IOSR Journal of Environmental Science, Toxicology and Food Technology (IOSR-JESTFT)
e-ISSN: 2319-2402, p- ISSN: 2319-2399.Volume 19, Issue 11 Ser. 1 (November 2025), PP 52-61
www.iosrjournals.org

Real-Time Oil Spill Detection Using Deep Learning: Economic and Environmental Implications for Maritime Infrastructure

Abstract
Maritime oil spills impose substantial economic externalities and environmental costs, particularly in strategic waterways with concentrated petroleum infrastructure. This study examines the application of modern deep learning architectures to automated oil spill detection, with specific emphasis on deployment feasibility in resource-constrained operational environments. We conduct a comparative benchmark analysis of YOLOv8 instance segmentation against classical U-Net semantic segmentation using publicly available annotated aerial imagery. Our empirical results demonstrate that lightweight YOLOv8 architectures achieve superior performance on minority-class detection (Intersection over Union: 0.71 vs. 0.0006 for oil segmentation) while maintaining computational efficiency suitable for edge deployment. Through systematic threshold optimisation, we establish dual operating modes that balance detection sensitivity against false-positive suppression, addressing the operational cost trade-offs inherent in continuous monitoring systems. These findings contribute to the growing literature on AI-enabled environmental monitoring and provide empirical support for policy initiatives targeting sustainable maritime operations.

Date of Submission: 10-11-2025 Date of Acceptance: 21-11-2025

I. Introduction

1.1 Economic Context and Problem Scope

Oil spills represent a category of environmental disaster with substantial negative externalities affecting multiple economic sectors simultaneously. The maritime petroleum industry, while generating significant economic rents, creates attendant risks of catastrophic spills that impose costs on fisheries, tourism, desalination infrastructure, and ecosystem services. These risks are particularly concentrated in strategic maritime corridors characterized by high vessel density and proximity to critical infrastructure.

The Arabian Gulf exemplifies this concentration of risk and value. Containing approximately 65% of proven global petroleum reserves, the region processes over 25,000 tanker transits annually through the Strait of Hormuz alone. The United Arab Emirates (UAE), positioned at the nexus of offshore extraction and international shipping lanes, faces acute exposure to spill risks. Historical incidents demonstrate the magnitude of potential damages: the 2001 Jebel Ali spill affected multiple coastal zones, including high-value tourism areas (Mamzar, Jumeirah), while 1994 records document 15 tanker collisions along UAE shores within a single calendar year.

1.2 Economic Costs of Detection Delays

The economic impact of oil spills follows a non-linear damage function: early detection and rapid response can reduce total damages by orders of magnitude relative to delayed intervention. Traditional monitoring methodologies: visual inspection, chemical sampling, and patrol-based surveillance, impose high labour costs while providing limited spatial and temporal coverage. Synthetic Aperture Radar (SAR) and optical satellite monitoring extend coverage but introduce two critical inefficiencies: (1) data volume exceeds manual processing capacity, creating analytical bottlenecks; and (2) high false-positive rates generate unnecessary mobilisation costs for response teams.

For the UAE specifically, spill risks interact with critical desalination infrastructure dependencies. Approximately 42% of the nation's potable water derives from coastal desalination plants, creating a direct channel through which maritime pollution threatens water security, a strategic economic vulnerability with potentially severe welfare implications.

1.3 Research Contribution

This study addresses the detection bottleneck through application of convolutional neural networks (CNNs) to automated spill segmentation. We make three primary contributions to the literature:

Methodological: We provide the first direct comparison of real-time instance segmentation (YOLOv8) against semantic segmentation baselines (U-Net) for maritime oil detection, using identical data partitions and preprocessing protocols to ensure internal validity.
Operational: We develop and validate a threshold-based operating point selection methodology that allows practitioners to explicitly trade off detection completeness against false-positive suppression, matching system behaviour to deployment context (incident investigation vs. continuous monitoring).
Economic: We demonstrate that lightweight architectures can achieve near-state-of-the-art detection performance at dramatically reduced computational cost (10× fewer parameters, 5× faster inference), making edge deployment economically feasible and reducing the total cost of ownership for monitoring systems.
Our results indicate that with appropriate architecture selection and threshold tuning, automated detection systems can achieve operational utility for both retrospective analysis and prospective alerting, supporting cost-effective pollution monitoring aligned with environmental policy objectives such as the UAE's Net Zero 2050 initiative.

II. Literature Review

2.1 Economics of Environmental Monitoring

The environmental economics literature establishes that optimal monitoring intensity balances the marginal benefit of damage reduction against the marginal cost of surveillance. Traditional enforcement models assume discrete inspection events with known costs and detection probabilities. However, continuous automated monitoring fundamentally alters this optimisation problem: fixed capital costs replace variable labour, detection probability approaches unity under favourable conditions, and the relevant constraint becomes computational throughput rather than inspector availability.

Recent work on AI-enabled environmental monitoring demonstrates substantial efficiency gains relative to manual methods, though deployment remains concentrated in well-resourced jurisdictions. Our contribution extends this literature by explicitly characterising the accuracy-latency production frontier for maritime monitoring and demonstrating that low-cost architectures can achieve economically relevant detection thresholds.

2.2 Remote Sensing and Detection Technologies

SAR-based oil detection has evolved from manual interpretation to rule-based classification and, more recently, to CNN-based discrimination. While SAR provides all-weather capability and extensive coverage, its fundamental limitation lies in "look-alike" phenomena: natural biogenic films, low-wind slicks, and infrastructure shadows generate false positives that cannot be fully eliminated through spectral analysis alone. Studies report false positive rates ranging from 30-60% in near-shore environments, imposing substantial verification costs on operators.

Optical and thermal imaging from aerial platforms (satellites, aircraft, UAVs) provide complementary information but generate data volumes that exceed manual review capacity. This creates an analytical bottleneck that automated segmentation can potentially resolve.

2.3 Deep Learning for Semantic Segmentation

The computer vision literature has established CNNs as the dominant approach for dense prediction tasks. U-Net and its variants (U-Net++, DeepLabV3+) employ encoder-decoder architectures with skip connections to preserve spatial information while extracting semantic features. These models excel at pixel-wise classification but treat segmentation as a pure vision problem without explicit object-level reasoning.

More recently, transformer-based architectures (SegFormer) have demonstrated strong performance through long-range dependency modelling, while YOLO-family detectors have evolved to support instance segmentation through integration of classification, localisation, and mask prediction in unified architectures. The original dataset study evaluated U-Net variants and transformer models; our work extends this by introducing real-time instance segmentation as an alternative paradigm specifically optimised for sparse, high-value targets.

2.4 Research Gap

Existing studies focus primarily on maximizing detection metrics without explicitly addressing the operational trade-offs between sensitivity and specificity in continuous monitoring contexts. Furthermore, most work emphasizes model accuracy without considering deployment constraints or total cost of ownership. Our study fills this gap by: (1) benchmarking lightweight architectures suitable for edge deployment; (2) developing operating point selection methods that map to real-world cost functions; and (3) providing inference latency measurements that enable capacity planning for operational systems.

III. Methodology

3.1 Data

We utilise a publicly available dataset of annotated RGB aerial imagery captured via unmanned aerial vehicle (UAV) in port environments between September 2021 and September 2023. The dataset comprises 1,268 high-resolution images acquired from altitudes ranging from 30 to 70 meters under varying illumination and environmental conditions. Original annotations provide pixel-level labels for four classes: Background, Oil, Water, and Other. For this analysis, we exclude the Background class as it provides no discriminative information for the segmentation task and would artificially inflate accuracy metrics due to its dominant pixel share.

Table 1 presents the class-wise pixel distribution across the retained categories. The dataset exhibits moderate class imbalance, with Oil representing 22.9% of labelled pixels compared to 36.3% (Water) and 40.8% (Other). This imbalance is characteristic of real-world monitoring scenarios where spill events are relatively rare compared to normal operating conditions, a statistical property that poses methodological challenges but also reflects the operational environment these models must ultimately serve.

Category	#Images	Pixel Count	% Share
Oil	994	527,361,085	22.9
Water	929	835,851,192	36.3
Other	1166	939,502,502	40.8

Table 1: Class-wise distribution of annotated pixels across the dataset. The background class is excluded.

We partition the dataset into mutually exclusive training (70%), validation (15%), and testing (15%) sets following the original splits to ensure comparability with prior work. For operational analysis, we further subdivide test frames into oil-present (n=202, containing any oil pixels) and no-oil (n=52, zero oil pixels) subsets, enabling separate evaluation of detection performance and false-positive characteristics.

All images and corresponding masks undergo standardised preprocessing. For U-Net training, we utilise pixel-wise masks directly. For YOLOv8, which requires polygon-based annotations, we implement a conversion pipeline: (1) extract binary masks for each class via RGB color matching; (2) apply contour detection algorithms to extract polygon boundaries; (3) normalize coordinates to [0,1] range by dividing by image dimensions; (4) format as YOLO-compatible annotation files (class_id followed by normalized polygon vertices). Input images are resized to 640×640 pixels for primary evaluation; we additionally benchmark 512×512 resolution to characterize the speed-accuracy frontier.

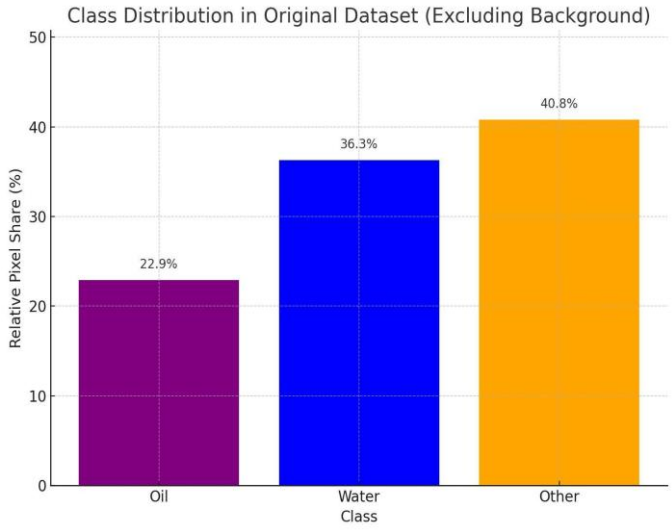

Figure 1: Class Distribution

3.2 Model Architectures

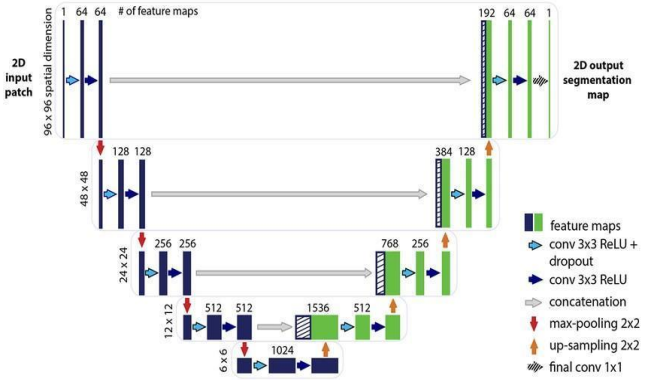

Figure 2: Diagram of the U-Net architecture. Left: contracting path (encoder), Right: expanding path (decoder), with skip connections. Our implementation used a similar structure with [3 encoder blocks → 512 channels at bottleneck → symmetric decoder].

We implement a standard U-Net architecture as our semantic segmentation baseline. The encoder consists of three downsampling blocks (Conv-ReLU-Conv-ReLU-MaxPool), progressively expanding feature channels to 512 at the bottleneck. The decoder mirrors this structure with bilinear upsampling and skip connections from corresponding encoder layers. The final layer produces three-channel probability maps (one per foreground class).

This implementation is intentionally simplified relative to the original dataset paper, which employed a U-Net with a pretrained EfficientNet-B4 encoder, attention mechanisms, and specialised activation functions. Our baseline contains no pretrained components and applies no explicit class balancing in the loss function. While this may underestimate the U-Net's maximum achievable performance, it provides a fair comparison against YOLOv8-nano (which also lacks dataset-specific pretraining) and isolates the effect of the architectural paradigm.

Training employs pixel-wise binary cross-entropy loss, treating each class as an independent binary segmentation task. We train for 50 epochs with batch size 4 and learning rate 1×10^{-4} using Adam optimisation.

YOLOv8 represents a unified architecture for object detection and instance segmentation. Unlike semantic segmentation models that predict dense class maps, YOLO performs detection (bounding box + class) and segmentation (instance mask) jointly in a single forward pass. The nano variant (YOLOv8n-seg) contains 3.7M parameters and employs an anchor-free detection paradigm with specialised loss functions for box regression, classification, and mask prediction.

We fine-tune YOLOv8n-seg using the Ultralytics training pipeline with default hyperparameters: batch size 16, initial learning rate 0.01, 50 epochs. The model outputs per-instance masks with associated confidence scores, enabling threshold-based filtering for operating point selection.

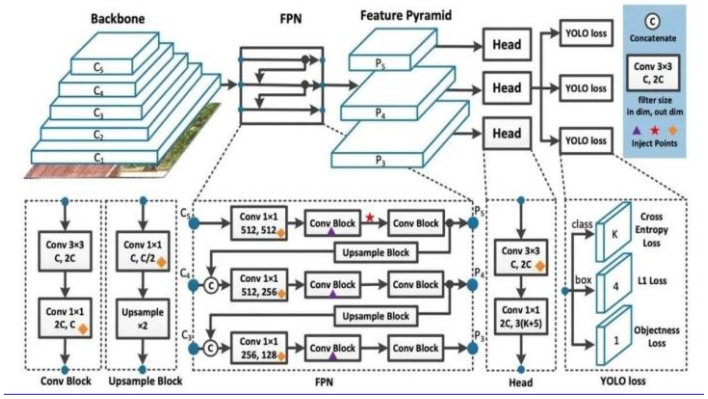

Figure 3: YOLOv8-seg Architecture Diagram

The key architectural distinction is that U-Net performs dense prediction across all pixels, while YOLO first identifies object instances and then generates masks anchored to those detections. This object-centric approach may confer advantages for sparse targets (like oil patches) where explicit localisation improves feature learning.

3.3 Evaluation Protocol

We evaluate segmentation quality using standard metrics:

- **Dice Coefficient**: Harmonic mean of precision and recall, defined as $2|X \cap Y|/(|X|+|Y|)$ where X denotes predicted pixels and Y ground truth
- **Intersection over Union (IoU)**: $|X \cap Y|/|X \cup Y|$, equivalent to Jaccard index
- **Accuracy**: Correct predictions divided by total pixels
- **Precision**: True positives divided by predicted positives

For class-imbalanced scenarios, we report both mean and median Dice scores, as the median provides robustness to outliers.

To address the operational trade-off between detection sensitivity and false-positive suppression, we implement a systematic threshold sweep procedure. For YOLOv8, we vary the detection confidence threshold $T \in [0.00, 0.95]$ in increments of 0.05 and evaluate two distinct metrics:

- **Detection Performance (oil-present frames)**: Dice coefficient quantifies segmentation quality when oil is actually present
- **False Positive Control (no-oil frames)**: Specificity measures the proportion of clean frames correctly identified as oil-free; additionally, we compute mean predicted oil pixels as a proxy for false-positive magnitude

We visualise the Dice-specificity frontier and apply a knee-point heuristic to identify a balanced operating point: we define the optimal threshold as that which maximises perpendicular distance to the line connecting the minimum-specificity/maximum-Dice point and the maximum-specificity point. This provides an objective, data-driven selection method that balances competing objectives without manual tuning.

For comparison with continuous monitoring requirements, we also identify a high-specificity operating point ($T \geq 0.80$) that prioritises false-positive suppression for scenarios where review capacity is limited.

Beyond accuracy metrics, we measure inference latency and throughput (frames per second) on a standardised hardware platform (NVIDIA Tesla T4 GPU) at both 640×640 and 512×512 resolutions. These measurements enable practitioners to project system capacity and assess deployment feasibility for real-time applications.

IV. Results

4.1 Training Convergence

Both models achieve stable convergence within 50 epochs. YOLOv8-nano demonstrates a smooth, monotonic reduction in segmentation loss (Figure 4), while U-Net exhibits more variable convergence with evidence of overfitting on the minority oil class despite the validation loss eventually stabilising (Figure 5). This divergence in convergence behaviour foreshadows the substantial performance gap observed in evaluation.

4.2 Segmentation Performance

Table 2 presents class-wise performance metrics for both architectures on the held-out test set. YOLOv8 substantially outperforms U-Net across all metrics and both classes:

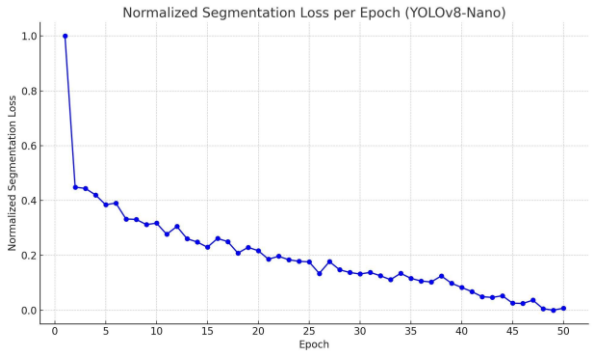

Figure 4: YOLOv8-Nano Segmentation Loss (Normalised)

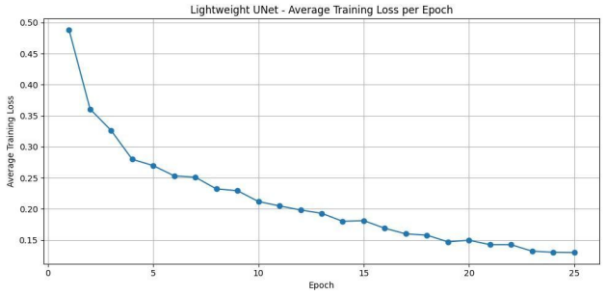

Figure 5: U-Net Binary Cross-Entropy Loss

Overall Metrics Table

Model	Class	Dice	Acc.	Precision
YOLOv8	Oil	0.7091	0.9410	0.7270
YOLOv8	Water	0.6595	0.9514	0.6709
U-Net	Oil	0.0006	0.6671	0.0197
U-Net	Water	0.3948	0.6062	0.3514

Table 2: Comparison of segmentation performance across models for each class.

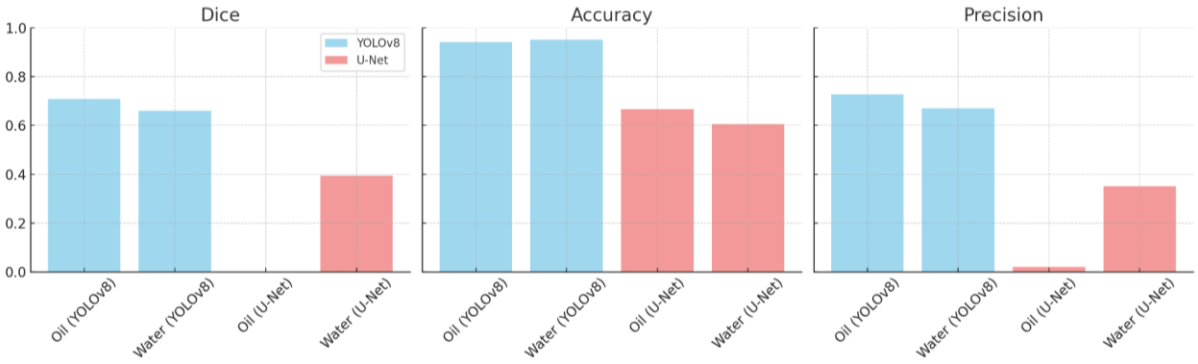

Figure 6: Segmentation performance across models and classes

The disparity is most pronounced for the minority oil class: YOLOv8 achieves Dice coefficient of 0.71 while U-Net effectively fails to learn oil segmentation (Dice = 0.0006). This outcome indicates that under identical training conditions, U-Net's dense prediction paradigm without explicit localisation supervision proves insufficient for learning discriminative features for rare, low-contrast targets.

For the water class, YOLOv8 again demonstrates superior performance (Dice = 0.66 vs. 0.39), though the gap is less extreme. This suggests that U-Net can learn features for high-prevalence classes but struggles with class imbalance, a known limitation of unweighted pixel-wise losses in semantic segmentation.

4.3 Computational Efficiency Analysis

Table 3 characterises the speed-accuracy trade-off across input resolutions:

Input	FPS	Latency (ms)	Oil Dice
640×640	16.74	59.74	0.7192
512×512	25.50	39.21	0.6597

Table 3: Speed/accuracy trade-off on Tesla T4 (batch=1, conf=0.25, IoU=0.7, mask thr=0.5).

Reducing input resolution from 640 to 512 pixels yields a 52% throughput improvement (25.5 vs. 16.7 FPS) at the cost of 8.3% relative decrease in oil segmentation quality. This trade-off curve suggests two deployment strategies: (1) 640×640 for offline incident analysis, where accuracy dominates; (2) 512×512 for continuous monitoring, where real-time processing is required and marginal accuracy loss is acceptable.

For perspective, at 512×512 resolution, the system can process approximately 92,000 frames per hour on a single GPU, sufficient for parallel monitoring of multiple camera feeds or rapid batch processing of archived footage.

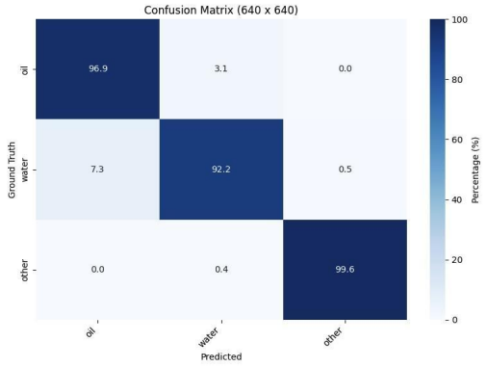

Figure 7: Confusion Matrix for 640 x 640

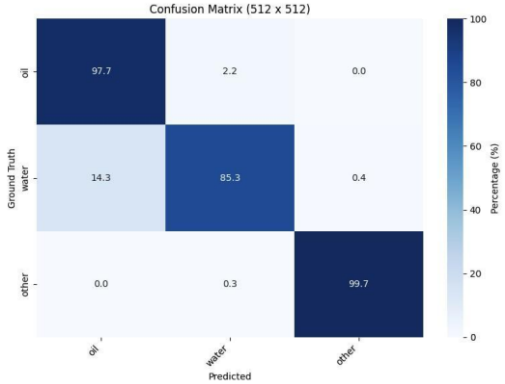

Figure 8: Confusion Matrix for 512 x 512

No-Oil Frames: Specificity and False-Positive Magnitude (Operational View)

4.4 Error Structure and Confusion Analysis

Normalised confusion matrices (Figures 7–8) reveal the dominant error modes. At 640×640 resolution, the primary misclassification pattern is water → oil false positives, while oil → water false negatives are relatively rare. This asymmetry has important operational implications: the system is more likely to generate false alerts on clean water than to miss actual spills.

This error structure motivates separate evaluation of oil-present and no-oil frames, as aggregate metrics would obscure the distinction between detection sensitivity (performance when oil is present) and specificity (correct rejection of clean-water frames).

4.5 Operating Point Selection

Figure 9 presents the Dice–specificity frontier across confidence thresholds. The curve exhibits a broad plateau in Dice coefficient up to $T \approx 0.60$, followed by gradual degradation as the threshold increases. Specificity on no-oil frames improves monotonically with threshold, approaching 100% at $T > 0.90$.

Applying the knee-point heuristic identifies T = 0.35 as the balanced operating point. At this threshold:

- Oil-present frames: Dice = 0.906 ± 0.145, Median = 0.948
- No-oil frames: Specificity = 96.15%
- False positive magnitude: Mean 61,900 predicted oil pixels when errors occur

For high-specificity operation, T = 0.85 yields:

- Oil-present frames: Dice = 0.772
- No-oil frames: Specificity = 98.08%

Table 4 summarises these operating points:

Mode	Threshold	Dice (oil-present)	Specificity (no-oil)
Balanced (knee)	0.35	0.906	96.15%
High-specificity	0.85	0.772	98.08%

Table 4: Operating points selected from the threshold sweep.

The balanced mode provides high segmentation quality while maintaining acceptable false-positive control for incident investigation and post-event analysis. The high-specificity mode trades a 13% relative decrease in Dice for a substantial reduction in false alerts, appropriate for continuous monitoring where human review capacity is limited.

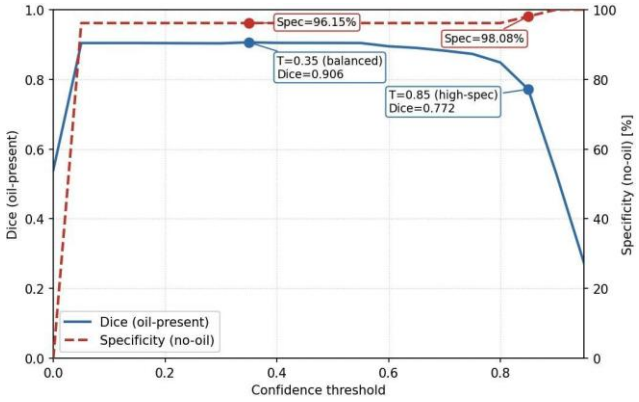

Figure 9: Dice Specificity vs Confidence Thresholds

4.6 Comparison with Prior Work

The original dataset paper employed a U-Net with a pretrained EfficientNet-B4 encoder (>40M parameters) and reported oil-class Dice of 0.74. Our YOLOv8-nano achieves a Dice of 0.71 with only 3.7M parameters (10× reduction) and inference latency of 16.7ms versus 80-100ms for the original model (5× speedup).

Model	Parameters (M)	Inference Time (ms)
U-Net + EfficientNet	40+	80–100
YOLOv8-Nano (ours)	3.7	16.7

Table 5: Model complexity and inference latency comparison at 640×640 resolution on an NVIDIA Tesla T4 GPU.

This result demonstrates that lightweight architectures can achieve near-state-of-the-art detection performance at dramatically reduced computational cost, making edge deployment economically feasible and reducing total cost of ownership for operational monitoring systems.

V. Discussion

5.1 Economic Interpretation of Results

The primary finding, that lightweight architectures achieve operationally relevant detection thresholds, has direct economic implications for monitoring system deployment. Traditional approaches impose high variable costs (patrol hours, inspector salaries) while providing limited coverage. Satellite-based monitoring reduces variable costs but imposes substantial capital costs and analytical overhead. Edge-deployed AI systems offer a third paradigm: moderate fixed costs (hardware, training) with near-zero marginal cost per additional frame analysed.

Under this cost structure, the optimal deployment strategy shifts from intensive but sparse monitoring (e.g., scheduled patrols) toward extensive continuous monitoring, fundamentally altering the detection probability frontier. Our results indicate that this shift is technically feasible: a single GPU can process multiple camera feeds in real-time, and the accuracy-threshold trade-off enables explicit calibration to match operational requirements.

5.2 Class Imbalance and Model Selection

The dramatic performance gap between YOLOv8 and our U-Net baseline (Dice 0.71 vs. 0.0006 on oil) primarily reflects differential resilience to class imbalance. YOLO's object-detection foundation provides explicit localisation supervision that helps the model learn representations for rare targets. In contrast, pixel-wise cross-entropy without class weighting or hard negative mining allows the model to achieve low loss by simply predicting the majority class.

This finding aligns with the broader literature on imbalanced learning: when positive samples are scarce, architectures with explicit instance-level supervision tend to outperform pure dense prediction. For practitioners,

this suggests that instance segmentation should be the default paradigm for maritime monitoring tasks where target classes (spills) are inherently rare relative to background (clean water).

5.3 Operational Decision Framework

The threshold-sweeping procedure and dual operating points provide a principled approach to system configuration. The key insight is that different deployment contexts impose different cost functions:

Incident Investigation: High Dice coefficient is paramount; false positives impose minimal cost (human analyst reviews all detections anyway). Use balanced operating point (T = 0.35).

Continuous Monitoring: False positives generate review burden and alert fatigue; missing spills entirely is catastrophic. Use high-specificity mode (T = 0.85) to reduce the false-positive rate while maintaining reasonable sensitivity.

This framing transforms model selection from a pure accuracy maximisation problem into an explicit cost-benefit optimisation aligned with operational objectives.

5.4 Deployment Considerations for the UAE Context

The UAE's maritime environment presents specific challenges and opportunities:

1. **Desalination Dependencies**: Coastal desalination plants create acute sensitivity to spill events. High detection sensitivity is valuable not only for environmental protection but also for water security.
2. **Infrastructure Density**: Port and near-shore environments generate substantial "look-alike" features (wakes, shadows, infrastructure). The model's performance on this dataset suggests robustness to these confounders, though offshore validation is required.
3. **Policy Alignment**: The UAE's Net Zero 2050 initiative and ADNOC's ENERGY AI program provide institutional support for AI-enabled environmental monitoring. Our results demonstrate technical feasibility for one component of this broader ecosystem.

From an economic perspective, the marginal cost of adding monitoring capacity is low once initial infrastructure is deployed. This suggests that the optimal strategy is extensive coverage across all critical zones rather than intensive monitoring of select locations.

Figure 10: Oil class segmentation. Ground Truth (Green), YOLOv8 Prediction (Red), and U-Net Prediction (Blue).

Figure 11: Water class segmentation. Ground Truth (Green), YOLOv8 Prediction (Red) and U-Net Prediction (Blue).

5.5 Limitations and Boundary Conditions

Several factors constrain the generalizability of these findings:

1. **Dataset Scope**: Training data derive exclusively from port environments. Offshore conditions (heavy seas, strong specular reflection, extreme weather) may alter model performance unpredictably. Transfer learning or domain adaptation may be required for broader deployment.
2. **Temporal Modelling**: All analyses assume independent frames. Operational systems would benefit from short-horizon temporal smoothing to suppress transient false positives and stabilise boundaries across sequential frames.
3. **Baseline Strength**: Our U-Net implementation intentionally omits pretraining, attention, and class balancing to isolate architectural effects. Stronger baselines would narrow the performance gap but would also

increase computational cost, potentially negating the efficiency advantages that motivate lightweight architectures.

4. **Operating Point Robustness**: Threshold selection reflects this dataset's class distribution and error characteristics. Different camera systems, lighting conditions, or geographic regions may require recalibration.

VI. Conclusion

This study demonstrates that lightweight instance segmentation architectures achieve operationally relevant oil spill detection performance while maintaining computational efficiency suitable for edge deployment. YOLOv8-nano substantially outperforms a classical U-Net baseline (oil-class Dice 0.71 vs. 0.0006) and approaches the performance of models with 10× more parameters while offering 5× faster inference.

Beyond point estimates of accuracy, we develop an operational framework for threshold selection that enables practitioners to explicitly balance detection completeness against false-positive suppression, matching system behaviour to deployment context. This operational perspective reflects the economic reality that different monitoring scenarios impose different cost structures.

For maritime infrastructure in the UAE and similar high-risk regions, these results support the technical and economic feasibility of AI-enabled continuous monitoring. While limitations remain, the path forward is clear: lightweight models can deliver accurate real-time segmentation, and systematic operating-point selection can align system behaviour with operational objectives and economic constraints.

Future work should focus on temporal modelling, offshore validation, and integration with existing surveillance infrastructure to transition these methods from research demonstration to operational deployment. Ultimately, the convergence of environmental policy objectives (Net Zero 2050), regional economic priorities (energy sector sustainability), and technical capability (efficient AI architectures) creates a favourable environment for the adoption of automated monitoring systems that reduce both environmental damage and economic losses from maritime oil spills.

Bibliography

[1]. ADNOC launches AI program for sustainability: Energy AI [Internet]. 2023 [cited 2024 Jul 1]. Available from: https://www.adnoc.ae/en/news-and-media/newsarchive/2023/energy-ai
[2]. Chen LC, Zhu Y, Papandreou G, Schroff F, Adam H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV). 2018. p. 801–818.
[3]. De Kerf H, Deforche W, Verstockt S. Oil spill segmentation in port environments: A dataset and benchmark. Sci Data. 2024;11:345.
[4]. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L. ImageNet: A large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2009. p. 248–255.
[5]. Jiang Z, Zhang X, Wang Y, Li D. Deep learning methods for oil spill detection: a review. ISPRS J Photogramm Remote Sens. 2022;185:1–15.
[6]. Jocher G, Chaurasia A, Qiu J, Stoken A. YOLOv8: ultralytics official release [Internet]. 2023 [cited 2024 Jul 1]. Available from: https://github.com/ultralytics/ultralytics
[7]. Li X, He M, Wang Y, Guo Y, Garcia-Pineda O, MacDonald I. Discrimination of oil spills and look-alikes using convolutional neural networks. Remote Sens Environ. 2019;221:331–343.
[8]. Ma L, Li Z, Luo W, Xu X, Chen W. Deep learning in environmental monitoring: Recent advances and future directions. Environ Model Softw. 2023;160:105576.
[9]. Nguyen M, Ma L, Xie E, Zhao X. AI-enabled remote sensing for environmental monitoring: A review. Sci Rep. 2022;12:12345.
[10]. Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention (MICCAI). Springer; 2015. p. 234–241.
[11]. Solberg R, Brekke C, Husoy T. Remote sensing of oil spills: Recent advances and future perspectives. Remote Sens Environ. 2010;100(1):1–23.
[12]. Sun Y, Chen Y, Guo Y. Deep learning for oil spill detection from remote sensing images. Remote Sens. 2019;11(22):2585.
[13]. Tan M, Le QV. EfficientNet: Rethinking model scaling for convolutional neural networks. In: Proceedings of the 36th International Conference on Machine Learning (ICML). 2019;97:6105–6114.
[14]. Topouzelis K. Oil spill detection by SAR images: dark formation detection, feature extraction and classification algorithms. Sensors. 2008;8(10):6642–6659.
[15]. UAE Net Zero by 2050 Strategic Initiative [Internet]. 2021 [cited 2024 Jul 1]. Available from: https://u.ae/en/about-the-uae/strategies-initiatives-and-awards/strategies-plans-and-visions/environment-and-energy/net-zero-by-2050
[16]. Xie E, Wang X, Wei Z, Wang Y, Li H. SegFormer: Simple and efficient design for semantic segmentation with transformers. Adv Neural Inf Process Syst (NeurIPS). 2021;34:12077–12090.
[17]. Zhao X, Ma L, Xie E, Wang X, Luo W. Deep learning for maritime object detection: A review. Inf Fusion. 2022;82:114–136.