95.0	95.5	97.0	97.7	98.2	81.9	83.6	89.0	72.5	74.0	98.5	99.0	**99.4**
brightness	99.4	99.4	**99.7**	99.5	99.5	94.5	95.0	95.6	96.1	96.4	99.4	99.6	99.4
contrast	57.0	59.1	61.7	65.6	69.6	64.0	67.4	74.4	49.2	48.9	89.3	93.2	**93.8**
elastic transform	**100.0**	**100.0**	**100.0**	**100.0**	99.9	99.6	**100.3**	99.4	**100.0**	**100.0**	99.9	99.9	99.9
pixelate	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	99.7	**100.3**	99.6	**100.0**	99.9	**100.0**	**100.0**	99.9
jpeg compression	99.7	99.8	99.8	**99.9**	**99.9**	99.0	99.5	98.9	99.0	99.4	99.3	99.5	99.4
occlusion	86.3	86.4	86.6	87.9	87.7	81.3	83.0	80.2	87.3	88.3	88.4	**88.9**	88.0
rain	78.2	79.3	88.8	90.1	90.0	88.8	90.7	90.9	88.7	**92.5**	76.1	77.9	75.5
raindrops	98.3	98.9	99.0	**99.3**	**99.3**	92.4	93.4	93.5	93.6	94.1	98.1	98.6	**99.3**
sun glare	95.5	96.5	96.4	96.3	**96.7**	91.1	92.1	91.6	92.0	93.9	92.9	94.7	95.7
gaussian noise+contrast	52.2	54.3	57.7	61.7	65.1	46.8	50.5	50.7	34.9	35.0	76.2	82.0	**83.4**
contrast+raindrops	40.5	45.8	48.3	51.9	52.4	37.7	39.0	47.6	25.9	26.0	54.5	61.4	**67.9**
rain+fog	74.6	75.7	85.4	87.5	**89.6**	64.9	70.1	72.9	56.7	58.2	76.2	73.7	73.6
frost+fog	41.7	46.4	51.8	55.9	**60.4**	32.8	36.0	41.4	24.3	24.7	49.5	51.1	56.3
sun glare+motion blur	92.3	94.1	94.6	95.0	**95.2**	86.5	88.0	85.9	89.6	91.0	84.9	85.6	88.7
random	67.5	70.6	74.2	75.1	**75.7**	57.9	60.7	60.9	59.9	62.4	63.3	65.8	68.4

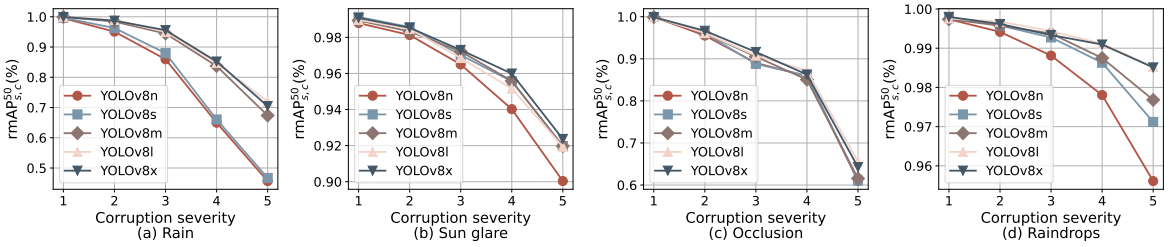


Figure 10: The results of YOLOv8 models under the four newly-added pure corruption types on SeaShips

SMD-C and SSAVE-C datasets (Stage 2); their results on SeaShips-C are already provided in Table 4. Notably, we observe that for a small number of corruptions, the relative performance slightly exceeds 100% compared to the clean dataset (e.g., 100.1% and 100.3%). This could be attributed to stochastic variations in performance, and the presence of noisy labels and imprecisely located bounding boxes, as the labeling issue within datasets such as SMD has been acknowledged in other works (Kim et al. (2022)).

D. Appendix: Supplementary figures for pure corruption vs. mixed corruption

Figure 10 presents the rmAP_c^{50} scores of different YOLOv8 model variants across five severity levels.

Table 5
rmAP_c^{50} (%) - Robustness of YOLOv8 on the SMD-C and SSAVE-C datasets.

Dataset	SMD-C					SSAVE-C				
Model	YOLO v8n	YOLO v8s	YOLO v8m	YOLO v8l	YOLO v8x	YOLO v8n	YOLO v8s	YOLO v8m	YOLO v8l	YOLO v8x
gaussian noise	56.5	62.5	**64.4**	62.7	61.0	76.7	79.3	81.4	**84.3**	82.0
shot noise	46.1	51.5	**54.5**	53.2	51.0	78.3	77.8	83.1	**84.6**	81.0
impulse noise	54.4	60.9	**63.2**	61.4	58.7	78.1	80.4	84.0	**86.1**	82.1
defocus blur	95.0	96.0	95.8	**96.3**	96.2	92.5	93.6	**95.5**	95.1	95.0
glass blur	98.0	98.3	98.3	**98.7**	98.6	93.3	93.6	**94.6**	93.4	93.6
motion blur	95.8	96.8	97.1	**97.3**	**97.3**	**87.3**	85.5	86.7	85.5	86.9
zoom blur	42.2	48.8	50.8	50.7	**51.8**	29.1	33.5	35.7	37.8	**39.5**
snow	44.9	62.0	66.2	68.7	**70.3**	75.7	81.7	89.5	88.9	**90.1**
frost	44.6	60.1	64.8	68.2	**73.1**	60.7	73.2	83.4	83.9	**85.9**
fog	90.2	94.3	96.4	97.0	**97.4**	78.1	87.2	92.8	93.6	**96.1**
brightness	98.0	98.7	**99.5**	**99.5**	**99.5**	96.5	97.0	**98.4**	97.7	**98.4**
contrast	52.6	65.3	68.1	73.8	**77.0**	45.3	53.7	62.2	62.8	**67.1**
elastic transform	99.9	99.9	**100.0**	**100.0**	**100.1**	97.7	**98.4**	97.7	97.1	98.0
pixelate	**100.0**	**100.0**	**100.1**	**100.0**	**100.1**	99.9	**100.0**	99.1	99.7	**100.2**
jpeg compression	96.1	97.2	97.1	**97.5**	96.8	95.6	**96.9**	95.9	95.3	96.3
occlusion	97.5	**97.8**	97.6	97.6	**97.8**	97.1	97.1	**98.0**	97.8	**98.0**
rain	63.4	83.0	87.5	90.8	**92.3**	89.0	92.7	96.9	97.7	**98.3**
raindrops	84.4	84.2	90.7	90.9	**92.1**	92.8	93.9	94.4	95.4	**96.5**
sun glare	93.5	94.9	95.8	95.8	**96.1**	93.2	94.2	95.4	95.3	**95.6**
gaussian noise+contrast	46.0	57.7	64.6	70.9	**77.0**	40.0	47.0	56.2	59.8	**60.9**
contrast+raindrops	24.0	28.7	35.7	35.9	**39.3**	31.9	40.8	50.3	53.0	**56.1**
rain+fog	69.8	79.5	85.6	87.9	**90.6**	57.3	69.2	85.6	88.5	**89.6**
frost+fog	44.9	57.9	59.5	65.6	**68.3**	28.9	44.4	56.1	63.6	**66.2**
sun glare+motion blur	89.4	91.6	92.8	**93.2**	**93.2**	82.8	82.3	82.8	83.6	**84.1**
random	55.2	61.5	65.9	66.8	**67.8**	67.6	69.5	75.9	**76.6**	76.4

An example in which a YOLOv8 model fails under mixed corruption but still detects the objects under pure corruptions is shown in Figure 11.



(a) Raindrops (b) Contrast (c) Raindrops+contrast

Figure 11: An example of a YOLOv8 model fails to detect an object under mixed corruption (raindrops+contrast, severity 2), while it succeeds in detecting this object under raindrops corruption (severity 2) and contrast corruption (severity 2), respectively. The images are from the SeaShips-C dataset. The two objects here are both general cargo ships.

The Fourier spectra visualization of the 25 corruptions on SMD-C is presented in Figure 12. Each spectrum is computed from the differences between corrupted and corresponding clean images, applying a 2D Fourier transform, and averaging across all images for each corruption. The center of each spectrum visualization represents low-frequency image content (global) while the outer regions display high-frequency content (details). Colors—ranging from dark blue (low amplitude) to red, yellow and white (high amplitude)—reflect the log-transformed magnitude of spectra.

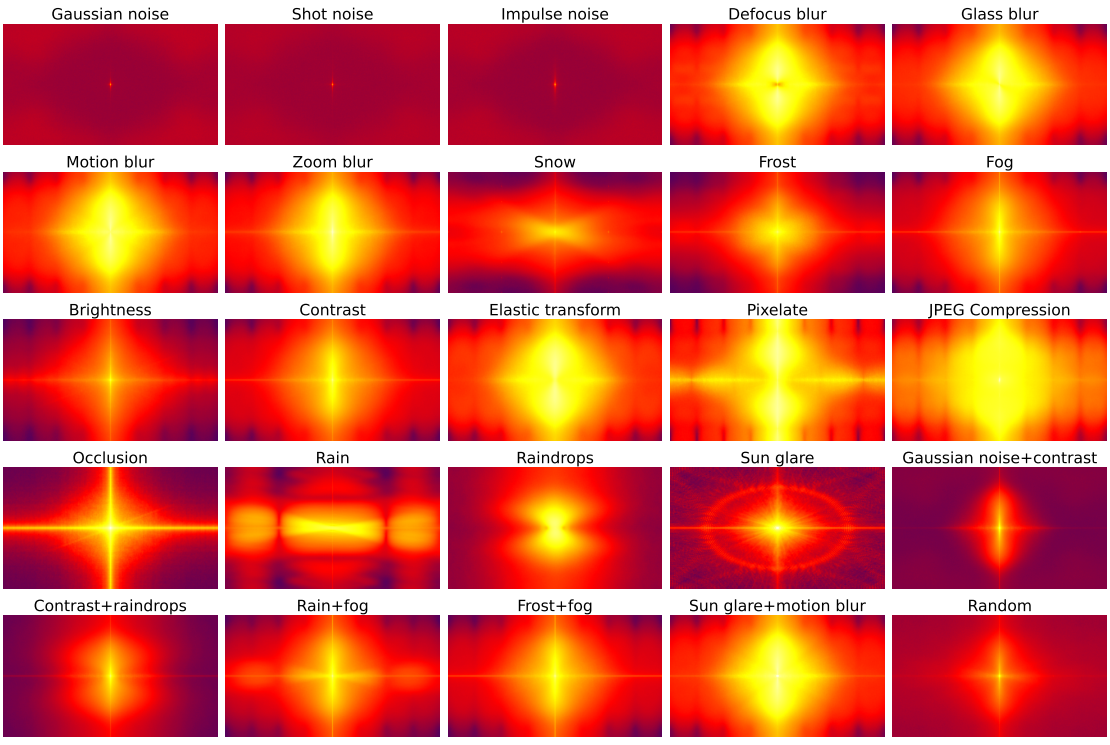


Figure 12: Fourier spectra visualization of the 25 corruptions in SMD-C at severity 3.

Notably, mixed corruptions (e.g., contrast+raindrops) exhibit distinct frequency patterns compared to their constituent pure corruptions (contrast, raindrops), underscoring the need to include mixed corruption types in the benchmark.

CRediT authorship contribution statement

Yunjia Wang: Conceptualization of this study, Data curation, Methodology, Software, Investigation, Writing – original draft. **Zihao Zhang:** Data curation, Methodology, Software, Investigation. **Kaizheng Wang:** Data curation, Methodology, Investigation. **Holger Caesar:** Supervision, Writing – review & editing. **Jeroen Boydens:** Supervision, Writing – review & editing. **Davy Pissoort:** Supervision, Writing – review & editing. **Mathias Verbeke:** Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

We have made the code and data associated with this work available at `https://gitlab.kuleuven.be/m-group-campus-brugge/dtai_public/publications/Benchmarking_robustness_object_detection_ASVs`.

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 955768 (AUTOBarge) and FET-Open grant agreement No 964505 (E-pi). This research was partially supported by Flanders Make, the strategic research center for the

manufacturing industry, as part of the Strategic Basic Research project SAIfety. We would like to express our sincere gratitude to Shiming Wang, PhD researcher at TU Delft, and Ming Wu, Postdoctoral researcher at KU Leuven, for their generous feedback and constructive comments that significantly enhanced this work.

References

Adams, S. (2014). Revolt–next generation short sea shipping.

Ahmed, M., Bakht, A. B., Hassan, T., Akram, W., Humais, A., Seneviratne, L., He, S., Lin, D., & Hussain, I. (2023). Vision-based autonomous navigation for unmanned surface vessel in extreme marine conditions. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 7097–7103). IEEE.

Arsenos, A., Karampinis, V., Petrongonas, E., Skliros, C., Kollias, D., Kollias, S., & Voulodimos, A. (2024). Common corruptions for evaluating and enhancing robustness in air-to-air visual object detection. *IEEE Robotics and Automation Letters*, .

Bai, X., Liu, M., Wang, T., Chen, Z., Wang, P., & Zhang, Y. (2016). Feature based fuzzy inference system for segmentation of low-contrast infrared ship images. *Applied Soft Computing*, *46*, 128–142.

Barbu, A., Mayo, D., Alverio, J., Luo, W., Wang, C., Gutfreund, D., Tenenbaum, J., & Katz, B. (2019). ObjectNet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d. Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems*. Curran Associates, Inc. volume 32.

Bertram, V. (2016). Unmanned & autonomous shipping: A technology review. In *Proceedings of the 10th Symposium on High-Performance Marine Vehicles, Cortona* (pp. 10–24).

Bloisi, D. D., Iocchi, L., Nardi, D., Fiorini, M. et al. (2015). Integrated visual information for maritime surveillance. In *Clean Mobility and Intelligent Transport Systems* (pp. 237–263). Institution of Engineering and Technology.

Bloisi, D. D., Previtali, F., Pennisi, A., Nardi, D., & Fiorini, M. (2016). Enhancing automatic maritime surveillance systems with visual information. *IEEE Transactions on Intelligent Transportation Systems*, *18*, 824–833.

Chen, B.-H., Ye, S., Yin, J.-L., Cheng, H.-Y., & Chen, D. (2021). Deep trident decomposition network for single license plate image glare removal. *IEEE Transactions on Intelligent Transportation Systems*, *23*, 6596–6607.

Chen, X., Qi, L., Yang, Y., Luo, Q., Postolache, O., Tang, J., Wu, H. et al. (2020). Video-based detection infrastructure enhancement for automated ship recognition and behavior analysis. *Journal of Advanced Transportation*, *2020*.

Cheng, Y., Xu, H., & Liu, Y. (2021). Robust small object detection on the water surface through fusion of camera and millimeter wave radar. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 15263–15272).

Dodge, S., & Karam, L. (2016). Understanding how image quality affects deep neural networks. In *2016 8th International Conference on Quality of Multimedia Experience (QoMEX)* (pp. 1–6). IEEE.

Dong, Y., Kang, C., Zhang, J., Zhu, Z., Wang, Y., Yang, X., Su, H., Wei, X., & Zhu, J. (2023). Benchmarking robustness of 3D object detection to common corruptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 1022–1032).

European Commission (2021). Proposal for a regulation laying down harmonised rules on Artificial Intelligence and amending certain union legislative acts. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A52021PC0206.

Fan, C., Wróbel, K., Montewka, J., Gil, M., Wan, C., & Zhang, D. (2020). A framework to identify factors influencing navigational risk for maritime autonomous surface ships. *Ocean Engineering*, *202*, 107188.

Ferreira, R. S. (2023). *Runtime safety monitoring of ML-based perception functions in autonomous systems*. Ph.D. thesis Centre National de la Recherche Scientifique.

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., & Brendel, W. (2018a). Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*.

Geirhos, R., Temme, C., Rauber, J., Schütt, H., Bethge, M., & Wichmann, F. (2018b). Generalisation in humans and deep neural networks. In *32nd Conference on Neural Information Processing Systems (NeurIPS 2018)* (pp. 7538–7550). Curran Associates, Inc.

Goyal, B., Dogra, A., Agrawal, S., Sohi, B. S., & Sharma, A. (2020). Image denoising review: From classical to state-of-the-art approaches. *Information fusion*, *55*, 220–244.

Guo, J., Feng, H., Xu, H., Yu, W., & shuzhi Ge, S. (2023). D3-Net: integrated multi-task convolutional neural network for water surface deblurring, dehazing and object detection. *Engineering Applications of Artificial Intelligence*, *117*, 105558.

Halder, S. S., Lalonde, J.-F., & Charette, R. d. (2019). Physics-based rendering for improving robustness to rain. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 10203–10212).

Hao, Z., You, S., Li, Y., Li, K., & Lu, F. (2019). Learning from synthetic photorealistic raindrop for single image raindrop removal. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*.

Hawkins, R., Paterson, C., Picardi, C., Jia, Y., Calinescu, R., & Habli, I. (2021). Guidance on the assurance of machine learning in autonomous systems (AMLAS). *arXiv preprint arXiv:2102.01564*, .

Hayes, M. P., & Gough, P. T. (2009). Synthetic aperture sonar: A review of current status. *IEEE journal of oceanic engineering*, *34*, 207–224.

Heidarsson, H. K., & Sukhatme, G. S. (2011). Obstacle detection and avoidance for an autonomous surface vehicle using a profiling sonar. In *2011 IEEE International Conference on Robotics and Automation* (pp. 731–736). IEEE.

Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M. et al. (2021). The many faces of robustness: A critical analysis of out-of-distribution generalization. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)* (pp. 8320–8329). IEEE Computer Society.

Hendrycks, D., & Dietterich, T. (2019). Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*.

Hendrycks, D., Mu, N., Cubuk, E. D., Zoph, B., Gilmer, J., & Lakshminarayanan, B. (2019). Augmix: A simple data processing method to improve robustness and uncertainty. In *International Conference on Learning Representations*.

Hesselbarth, A., Medina, D., Ziebold, R., Sandler, M., Hoppe, M., & Uhlemann, M. (2020). Enabling assistance functions for the safe navigation of inland waterways. *IEEE Intelligent Transportation Systems Magazine*, *12*, 123–135.

Huang, X., Kroening, D., Ruan, W., Sharp, J., Sun, Y., Thamo, E., Wu, M., & Yi, X. (2020). A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. *Computer Science Review*, *37*, 100270.

Huang, Y., Wang, H., Ma, J., Lou, J., & Yi, H. (2021). Research and practical exploration of test and validation technologies applied on unmanned surface vehicle optical recognition. In *2021 IEEE International Conference on Unmanned Systems (ICUS)* (pp. 976–981). IEEE.

IEC (2010). *IEC 61508: Functional safety of electrical/electronic/programmable electronic safety-related systems*. Technical Report IEC 61508-1:2010 IEC.

ISO and IEC (2023). *ISO/IEC 5338:2023 Information technology — Artificial intelligence — AI system life cycle processes*. Technical Report ISO/IEC 5338:2023 ISO/IEC. URL: https://www.iso.org/standard/81118.html.

ISO and IEC (2024). *ISO/IEC TR 5469:2024 Artificial Intelligence – Functional safety and AI systems*. Technical Report ISO/IEC TR 5469:2024 ISO/IEC. URL: https://www.iso.org/standard/81283.html.

Joaedl (2019). RainGenerator. https://github.com/joaedl/RainGenerator.

Jocher, G., Chaurasia, A., & Qiu, J. (2023). YOLO by Ultralytics. URL: https://github.com/ultralytics/ultralytics.

Kim, J.-H., Kim, N., Park, Y. W., & Won, C. S. (2022). Object detection and classification based on YOLO-V5 with improved maritime dataset. *Journal of Marine Science and Engineering*, *10*, 377.

Komianos, A. (2018). The autonomous shipping era. operational, regulatory, and quality challenges. *TransNav: International Journal on Marine Navigation and Safety of Sea Transportation*, *12*.

Lahouli, R., Cubber, G. D., Pairet, B., Hamesse, C., Fréville, T., & Haelterman, R. (2022). Deep learning based object detection and tracking for maritime situational awareness. In *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications* (pp. 643–650).

Li, C. (2018). High quality, fast, modular reference implementation of SSD in PyTorch. https://github.com/lufficc/SSD.

Liang, W., Long, J., Li, K.-C., Xu, J., Ma, N., & Lei, X. (2021). A fast defogging image recognition algorithm based on bilateral hybrid filtering. *ACM transactions on multimedia computing, communications, and applications (TOMM)*, *17*, 1–16.

Liu, C., Dong, Y., Xiang, W., Yang, X., Su, H., Zhu, J., Chen, Y., He, Y., Xue, H., & Zheng, S. (2024). A comprehensive study on robustness of image classification models: Benchmarking and rethinking. *International Journal of Computer Vision*, (pp. 1–23).

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14* (pp. 21–37). Springer.

Lyu, H., Shao, Z., Cheng, T., Yin, Y., & Gao, X. (2022). Sea-surface object detection based on electro-optical sensors: A review. *IEEE Intelligent Transportation Systems Magazine*, *15*, 190–216.

Michaelis, C., Mitzkus, B., Geirhos, R., Rusak, E., Bringmann, O., Ecker, A. S., Bethge, M., & Brendel, W. (2019). Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv preprint arXiv:1907.07484*, .

Moosbauer, S., Konig, D., Jakel, J., & Teutsch, M. (2019). A benchmark for deep learning based object detection in maritime environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Mu, N., & Gilmer, J. (2019). MNIST-C: A robustness benchmark for computer vision. *arXiv preprint arXiv:1906.02337*, .

Negenborn, R. R., Goerlandt, F., Johansen, T. A., Slaets, P., Valdez Banda, O. A., Vanelslander, T., & Ventikos, N. P. (2023). Autonomous ships are on the horizon: here's what we need to know. *Nature*, *615*, 30–33.

Nobis, F., Geisslinger, M., Weber, M., Betz, J., & Lienkamp, M. (2019). A deep learning-based radar and camera sensor fusion architecture for object detection. In *2019 Sensor Data Fusion: Trends, Solutions, Applications (SDF)* (pp. 1–7). IEEE.

Prasad, D. K., Rajan, D., Rachmawati, L., Rajabally, E., & Quek, C. (2017). Video processing from electro-optical sensors for object detection and tracking in a maritime environment: A survey. *IEEE Transactions on Intelligent Transportation Systems*, *18*, 1993–2016.

RangiLyu (2021). NanoDet-Plus: Super fast and high accuracy lightweight anchor-free object detection model. https://github.com/RangiLyu/nanodet.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 779–788).

Ren, J., Pan, L., & Liu, Z. (2022). Benchmarking and analyzing point cloud classification under corruptions. In *International Conference on Machine Learning* (pp. 18559–18575). PMLR.

Rødseth, Ø. J. (2017). From concept to reality: Unmanned merchant ship research in norway. *Proceedings of Underwater Technology (UT), IEEE, Busan, Korea*, .

Rødseth, Ø. J., & Burmeister, H.-C. (2015). Risk assessment for an unmanned merchant ship. *TransNav: International Journal on Marine Navigation and Safety of Sea Transportation*, *9*, 357–364.

Rothblum, A. M. (2000). Human error and marine safety. In *National Safety Council Congress and Expo, Orlando, FL*. volume 7.

Shao, Z., Wang, L., Wang, Z., Du, W., & Wu, W. (2019). Saliency-aware convolution neural network for ship detection in surveillance video. *IEEE Transactions on Circuits and Systems for Video Technology*, *30*, 781–794.

Shao, Z., Wu, W., Wang, Z., Du, W., & Li, C. (2018). Seaships: A large-scale precisely annotated dataset for ship detection. *IEEE Transactions on Multimedia*, *20*, 2593–2604.

Stanislas, L., & Dunbabin, M. (2018). Multimodal sensor fusion for robust obstacle detection and classification in the maritime RobotX challenge. *IEEE Journal of Oceanic Engineering*, *44*, 343–351.

Szpak, Z. L., & Tapamo, J. R. (2011). Maritime surveillance: Tracking ships inside a dynamic background using a fast level-set. *Expert systems with applications*, *38*, 6669–6680.

Villa, J., Aaltonen, J., & Koskinen, K. T. (2020). Path-following with lidar-based obstacle avoidance of an unmanned surface vehicle in harbor conditions. *IEEE/ASME Transactions on Mechatronics*, *25*, 1812–1820.

Volden, Ø., Stahl, A., & Fossen, T. I. (2022). Vision-based positioning system for auto-docking of unmanned surface vehicles (USVs). *International Journal of Intelligent Robotics and Applications*, *6*, 86–103.

Wang, H., Xie, Q., Zhao, Q., & Meng, D. (2020). A model-driven deep neural network for single image rain removal. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 3103–3112).

Wang, Y., Wang, K., Zhang, Z., Boydens, J., Pissoort, D., & Verbeke, M. (2024). Navigating the waters of object detection: Evaluating the robustness of real-time object detection models for autonomous surface vehicles. In *2024 IEEE Conference on Artificial Intelligence (CAI)* (pp. 985–992). IEEE.

Waterborne Technology Platform (2011). Waterborne implementation plan: Issue May 2011.

Wei, Y., Zhang, Z., Wang, Y., Xu, M., Yang, Y., Yan, S., & Wang, M. (2021). Deraincyclegan: Rain attentive cyclegan for single image deraining and rainmaking. *IEEE Transactions on Image Processing*, *30*, 4788–4801.

Wen, L., Ding, J., & Xu, Z. (2021). Multiframe detection of sea-surface small target using deep convolutional neural network. *IEEE Transactions on Geoscience and Remote Sensing*, *60*, 1–16.

Wu, X., & Pradalier, C. (2019). Illumination robust monocular direct visual odometry for outdoor environment mapping. In *2019 International conference on robotics and automation (ICRA)* (pp. 2392–2398). IEEE.

Yin, D., Gontijo Lopes, R., Shlens, J., Cubuk, E. D., & Gilmer, J. (2019). A fourier perspective on model robustness in computer vision. *Advances in Neural Information Processing Systems*, *32*.

Zhang, C., Guo, C., & Zhang, D. (2018). Ship navigation via GPS/IMU/LOG integration using adaptive fission particle filter. *Ocean Engineering*, *156*, 435–445.

Zhang, H., & Patel, V. M. (2018). Density-aware single image de-raining using a multi-stream dense network. URL: https://arxiv.org/abs/1802.07412. arXiv:1802.07412.

Zhang, K., Ren, W., Luo, W., Lai, W.-S., Stenger, B., Yang, M.-H., & Li, H. (2022). Deep image deblurring: A survey. *International Journal of Computer Vision*, *130*, 2103–2130.

Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y., & Chen, J. (2024). DETRs beat YOLOs on real-time object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 16965–16974).