Real-Time Vessel Identification on Edge Devices

Dominique Heller, Cédric Seguin, Paul Gautier, and Johann Laurent
Université Bretagne-Sud, CNRS-UMR 6285 Lab-STICC, Lorient, France
{firstname}.{lastname}@univ-ubs.fr

***Abstract*—Maritime surveillance plays a critical role in ensuring the security and safety of coastal regions, demanding reliable identification and tracking of vessels across spatially distributed camera networks. This task is particularly challenging as the system must identify a vessel even if it appears in different lighting conditions, angles, or environments. In this work, we propose a real-time visual vessel re-identification system optimized for deployment on edge devices. We evaluate our method on a custom maritime dataset and benchmark its performances on the NVIDIA Jetson Orin NX platform, AMD-Xilinx Kria KV260 Vision AI Kit, and a Raspberry Pi 5 with Hailo-8 accelerator. This work highlights the feasibility of deploying advanced Re-ID systems at the edge, enabling scalable, real-time maritime monitoring solutions. The proposed deployments demonstrate promising results with an inference speed of 20 FPS and a limited degradation of 3% in mean average precision in the worst case due to 8-bit quantization.**

***Index Terms*—Deep-learning, Edge computing, Marine vehicles identification**

I. Introduction

According to the United Nations Conference on Trade and Development (UNCTAD), the maritime transport sector accounts for over 80 percent of world trade volume. It is essential to guarantee the safety and security of maritime traffic from coastguard stations, Maritime Affairs ships patrols, and aircraft. The purpose is to maintain control over all activities related to the maritime environment, including commercial traffic management, sea fishing, and the monitoring of marine pollution, among others. Maritime surveillance is the process of monitoring, detecting, identifying, and tracking vessels and objects in or near a marine environment. It can be conducted using a variety of technologies and methods, including satellite imagery, automatic identification system (AIS), radars and cameras.

Currently, human analysis, assisted by simple intelligent methods, remains the most common approach for processing large-scale maritime surveillance videos. With the decreasing cost of cameras and sensors, the volume of usable data has surged, making analysis increasingly tedious for operators. This time-consuming process is also prone to errors, including missed detections due to operator fatigue. To address these challenges, research on automated identification systems has become a dynamic area of computer vision.

However, maritime environments present unique challenges, including variable weather conditions, the dynamic nature of the ocean, and the vastness of the surveillance area. These factors can decrease the quality of visual data and complicate object detection. Sensor fusion combining visual data such as camera, radar, LiDAR, infrared, or even satellite feeds, allows for better detection, even in low-visibility conditions (fog, night-time, or adverse weather), improving the robustness of maritime surveillance systems.

Additionally, deploying drones with embedded computer vision systems closer to the target area can help overcome challenges posed by the maritime environment, improve identification accuracy, and reduce the cost of maritime surveillance missions. Furthermore, an image-based classification and identification system is complementary to radar and AIS data. It helps to remove doubts about radar detections, which are limited in terms of target classification and identification due to the lack of return information (radar-equivalent surface), and to fill the gaps left by the AIS, such as jamming, shutdown or identity theft.

Re-identifying (Re-ID) vessels is a particularly important task in scenarios such as surveillance or tracking ships over time, across different images or video frames. However, these computer vision algorithms may be too resource-intensive to be embedded on edge devices, such as those on drones. Our research introduces a real-time system for re-identifying marine vessels across different images. This approach that combines a deep learning technique is based on triplet loss network, with embedding analysis, and a k-nearest neighbors algorithm (K-NN). It considers the appearance, category, and orientation of vessels, to determine whether different images represent the same object. The proposed system is designed for deployment on edge computing devices embedded in maritime and aerial vehicles, using initially an RGB camera.

The rest of the paper is organized as follows. Section II presents related work on marine vessel Re-ID, while Section III introduces a new large-scale marine vessel dataset. Section IV provides training analysis, and Section V presents inference results on edge devices. Finally, we discuss our results and future work in Section VI, followed by the conclusion in Section VII.

II. Related works

This section introduces the concept of Re-Identification (Re-ID) by briefly reviewing research on person and vehicle Re-ID, followed by an exploration of recent advancements in marine vessel detection and orientation recognition.

A. Person and Vehicle Re-Identification

Re-ID focuses on retrieving an entity of interest across multiple non-overlapping camera views. This is a challenging computer vision task, as its goal is not only to differentiate

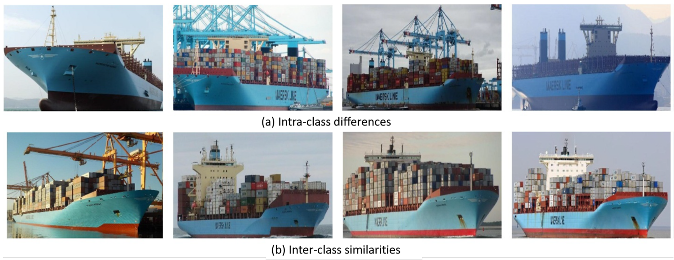

Fig. 1. Complexity of vessel Re-ID [1].

between object categories, as in classification tasks, but also to recognize the same individual objects across different images. The main challenge is to accurately associate the same entity captured under different conditions, including variations in lighting, pose, viewpoint, background, and occlusions. Re-ID is widely used in surveillance through Person Re-ID and in traffic monitoring and automated tolling through vehicle Re-ID.

The main method, in most of the literature of Re-ID, is to locate instances of a query object (probe) from a group of candidates (gallery) captured from different non-overlapping camera views. Features are extracted from each image of a person, and mathematical techniques are used to measure the distance between image pairs.

Building a Re-ID system requires five main steps: raw data collection, bounding box generation, data annotation, model training, and object retrieval.

B. Vessel Re-Identification

Compared with the re-identification of people and vehicles, vessel re-identification offers additional complexities due to several domain-specific factors, such as:

- *Small inter-class similarity*: Vessels from different classes may appear visually similar, especially when they belong to the same ship models or companies.
- *Large intra-class similarity*: The same vessel can look drastically different depending on the viewpoint, making consistent identification difficult.
- *Environmental influences*: Factors such as occlusion, illumination changes, and other environmental noise (fog, rain) further impact the vessel's appearance.

Figure 1, extracted from [1], shows a) the intra-class differences caused by viewpoint changes for the same ship, while b) highlights the inter-class similarity for different vessels of the same type.

Figure 2, shows some sample images of our dataset for different challenging scenarios: different illuminations, variation in scale, change in background, and different viewpoints.

Most of the works consider only similarities and dissimilarities to the vessel identification task, and use the TriNet model [2], or [3]. The loss function optimized to learn such features is the triplet loss. During the learning, it uses three images the anchor (current ship), the positive (another image of the current ship), and the negative (image of another ship). The TriNet

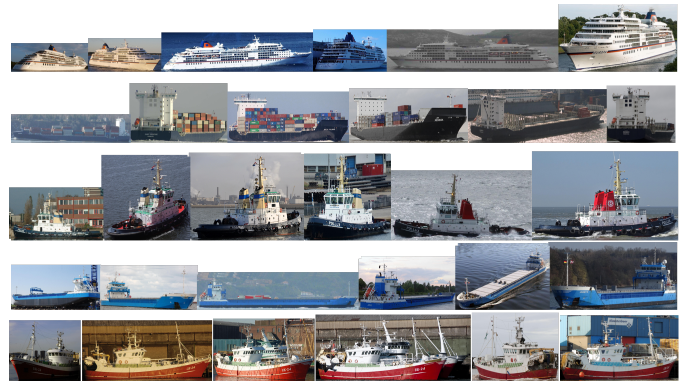

Fig. 2. Samples of our VesselReid-12k dataset. Each row of the figure shows six images of a ship in different scenarios: different Illuminations, variation in scale, change in background, and different viewpoints.

model is trained to minimize the distance between the anchor and the positive sample and to maximize the distance between the anchor and the negative sample. These three images are passed through a Convolution Neural Network (CNN) layer, generating a 1-dimensional feature vectors (embedding) which is used to calculate the distances between them. The CNN is often a customized ResNet-50 architecture that is suitable for embedded applications because of its low computational complexity. At the inference stage, only a single subnetwork is used to generate the embeddings of new input samples and a K-Nearest Neighbors (KNN) algorithm is performed to find the best matches. This approach efficiently identifies vessels based on learned visual features and ensures accurate matching through optimized distance calculations. IORNet [3] proposes an identity-oriented re-identification model that combines triplet loss and cross-entropy loss, using ResNet-50 as the core feature extraction network. [4] study the ship retrieval methods for intelligent water transportation system in smart cities and employed a pyramid structure to deal with variations in ship shapes and sizes. In [5], the authors proposed a quadruplet learning and improve the recognition accuracy taking four images : anchor, positive, negative high-similar (same class vessel) and negative. In [6] the authors proposed a new Vessel Re-ID network (VesselNet), employing ResNet-50 to extract image features and incorporating a hybrid attention module to effectively capture significant features in the images. In [7] a fine-grained feature extraction network (FGFN) is proposed. The authors improve the ResNeSt [8] architecture through incorporating a self-attention mechanism and generalized mean pooling. In [9] the authors propose a two-branch network with dynamic feature enhancement and dual attention to address the issue of low accuracy in ship Re-ID under foggy weather, enabling simultaneous learning of defogging and ship Re-ID tasks in an end-to-end manner.

Specific parts of ships, such as the bow, stern, and equipment on the deck, often possess high uniqueness and discriminability [10]. Capturing these local features allow for more accurate identifications and differentiations of ships, [11] adopts a dual-branch architecture for global and local feature learning, allowing each branch to focus independently on global or local characteristics.

The GLF-MVFL framework [2] proposes a feature learning method based on global and local fusion, combining cross-entropy loss with orientation-guided quintuplet loss. As large vessels are more sensitive to a change in viewpoint they add two extra image samples (one positive and one negative) to the triplet to constitute the quintuplet : anchor image, a positive image from the same viewpoint, a positive image from a different viewpoint, a negative image from the same viewpoint, and a negative image from a different viewpoint. Compare to IORNet [3], they increased the mean average precision (mAP) and Rank-1 by 7% and 3% with the same backbone ResNet-50 and achieved 74.9% for mAP and 61.4% for Rank-1.

In this article, we propose a large vessel Re-ID dataset called VesselReID-12k and use ResNeSt as the feature extraction network [8]. We also employ generalized mean pooling, hard triplet mining, and re-ranking optimization to achieve state-of-the-art ship re-identification results.

We will describe the large-scale dataset we constructed in the next section.

III. THE LARGE-SCALE MARINE VESSEL DATASET

Since the well-known vessel re-identification datasets VesselReid [12] and VesselReid-539 [2] are not publicly available, we created a large-scale, well-annotated dataset with rich attribute labels, including vessel identities, vessel category (36 classes based on AIS ship types), as well as the main orientations: front, back, one side, oblique front-side, and oblique back-side. The process for collecting, cleaning and annotating data is described in [13].

At the end of this process, our dataset consists of 263,488 images of 12,777 unique vessel IDs, each annotated with a 7-bin orientation and a 36-class vessel-type label.

Figure 2 shows a few representative samples from the dataset VesselReid-12k.

A. Statistical distribution of the dataset

Figure 3 statistically analyzes the VesselReid-12k dataset in terms of vessel types and orientations. This paper classifies vessel types into the thirty-six classes.

It can be noticed that there is an issue of class and orientation imbalance. In future work, we will integrate new images sourced from VesselFinder or Marine Traffic to better balance our dataset. VesselFinder and Marine Traffic are international, free-to-use websites for real-time ship tracking where each boat contains a variable number of images captured from different viewpoints and distances across different times and locations.

In addition, to analyze scale variations within our dataset, as shown in Figure 4 we calculate the mean and standard deviation of the width, the height and the aspect ratio of ship images. The vessel image size and aspect ratio vary greatly. The standard deviation of the aspect ratio (width-to-height ratio) of our dataset is 1.113, which is higher than that found in other Re-ID datasets for ships [11]. Therefore, the greater the diversity of our dataset, the less sensitive the trained model must be to scale change.

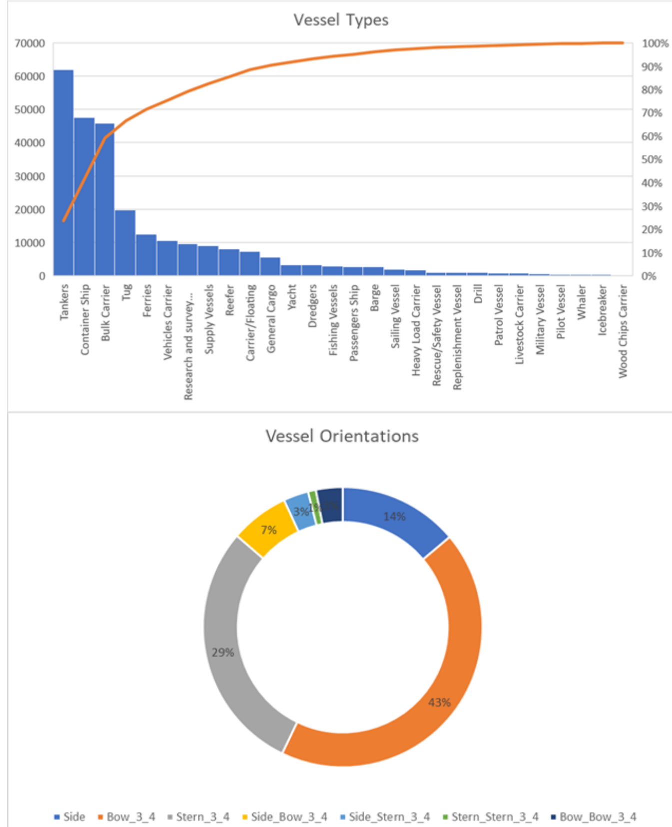

Fig. 3. VesselReid-12k representation in terms of vessel types and vessel orientations.

B. Comparison with other vessel datasets

Table I presents a comparison between our dataset and other existing vessel datasets. [2] proposed a ship retrieval dataset named VesselID-539, created by selecting images from the Marine Traffic website. The training set contains 104,554 images of 377 identities, while the testing set consists of 44,809 images of 162 identities. [12] introduced a new maritime vessel re-identification dataset named VR-VCA, which includes 729 unique identities along with 5-bin orientation and 8-class vessel-type annotations. [4] constructed a fine-grained ship retrieval dataset (FGSR), consisting of 30,000 field-captured images of 1,000 ships. The VesselID-700 dataset comprises 56,069 images covering seven typical ship classes. Additionally, the Warships-ReID dataset [14] includes 4,780 images of 163 vessels. The ShipReID-2400 dataset is compiled from a real-world intelligent waterway traffic monitoring system. It comprises 17,241 images of 2,400 distinct ship identities collected over 53 months, ensuring diversity and representativeness. Finally, CMShipRied is cross-modality ship re-identification dataset which contains visible light, near-infrared, and thermal infrared modalities collected by autonomous aerial vehicle. It consists of ten categories, about 138 identifications, and 8337 images. Compared to other

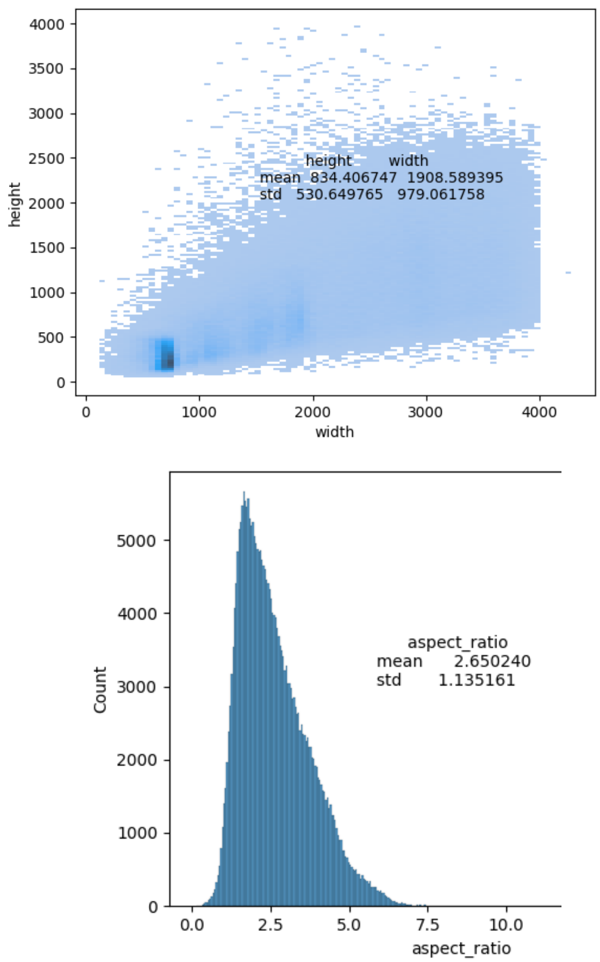

Fig. 4. Scale variation within the VesselReid-12k dataset.

vessel re-identification datasets, our dataset VesselRied-12k contains more ships, more images, and a greater diversity of viewpoints and aspect ratios.

We split our dataset into 3 parts: the training subset includes 7497 IDs and 197616 images, the query (validation) subset contains 5984 IDs with 13175 images, and the gallery (test) subset contains 7481 IDs with 52697 images. In the next section, we will detail the training and associated statistics.

IV. Training analysis

To distinguish objects based on their physical characteristics at the pixel level, an appearance descriptor called an embedding is generated using a Siamese triplet network. During training, the network uses three images: the anchor (previous detection), the positive (current and future detections of the same object), and the negative (other objects). It is trained to minimize the distance between the anchor and the positive sample, while maximizing the distance between the anchor and the negative sample. These three images are passed through a CNN layer (the backbone), generating a one-dimensional embedding, used to compute the distances. During inference, only one network is used to produce an embedding layer for the candidate track (anchor). Then results are compared to current detection to find the best match based on minimum distance.

To guide feature learning during training, the most commonly used loss functions are identity (cross entropy) loss and triplet loss. Identity loss treats the training process of the identification model as an image classification problem, where each identity ID is considerate as a different class. Triplet loss, on the other hand, considers the training process of the identification model as a retrieval ranking problem, where the distance between positive sample pairs should be smaller than between negative sample one. The choice of positive and negative sample is crucial for improving the retrieval performance of re-identification. This process of selecting effective triplets is often referred to hard mining, while re-ranking is typically applied during the inference stage to refine retrieval results based on the initial ranking.

The basic idea of re-ranking is to utilize gallery-to-gallery similarities to refine the initial ranking list. When fine-tuning with the ranking loss, it is crucial to mine hard triplets efficiently, as randomly selected triplets often result in easy samples or triplets with little, or no loss, contribution.

A. Training process

The training processes are conducted using *FastReid* framework on Nvidia Quadro RTX 3090. The framework FastReID [17] implements state-of-the-art re-identification algorithms. In the image pre-processing stage, resizing and data augmentation techniques such as flipping, cutout, and random erasing are done. The input images are downsampled to resolution of 256×256.

For the backbone that maps images to features representations, four architectures were used and compared: Resnet-50, Resnet50+IBN (with instance batch normalization), Vision Transformer (ViT) and ResNeSt. Transfer learning is adopted by initializing the models with the weight values of models previously trained on popular large-scale ImageNet dataset. The model is trained over 300 epochs.

During training, we use a combination of cross-entropy loss and triplet loss as the loss function. Following the approach in [18], we also apply hard triplet mining to enhance the discriminative power of the triplet loss and mitigate the impact of class imbalance. The aggregation layer is designed to combine the feature maps generated by the backbone into a global feature representation. At this stage, we employ average pooling. For distance metrics, we use the cosine distance, which has yielded better experimental results than the Euclidean distance on normalized embeddings.

B. Evaluation of trained models

The table II shows comparative performance evaluation of 4 backbone configurations. The key performance indicators are the ranked accuracy of re-identification and the mean average precision (mAP). Ranked accuracy is a method of computing accuracy where the top-K highest-confidence predictions are

TABLE I
COMPARISON OF PROPERTIES OF VESSEL RE-ID DATASETS

Dataset	ID Volume	Dataset Scale	Angle of View	Vessel Type
VR-VCA [12]	729	4614	5-bin orientation	8 classes
VesselID-700 [7]	700	56069	5-bin orientation	7 classes
VesselReID [15]	1248	30587	5-bin orientation	7 classes
FGSR [4]	1000	30000	2 cameras view points	none
VeRiS [6]	2,904	150,623	5-bin orientation	7 classes
VesselID-539 [2]	539	149465	superstructure	8 classes
Warships-ReID [2]	169	4780	none	8 classes
VesselReID-1656 [1]	1656	135866	5-bin orientation	12 classes
ShipReID-2400 [11]	2400	17241	8 cameras view points	none
CMShipReID [16]	138	8337	3 image source (VIS,NIR, TIR)	10 classes
Ours (VesselReID-12k)	12777	263488	7-bin orientation	36 classes

TABLE II
TRAINING RESULTS

Model	Feature dimension (Bits)	mAP (%)	Rank-1 accuracy (%)	Rank-5 accuracy (%)	Rank-10 accuracy (%)
Resnet-50	2048	67.05	72.32	89.78	94.89
Resnet-50+ibn	2048	69.66	74.51	90.86	95.60
Vit	768	69.07	73.21	90.89	95.96
ResNeSt	2048	85.19	88.58	96.10	97.90

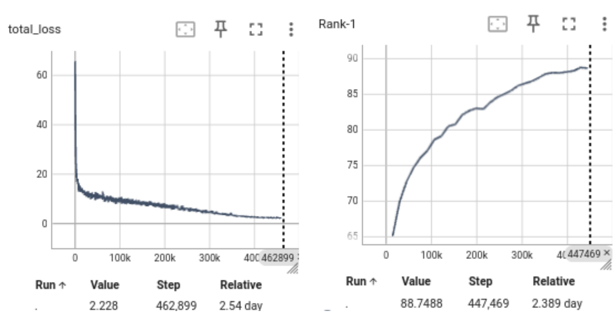

Fig. 5. Training results of ResNeSt configuration: Total (Triplet+Class) loss and rank-1.

compared to the ground truth label. In our case, we compute rank-1 (Figure 5), rank-5 and rank-10 accuracies. This means that for rank-10, if the ground truth label appears among the top-10 predicted labels for a given sample, it is considered as a correct match. The mAP measures the overall prediction accuracy, reflecting how well the model retrieves correct identities. In our case, it evaluates how accurately the model predicts the identity (ID) of a vessel.

The ResNeSt configuration outperforms all other configurations by up to 15% in mean average precision (mAP). It achieves outstanding performance, with a Rank-1 accuracy of 88.58% and an mAP of 85.19%, reaching state-of-the-art levels for vessel Re-ID. In comparison, MVR-net [12] yields a 74.5% mAP and a 77.9% Rank-1 score, while the GLF-MVFL framework [2] achieves 74.9% mAP and 61.4% Rank-1 accuracy.

Training the same ResNeSt configuration on the Market1501 dataset for person Re-ID achieves a Rank-1 accuracy of 95.2% and an mAP of 88.7%. While this performance is slightly better, the complex observation conditions in vessel Re-ID—such as long-range views, foggy skies, and sea reflections—along with the significant variation in vessel sizes and shapes across different viewpoints, reduce re-identification performance compared to pedestrians or vehicles. Additionally, vessels exhibit varying degrees of tilt and different draft depths due to differing loads, further complicating consistent feature extraction.

Fig. 6. Illustrations of the top-10 ranking list for retrieval results

We provide representative visualization results to intuitively demonstrate the accuracy of our vessel re-identification model in Figure 6. The left panel shows the query input, while the right panel displays the top-10 retrieved results sorted by similarity. Green boxes indicate correct ID matches, whereas red boxes represent inconsistent re-identification results. In the case of an easy sample (a tanker), the model not only retrieves the correct ID from the gallery set but also ranks it highly in the results. For a hard sample (a tugboat), our model successfully retrieves the correct ID three times, including a correct match at the top-1 rank.

V. Inference on Edge Device

Deep learning-based object detection on embedded systems must be optimized for low latency, high detection accuracy, and low power consumption. In general, the deployment process comprises two stages. In the first stage, the weights and/or activations are quantized to the desired bit-width and representation (e.g., FP16 or INT8). Quantization refers to the process of converting the weights and activations of a trained deep learning model from high-precision floating-point numbers (e.g., 32-bit) to lower-precision fixed-point or integer representations (e.g., 8-bit). This is typically done using a heuristic method that leverages a selected subset of images from the training dataset, commonly referred to the golden reference pool. This reduction in precision reduces the memory and computation requirements, making it possible to efficiently deploy neural networks on hardware with limited resources.

In the second stage, the quantized model is compiled to generate the instruction sequence. During this stage, the model is further optimized based on the target device's architecture. This study considers three different platforms: one GPU-based, one ASIC-based and one FPGA-based. Their hardware specifications are provided in Table III, which presents the technical details of the embedded edge devices targeted in this work.

A. Deployment targeting Nvidia Jetson Orin Nx

The trained model is deployed using TensorRT to achieve lower latency and higher throughput during inference on NVIDIA platforms. TensorRT is a software development kit (SDK) provided by NVIDIA for high-performance deep learning inference. It is compatible with most deep learning frameworks and is used to achieve high performance and platform portability. It comprises an inference optimizer that implements several techniques, such as kernel fusion, precision calibration, kernel auto-tuning, dynamic tensor memory management, and multi-stream execution, to optimize the inference of the trained model.

Since the *FastReID* framework is not supported by TensorRT, the target model is first converted using the Open Neural Network Exchange (ONNX) format. Next, the model is quantized to FP16 representation. Table IV presents the results in terms of detection performance and inference speed. It compares the original trained model with the TensorRT-converted model when deployed on the Jetson Orin NX. The comparison shows that using TensorRT increases the inference rate, with only a slight degradation in mAP (2.6 %).

TABLE III
Specifications of target edge embedded devices

Target Device	Nvidia Jetson Orin NX 8Go	PI5 + M2 Hailo 8	Kria KV260 Vision AI Kit
Edge accelerator	1024-core 32 Tensor Cores	Hailo 8L	1x DPU configurations B4096 at 300 MHz
AI Performance (estimated FP16)	54 TFLOPS	Unkown	Unkown
AI Performance (estimated INT8)	70 TOPS	13 TOPS	1.43 TOPS
Max Power consumption	30 W	12+3 W	8 W
Price	600$	300$	300$

TABLE IV
Obtained results on Jetson Orin NX

Network	ResNeSt	
Input Resolution	256×256	
Model	Original	FP16
Mean Average Precision	0.852	0.826
FPS	15.7	19.9

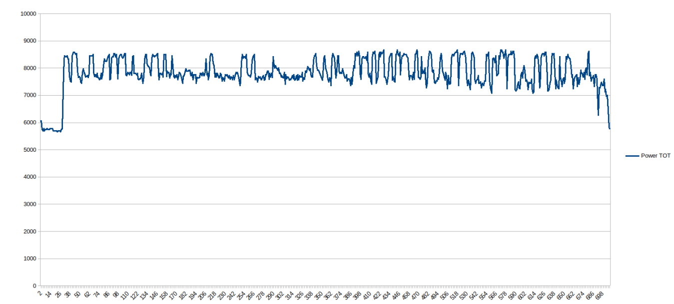

Fig. 7. Power Consumption during inference on query subset on Nvidia Jetson Orin NX.

As the Jetson board incorporates a power monitor, we recorded power consumption during inference on the query subset (5984 IDs with 13175 images) as shown in Figure 7 and average power consumption is around 8 watts for 20 fps, or 2.5 FPS/watt.

B. Deployment targeting RPI5+Hailo 8l NPU

The ResNeSt model is not supported by Hailo's dataflow compiler. Therefore, we used the basic Resnet-50 model for inference. Since one operation (pow 3) is not supported in the aggregation stage either, only the backbone (ResNet-50) was accelerated with the Hailo 8L NPU. The quantization performed by Hailo is a mix of FP16 and INT8 depending on the available resources. After compilation, 60% of the computation resources and 66% of the memory resources of the Hailo8L ASIC are used. The quantization steps results in a 2% loss mAP precision.

The Resnet-50 backbone alone runs at 23 fps. However, with the pre-processing and post-processing code (Head part with GlobalAveragePooling) running on the CPU, the framerate drops to 12 fps. Dispite this, the inference time is sufficient to track ships travelling at typical nautical speeds. Without equipment to measure the power consumption of the Raspberry Pi M.2 HAT, we were unable to determine the power consumption of the Hailo NPU.

C. Deployment targeting Xilinx-AMD Kria KV260 AI Vision Kit

Like the Hailo8 hardware target, only the backbone Resnet-50 model is supported by the framework Vitis AI as clamp and pow operations of GeneralizedMeanPooling layer (Head part) are not supported by the DPU. Also, due to data layout difference between Pytorch training('NCHW') and XIR

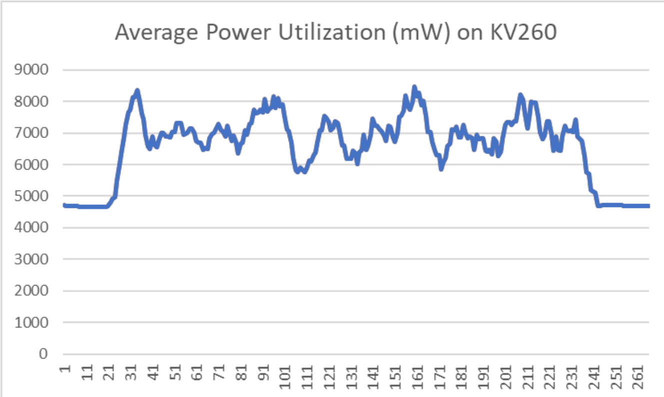

Fig. 8. Average Power utilization (mW) for inference on Xilinx KV260.

DPU('NHWC'), a permution is done on inputs. Using Vitis AI toolset version 3.5, the Resnet-50 model is quantized (Post Training Quantization) into INT8 representation and then compiled targeting DPUCZDX8G architecture. The post training quantization step results in a 3.27% loss mAP precision using the query subset of VesselReid-12K as calibration dataset.

The Resnet-50 backbone alone runs at 24 fps. We recorded power consumption during inference on a subset of the query part (5000 images) as shown in Figure 8 and average power consumption is around 6.8 watts for 24 fps or 3.5 FPS/Watt..

D. Analysis

The Jetson Orin NX is the more powerful device of the three platforms in terms of detection performance (speed and accuracy). It achieves the highest inference speed while maintaining high accuracy with the best vessel Re-ID model, ResNeSt. However, it requires a higher power budget. The Hailo and Xilinx platforms suffer from the limitations of their data flow compilers, which do not support all Re-ID models and require longer development times. Considering performance per watt criterion only, the KRIA KV260 kit outperforms the Jetson Nvidia devices when running Resnet-50 backbone.

VI. Discussions and Future Works

To improve the performance of the appearance descriptor, we plan to build on the approach of [5] and [2] by implementing an enhanced loss function strategy. As depicted in fig 9, we will first introduce the class feature into a quadruplet loss function: an image of another vessel from the same category is added as a second negative sample. This helps to further differentiate intra-class variations. Then, we will develop an orientation-guided quintuplet loss that comprises five images: the anchor image, a positive image from the same viewpoint, another positive image from a different viewpoint, a negative image from the same viewpoint and same category, and a final negative image from a different viewpoint and different category. This loss function is designed to create more robust and discriminative feature representations by considering multiple contextual aspects of the images. To enhance the performance in ship Re-ID in foggy weather, we will study the

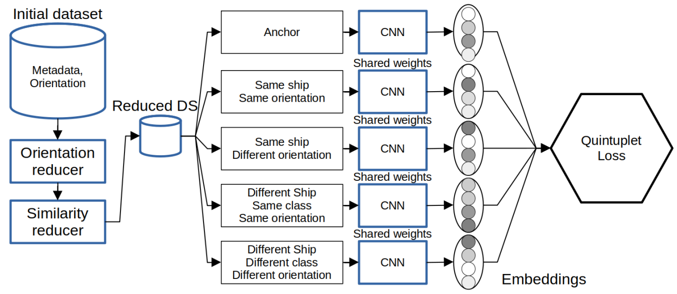

Fig. 9. Training phase

method described in [9] that proposes a two-branch network enabling simultaneous learning of defogging and ship Re-ID tasks in an end-to-end manner. In addition, an atmospheric scattering model [19] is employed for the synthesis of foggy ship images. Quantization usually results in a loss of accuracy due to information lost during the quantization process. For FPGA target, we will use QAT (Quantization-Aware Training) to improve the accuracy of quantization. Finally, Since ships are likely to have distinctive identifying features, such as flags and printed names, our future research will focus on employing a text detector and optical character recognition (OCR). Additionally, as our labeled data have a category field similar to the AIS ship field, we plan to develop a data fusion approach that combines visual detection techniques with Automatic Identification System (AIS) data [20]. Our previous work has already explored real-time classification [21], as well as the fusion of data from cameras, radars, and AIS for classification, situational awareness, and collision avoidance [22].

VII. Conclusion

This paper tackles the topic of Vessel re-identification using deep learning techniques on embedded edge devices. Vessel Re-ID is a challenging task that require to consistently recognize the same boat across different time period contexts and camera viewpoints, despite significant extra-class variations, and intra-class variations caused by changes in viewing angles, illumination conditions, image resolution, occlusions, and appearance alterations such as color shifts. Moreover, maritime environment faces unique challenges, such as changing weather, moving sea conditions, and large monitoring areas. These factors can reduce the quality of visual data and make object detection more difficult. To address the challenges of maritime surveillance, we proposed a real-time solution to vessel Re-ID for maritime surveillance on edge devices. The solution can be carried on drones, equipped with computer vision systems, positioned as close as possible to the target areas. Furthermore, integrating image-based classification and identification with radar and AIS data can enhance the accuracy of vessel identification. We constructed a specialized large-scale dataset for marine vessels, comprising 263,488 images associated with 12,777 unique vessel identities. Each image is annotated with a 7-bin orientation label and categorized into one of 36 vessel-type classes. This optimized dataset is employed to train a Siamese Triplet Network that learns to generate distinctive high-dimensional embeddings, enabling the computation of similarity distances between entities. During inference, a single network is used to produce an embedding for a candidate track, which is then compared against current detections. The best match corresponds to the detection with the smallest embedding distance. The deployment of the trained models on recent edge devices is considered. We evaluated our solution on a Jetson ORIN Nx, a Raspberry Pi 5 equipped with a Hailo-8 accelerator and the AMD-Xilinx Kria KV260 Vision AI Kit. The results demonstrate that the vessel re-identification system is capable of successfully identifying ships moving at typical maritime speeds with low power consumption. For example, 20 FPS inference speed is achieved on Jetson Orin NX with a mean average precision of 82.6% and average power consumption is around 8 watts. On an embedded system, the choice of hardware target will depend on weight and power consumption constraints. For a system with severe constraints, such as a small flying drone, the best solutions are the Xilinx Kria KV 260 or Hailo-8 accelerator. But for a unmanned surface vehicle, the best solution is the Nvidia Jetson board. It offers greater flexibility, shorter development times, better performance and reasonable power consumption of 8 W.

References

[1] Z. Lu, L. Sun, P. Lv, J. Hao, B. Tang, and X. Chen, "A new large-scale dataset for marine vessel re-identification based on swin transformer network in ocean surveillance scenario," *IET Computer Vision*, vol. 19, no. 1, p. e70007, 2025.

[2] D. Qiao, G. Liu, F. Dong, S.-X. Jiang, and L. Dai, "Marine vessel re-identification: A large-scale dataset and global-and-local fusion-based discriminative feature learning," *IEEE Access*, vol. 8, pp. 27 744–27 756, 2020.

[3] A. Ghahremani, Y. Kong, E. Bondarev, and P. H. de With, "Towards parameter-optimized vessel re-identification based on iornet," in *Computational Science–ICCS 2019: 19th International Conference, Faro, Portugal, June 12–14, 2019, Proceedings, Part V 19*. Springer, 2019, pp. 125–136.

[4] Y. Xian, J. Xian, L. Lu, and J. Tang, "Fgsr: A fine-grained ship retrieval dataset and method in smart cities," *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, p. 2807139, 2022.

[5] B. Solmaz, E. Gundogdu, V. Yucesoy, A. Koc, and A. A. Alatan, "Fine-grained recognition of maritime vessels and land vehicles by deep feature embedding," *IET Computer Vision*, vol. 12, no. 8, pp. 1121–1132, 2018.

[6] Z. Yu, J. Liu, S. Zou, and Y. Cao, "Vesselnet: A large-scale dataset and efficient mixed attention network for vessel re-identification," in *2023 2nd International Conference on Machine Learning, Cloud Computing and Intelligent Mining (MLCCIM)*. IEEE, 2023, pp. 437–441.

[7] W. Dou, L. Zhu, Y. Wang, and S. Wang, "Research on key technology of ship re-identification based on the usv-uav collaboration," *Drones*, vol. 7, no. 9, p. 590, 2023.

[8] H. Zhang, C. Wu, Z. Zhang, Y. Zhu, H. Lin, Z. Zhang, Y. Sun, T. He, J. Mueller, R. Manmatha *et al.*, "Resnest: Split-attention networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 2736–2746.

[9] W. Sun, F. Guan, X. Zhang, X. Shen, and K. Wang, "Ship re-identification in foggy weather: A two-branch network with dynamic feature enhancement and dual attention," *Engineering Applications of Artificial Intelligence*, vol. 143, p. 109974, 2025.

[10] M. Zhang, Q. Zhang, R. Song, P. L. Rosin, and W. Zhang, "Ship landmark: An informative ship image annotation and its applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 11, pp. 17 778–17 793, 2024.

[11] B. Liu, R. Huang, X. Pan, C. Li, J. Sun, J. Dong, and X. Wang, "Advancing ship re-identification in the wild: The shipreid-2400 benchmark dataset and d2internet baseline method," in *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '25. New York, NY, USA: Association for Computing Machinery, 2025, p. 106–115. [Online]. Available: https://doi.org/10.1145/3726302.3729892

[12] A. Ghahremani, T. Alkanat, E. Bondarev, and P. H. de With, "Maritime vessel re-identification: novel vr-vca dataset and a multi-branch architecture mvr-net," *Machine Vision and Applications*, vol. 32, pp. 1–14, 2021.

[13] C. Seguin, D. Heller, T. Marques, and J. Laurent, "Real-time visual vessel re-identification for maritime surveillance on edge devices," in *OCEANS 2025 Brest*, 2025, pp. 1–7.

[14] G. Zeng, R. Wang, W. Yu, A. Lin, H. Li, and Y. Shang, "A transfer learning-based approach to maritime warships re-identification," *Engineering Applications of Artificial Intelligence*, vol. 125, p. 106696, 2023.

[15] Q. Zhang, M. Zhang, J. Liu, X. He, R. Song, and W. Zhang, "Unsupervised maritime vessel re-identification with multi-level contrastive learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 5, pp. 5406–5418, 2023.

[16] C. Xu, L. Gao, Y. Liu, Q. Zhang, N. Su, S. Zhang, T. Li, and X. Zheng, "Cmshipreid: A cross-modality ship dataset for the reidentification task," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 18, pp. 10 503–10 513, 2025.

[17] L. He, X. Liao, W. Liu, X. Liu, P. Cheng, and T. Mei, "Fastreid: A pytorch toolbox for general instance re-identification," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 9664–9667.

[18] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," *arXiv preprint arXiv:1703.07737*, 2017.

[19] S. Liang, X. Liu, Z. Yang, M. Liu, and Y. Yin, "Offshore ship detection in foggy weather based on improved yolov8," *Journal of Marine Science and Engineering*, vol. 12, no. 9, 2024. [Online]. Available: https://www.mdpi.com/2077-1312/12/9/1641

[20] Y. Lu, H. Ma, E. Smart, B. Vuksanovic, J. Chiverton, S. R. Prabhu, M. Glaister, E. Dunston, and C. Hancock, "Fusion of camera-based vessel detection and ais for maritime surveillance," in *2021 26th International Conference on Automation and Computing (ICAC)*, 2021, pp. 1–6.

[21] D. Heller, M. Rizk, R. Douguet, A. Baghdadi, and J.-P. Diguet, "Marine objects detection using deep learning on embedded edge devices," in *2022 IEEE International Workshop on Rapid System Prototyping (RSP)*. IEEE, 2022, pp. 1–7.

[22] R. Douguet, D. Heller, and J. Laurent, "Multimodal perception for obstacle detection for flying boats-unmanned surface vehicle (usv)," in *OCEANS 2023-Limerick*. IEEE, 2023, pp. 1–8.