ENGINEERING OF A HIGH-THROUGHPUT MULTI-CAMERA SCALE-INVARIANT COLLISION AVOIDANCE PIPELINE

Alexander Klein, Aaron Lye, Jannis Stoppe

ALEXANDER.KLEIN@DLR.DE, AARON.LYE@DLR.DE, JANNIS.STOPPE@DLR.DE

DLR

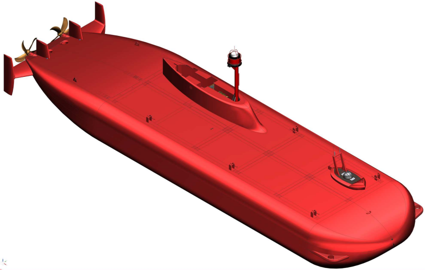
PROBLEM STATEMENT

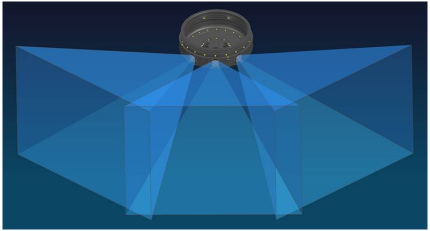
Motivation

To navigate safely in complex maritime environments, autonomous surface vessels require a **fast, high-resolution optical collision avoidance system**, capable of detecting objects across a wide range of sizes and distances, optimized for **minimal delay** and **maximal throughput** on the target hardware, guaranteeing fast response times.

Hardware Setup - Perception System

- Deployment of 3x **high-resolution cameras** with precise **temporal synchronization**.
- **Centralized computing architecture** equipped with a high-performance GPU.
- Stateless platform-independent **API** to enable seamless integration with the vessel’s software infrastructure.





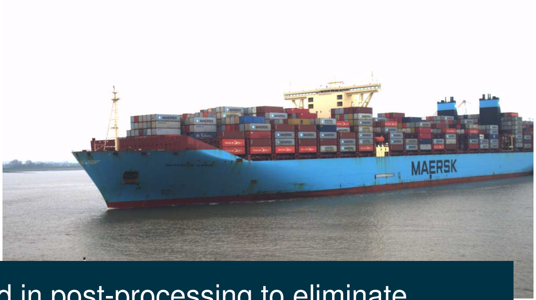
Gefördert durch:
Bundesministerium für Wirtschaft und Energie
aufgrund eines Beschlusses des Deutschen Bundestages

thyssenkrupp
EvoLogics®
Technische Universität Berlin
Universität Rostock
Traditio et Innovatio

Object Detection

Identify and localize obstacles in a given view for objects at varying distances and scales.

- Extract information based on patterns to compute predictions
- Shifting patterns can cause- assignment problems

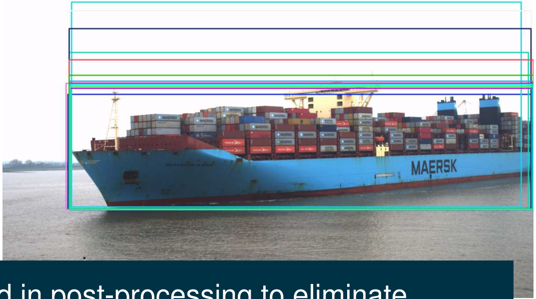


- Non-Maximum Suppression (NMS) is used in post-processing to eliminate duplicate predictions.
- NMS is not part of the model’s computational graph.

Object Detection

Identify and localize obstacles in a given view for objects at varying distances and scales.

- Extract information based on patterns to compute predictions
- Shifting patterns can cause- assignment problems

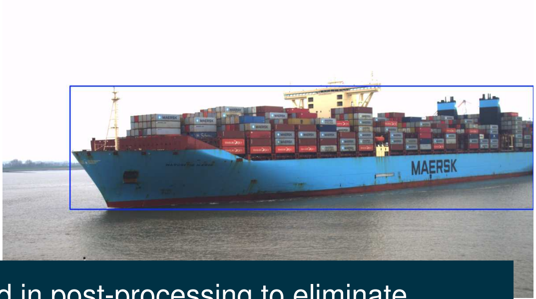


- Non-Maximum Suppression (NMS) is used in post-processing to eliminate duplicate predictions.
- NMS is not part of the model's computational graph.

Object Detection

Identify and localize obstacles in a given view for objects at varying distances and scales.

- Extract information based on patterns to compute predictions
- Shifting patterns can cause- assignment problems



- Non-Maximum Suppression (NMS) is used in post-processing to eliminate duplicate predictions.
- NMS is not part of the model’s computational graph.

SAHI: Slicing Aided Hyper Inference

Problem: Object detectors work on a fixed window size, which limits their receptive field.

SAHI: Slicing Aided Hyper Inference

Problem: Object detectors work on a fixed window size, which limits their receptive field.

- SAHI partitions a high-resolution image into patches matching the receptive field of the model.

SAHI: Slicing Aided Hyper Inference

Problem: Object detectors work on a fixed window size, which limits their receptive field.

- SAHI partitions a high-resolution image into patches matching the receptive field of the model.
- An encapsulated detector is used for each patch generating corresponding detections.

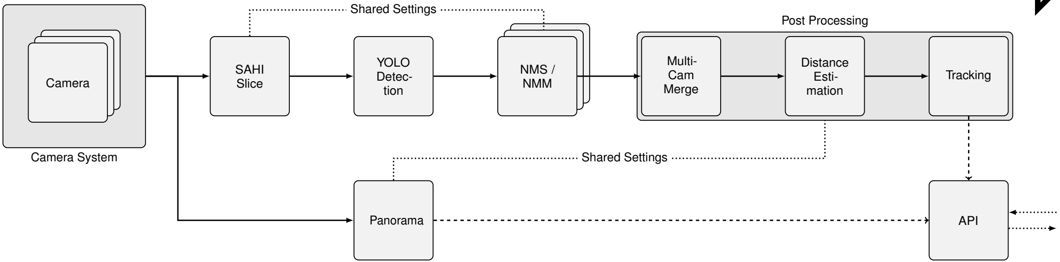
SAHI: Slicing Aided Hyper Inference

Problem: Object detectors work on a fixed window size, which limits their receptive field.

- SAHI partitions a high-resolution image into patches matching the receptive field of the model.
- An encapsulated detector is used for each patch generating corresponding detections.
- Using non-maximum merging (NMM) the patch detections are merged into the original image space.

IMPLEMENTATION

Software Pipeline



- The pipeline minimizes complexity and overhead by adopting a **single-input rule** for each independent module.
- The data flow works on a **push-based principle**.
- **Parallelization** doesn't necessarily improve delay but **optimize throughput**. The execution time is bound by the slowest upstream module.
- While the detection module is the largest overall bottleneck the panorama modules doesn't share it's dependencies and can therefore output a higher framerate.

Graphics Processing Unit Accelerations

- Due to their highly optimized instruction sets and thousands of specialized cores, GPUs excel at massive parallel computation, particularly for matrix operations, convolutions, and vectorized arithmetic.
- Model optimizations are part of any major ML library and offer a ”free” speedup.
- **Example:** We optimized the sequential NMM algorithm using vectorization. Reducing the sequential comparison to a single matrix operation.
- **Challenge:** GPU overutilization and read / write operations can cause a significant slow down in the detection module.

EVALUATION

Small Object Detection

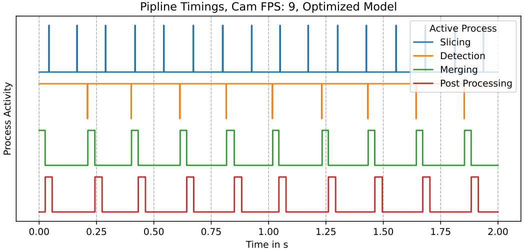
Evaluation of the pipline against the sequential execution using the Github repository of the SAHI framework.

Given the 3 high-resolution cameras we generated **195 image patches** for the full field of view.

Method	μ in s	σ in s
Baseline	0.740	0.015
Sliced SAHI Framework	4.654	0.037
Sliced Pipeline at 1 fps	0.222	0.017
Sliced Pipeline Opt. at 1 fps	**0.181**	**0.006**

Pipeline Timings

Target fps		3	4	5	6	7
base	delay in s	0.31	0.35	0.48	0.47	0.48
	fps	3.00	4.00	3.97	3.87	3.64
opt	delay in s	0.28	**0.24**	0.27	0.40	0.40
	fps	3.00	4.00	5.00	**5.28**	4.85



- Pareto-type optimization problem of delay and throughput.
- Read/Write operations forces GPU stream synchronizations influencing the complete pipeline throughput

Conclusion

- We developed a real-time maritime obstacle-detection pipeline for multiple high-resolution cameras, leveraging SAHI-based patch extraction for a centralized computing architecture.
- Our work discusses the advantages, weaknesses, and pitfalls of a centrally-executed collision-avoidance pipeline, along with a general software architecture for such a system."
- Considering the Pareto-type optimization problem of latency and throughput, extensive tests on the final hardware are conducted to fine tune operational parameters show significant speedups.

- **Future Work**: Improved SAHI patch selection, adaptive flow control, parameter tuning

Thank you. Questions?

Subject:	Engineering of a High-throughput Multi-Camera Scale-invariant Collision Avoidance Pipeline
Date:	24.06.2026
Presenter:	Alexander Klein Alexander.Klein@dlr.de
Institute:	Institute for the Protection of Maritime Infrastructures