



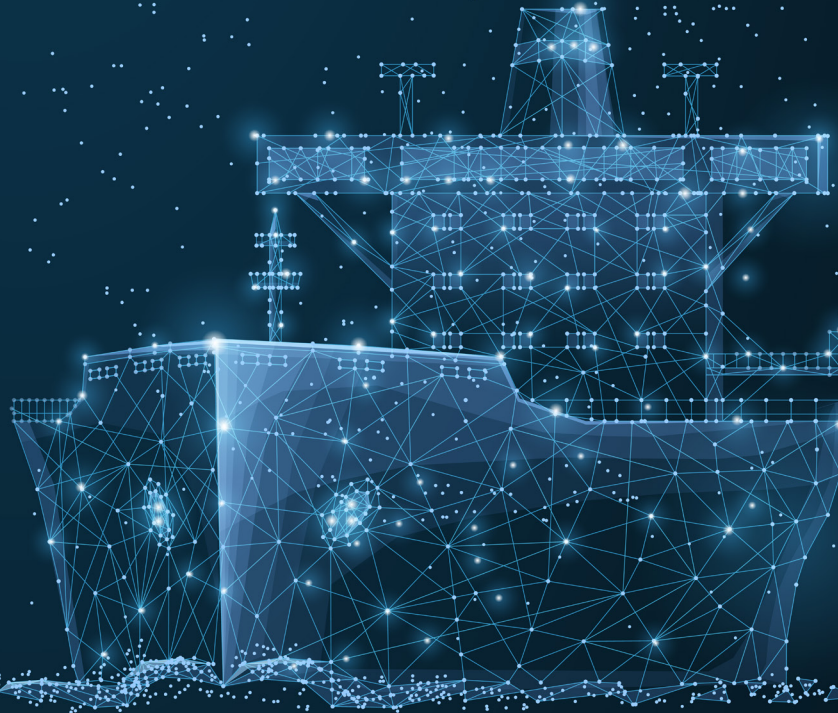
Fraunhofer-Center für Maritime
Logistik und Dienstleistungen CML



Fraunhofer-Institut für Graphische
Datenverarbeitung IGD



BUNDESAMT FÜR
SEESCHIFFFAHRT
UND
HYDROGRAPHIE



BAnA

Studie zur Bewertung von Algorithmen
für nautische Anwendungen

BAnA

Studie zur Bewertung von Algorithmen für nautische Anwendungen

Fraunhofer-Center für Maritime Dienstleistungen und Logistik CML

Paul Koch

Thomas Stach

Manfred Constapel

Hans-Christoph Burmeister

Fraunhofer-Institut für Graphische Datenverarbeitung IGD

Alexander Gillert

Matthias Vahl

Akronyme

AIS Automatic Identification System.

AUPRC Area Under Precision-Recall Curve.

BAnA Bewertung von Algorithmen für nautische Anwendungen.

CNN Convolutional Neural Networks.

DETR Detection Transformer.

DIN Deutsches Institut für Normung.

DL Deep Learning.

EVA Eingabe, Verarbeitung und Ausgabe.

FLOPs Floating Point Operations per Second.

FN False Negative.

FP False Positive.

GAN Generative Adversarial Networks.

GNSS Globales Navigationssatellitensystem.

GPL GNU General Public License.

GPS Global Positioning System.

IoU Intersection over Union.

ISO Internationale Organisation für Normung.

KI Künstliche Intelligenz.

Lidar Light Detection and Ranging.

LZM Lebenszyklusmodell.

mAP mean Average Precision.

mAR mean Average Recall.

MASS Maritime Autonomous Surface Ship.

Mastr Maritime Semantice Segmentation Training Dataset.

mIoU mean Intersection over Union.

ML Maschinelles Lernen.

NN Neuronales Netzwerk.

Radar Radio Detection and Ranging.

RGB Rot, Grün, Blau.

SMD Singapore Maritime Dataset.

TP True Positive.

VAE Variational Autoencoder.

ViT Vision Transformer.

YOLO You Only Look Once.

Inhaltsverzeichnis

1	Einleitung	2
2	Stand der Technik	2
3	Leitfaden zur Entwicklung sicherheitskritischer Deep-Learning-Anwendungen	5
3.1	Status Quo	5
3.2	Qualitätsmanagement-Konzept	8
3.2.1	Konzeptionierung	9
3.2.2	Entwicklung	11
3.2.3	Inbetriebnahme	12
3.2.4	Betrieb	13
3.2.5	Außerbetriebnahme	13
4	Untersuchung und Vergleich von Objekterkennungsalgorithmen	14
4.1	Leistungsmetriken	14
4.2	Vergleich von Architekturen	16
4.3	Training	18
4.4	Auswertung für Anwendungen im maritimen Umfeld	19
4.5	Vergleich zu generischen Anwendungen	20
5	Methoden zur Erstellung eines Datensatzes für die Entwicklung von maritimen Objekterkennungsalgorithmen	22
5.1	Einordnung der Datenbeschaffung und -verwendung in Lebenszyklusmodell	22
5.2	Allgemeine Anforderungen an Bilddaten	22
5.3	Aggregation und Sammlung von realen Bilddaten	23
5.3.1	Evaluierung bestehender Datensätze für die maritime Objekterkennung	23
5.3.2	Herausforderungen in der Nutzung von realen Bilddaten	24
5.3.3	Augmentation von Datensätzen	26
5.3.4	Annotation von Echtdaten an Bord eines Schiffes	27
5.4	Synthese von Bilddaten	27
5.4.1	Aktuelle Verfahren	28
6	Zusammenfassung	31
7	Literatur	32
8	Würdigung und abschließende Anmerkung	36

1 Einleitung

Der maritime Transport ist gegenwärtig durch eine zunehmende Digitalisierung und insbesondere Automatisierung und Autonomisierung von Prozessen an Bord von Schiffen geprägt. Neben klassischen regelbasierten Methoden erweist sich bei der Kollisionsverhütung der Einsatz von Künstlicher Intelligenz (KI) als vielversprechend, um frühzeitig und proaktiv unübersichtliche maritime (Verkehrs-)Situationen zu identifizieren und sicher aufzulösen.

KI und Maschinelles Lernen (ML) bieten vielversprechende Einsatzmöglichkeiten, doch hängen in ihrer Zuverlässigkeit direkt von der Wahl der verwendeten Verfahren und Datensätze ab. Dies trifft insbesondere auf Deep Learning (DL) zu.

Die endliche und einschränkende Verfügbarkeit an Datensätzen macht die Lösung und Generalisierung einer Anwendung unter Nutzung von ML und DL zu einer Herausforderung.

Im Rahmen der vorliegenden Studie wird diese Thematik anhand folgender Zielstellungen bearbeitet:

- Umfassende Recherchen zu bestehenden Regularien mit Bezug auf die Entwicklung sicherheitskritischer Anwendungen basierend auf DL in der maritimen Branche als Schwerpunkt. Ableitung und Nutzung von Handlungsempfehlungen, welche im Rahmen der Studie VerifAI für die Entwicklung und den Betrieb von Maritime Autonomous Surface Ship (MASS) entwickelt worden sind, sowie eine Analyse, inwiefern sich diese Prozesse auf die spezifische Domäne der Objekterkennung übertragen lassen (s. Kapitel 3).
- Ergebnisse von Objektdetektions- und Segmentierungsalgorithmen, welche im Bereich der Computer Vision Anwendung finden zu vergleichen sowie eine Gegenüberstellung der Ergebnisse gleichartiger Algorithmen, welche mit Fokus auf maritime Daten trainiert wurden, durchzuführen (s. Kapitel 4).
- Identifizierung von Herausforderungen bei der Verwendung von Bilddaten für das Training entsprechender Objekterkennungsalgorithmen, Strategien zur Augmentation von Bilddaten oder zur Bereitstellung vollständig synthetischer Bilddaten mit Hilfe von Algorithmen des maschinellen Lernens (s. Kapitel 5).

Der Fokus der Studie liegt dabei in der Betrachtung aus der nautischen Perspektive und solcher Anwendungen, die mit Hilfe von ML entwickelt werden. Eine Marktanalyse in Kapitel 2 zeigt den Stand der Technik hinsichtlich dieser Perspektive auf und unterstreicht die Bedeutsamkeit dieser Studie und der Umsetzung ihrer Ergebnisse.

Modelle zur Objektdetektion und semantischen Segmentierung können als Werkzeuge dienen, um kritische Informationen über Position, Bewegung und Identität maritimer Objekte zu extrahieren und mit vorhandenen Daten und Informationen zu fusionieren. Dies ist notwendig, um den Schiffsverkehr zu überwachen und einen sicheren und effizienten Verkehr zu gewährleisten. Diese Technologien stellen somit eine wichtige Komponente für die Automatisierung und Autonomisierung von Schiffen dar.

Die Gründe für den zunehmenden Einsatz dieser Algorithmen liegen zum einen in den Fortschritten der ML-basierten Verfahren und zum anderen in fortwährenden Leistungssteigerungen geeigneter Hardware. Bei den Verfahren spielen insbesondere Convolutional Neural Networks (CNN) und Transformer-Netzwerke eine entscheidende Rolle (s. Kapitel 4).

2 Stand der Technik

Die Untersuchung mit Blick auf den Stand der Technik basiert auf einer Sichtung der Produkte und den entsprechenden, öffentlich zugänglichen Produktbeschreibungen. Korrespondierende wissenschaftliche Artikel zu den Produkten existieren in der Regel nicht. Die Produktbeschreibungen dienen der Produktvermarktung und lassen technische Spezifikationen offen, sodass nur Annahmen über zur Entwicklung verwendete Datenquellen können. Die Ergebnisse hierzu sind in Tabelle 1 zusammengefasst. Die getroffenen Annahmen basieren auf der technischen Kompetenzen des Fraunhofer CML und Fraunhofer IGD im Bereich (teil-)autonomer Navigation, Lagebilderstellung und bildbasierender Informatik sowie den begrenzt vorhandenen technischen Datenblätter der Hersteller.

Tabelle 1: Gesichtete Unternehmen oder Produkte mit KI-basierten Technologien kategorisiert nach ihren Datenquellen. Mit * gekennzeichnete Unternehmen oder Produkten erstreben oder ermöglichen bereits eine autonome Schiffsführung.

Unternehmen/Produkt	RGB ¹ - Kamera	Infrarot- Kamera	LIDAR/ RADAR	IMU	GNSS	AIS
Avikus HiNAS 2.0* [1]	X	X	X		X	X
Avikus NeuBoat [2]	X	X	X		X	X
Awarion [3]	X	X				
Captain AI [4]	X		X		X	X
Groke Pro [5]	X	X	X	X	X	X
Infoceanus Command [6]	X					
Kongsberg Maritime Autonomous Shipping [7]	X		X	X	X	X
Kongsberg Maritime Intelligent Awareness [8]	X		X	X	X	X
Marine AI Guardian Autonomy* [9]	X		X			X
Mayflower Autonomous Ship* [10]	X		X	X	X	X
Meguri2040* [11]	X				X	X
MTU NautIQ Copilot* [12]	X	X	X	X	X	X
Orca AI [13]	X	X				
Rite-Solutions [14]	X					
Roboat* [15]	X		X		X	
Robopec [16]		X	X		X	X
Saildrone* [17]	X			X	X	X
Sea Machines SM300 [18]	X		X	X	X	X
Sea.AI [19]	X	X		X	X	
Seadronix AVISS [20]	X					
Seafar [21]	X		X	X	X	X
Seasight* [22]	X	X	X	X	X	X
SEDOR AI Ship Detection [23]	X					
SVESSEL Samsung Smart Ship [24]	X					X
Wärtsilä Voyage Autonomy Solutions [25]	X	X	X		X	X
Yara Birkeland* [26]	X	X	X	X	X	X
Summe	25	11	16	11	18	18

Insgesamt wurden 26 Produkte gesichtet. Die Anwendungsspanne dieser KI-gestützten Produkte reicht von Schiffserkennung, welche in allen gesichteten Produkten vorhanden ist, bis zur autonomen Schiffsführung, welche von insgesamt neun gesichteten Produkten ermöglicht oder angestrebt wird. Allgemein ist deutlich erkennbar, dass Produkte mit höherem Autonomisierungsgrad weniger auf dem Markt verfügbar sind.

Die Systeme greifen auf die Daten verschiedener Sensoren, wie RGB- oder Infrarot-Kamerasystemen, Light Detection and Ranging (Lidar), Radio Detection and Ranging (Radar), Globales Navigationssatellitensystem (GNSS) und Automatic Identification System (AIS), zurück. Mithilfe der KI-gestützten Systeme werden die Daten zusammengeführt und interpretiert, sodass Hindernisse, Schiffe und mögliche Routen automatisch erkannt und geschätzt werden können. Fast alle Produkte (25) verwenden die Daten aus optischen Kameras. Die nächst häufigsten Datenquellen sind ein GNSS, beispielsweise Global Positioning System (GPS), und das weit verbreitete Funksystem AIS (je 18 Anwendungsfälle).

Daten aus RGB-Kamerasystemen haben in der Hindernis- und Schiffserkennung einen entscheidenden Vorteil gegenüber AIS- und GNSS-Daten: Richtig interpretiert können KI-gestützte Systeme unvorhergesehene Hindernisse wie treibende Container oder nicht kartierte Landmassen erkennen. Würden die Systeme nur AIS- oder GNSS-Daten nutzen, könnten sie entsprechend nur Hindernisse mit den passenden Empfängern erkennen. Bei guten Wetterbedingungen sind optische Kameras jedoch detaillierter als Radar-Systeme und eine höhere Reichweite als Lidar-Systeme [27]. Aus den betrachteten Produktbeschreibungen geht hervor, dass RGB-Kamerasysteme nicht nur die am häufigsten genutzten Datenquellen sind, sondern dass auch in mindestens 15 Fällen die KI-gestützte Schiffs- und Hinderniserkennung hauptsächlich auf Grundlage dieser Daten basiert.

Trotz der weiten Verbreitung der KI-gestützten Kamerasysteme fehlen die entsprechenden Prüf- und Zulassungsprozesse [28]. Abhängig von der Wetterlage, dem zu erkennenden Objekt und des Trainingsdatensatzes kann die Leistung der Systeme variieren. Fehler wie ein nicht erkanntes Schiff bei nebliger Wetterlage können zu schwerwiegenden Folgen führen. Um die Sicherheit auf See zu gewährleisten, werden Prüfsysteme für die entsprechenden

¹ Rot, Grün, Blau (RGB)-Kamerasysteme sind jene, die das sichtbare Farbspektrum bedienen.

KI-gestützten Systeme benötigt. Sie müssen in der Lage sein, die Produkte auf ihre Zuverlässigkeit zu testen und so das Risiko für Unfälle auf ein tolerierbares Maß begrenzen.

3 Leitfaden zur Entwicklung sicherheitskritischer Deep-Learning-Anwendungen

Das Kapitel zum Leitfaden zur Entwicklung sicherheitskritischer Deep-Learning-Anwendungen besteht aus zwei Abschnitten. Zunächst wird der aktuelle Stand behandelt, welcher insbesondere durch Veröffentlichungen von den Einrichtungen Deutsches Institut für Normung (DIN) oder Internationale Organisation für Normung (ISO) abgebildet wird. Anschließend werden die für diese Studie relevanten Maßnahmen zusammengefasst und, sofern notwendig, ergänzt.

3.1 Status Quo

Im Rahmen der Studie sind mehrere Dokumente auf Anwendbarkeit bei der Entwicklung von DL-basierten, sicherheitskritischen Objekterkennungs- und -segmentierungssystemen untersucht worden. Dabei handelt es sich um folgende Dokumente:

- VerifAI, Studie zur zielbasierten Standardisierung in der Prüfung und Zulassung intelligenter Entscheidungseinrichtungen von teilautonomen Überwasserfahrzeugen [28]
- DIN SPEC 92001-1:2019-04, Künstliche Intelligenz - Life Cycle Prozesse und Qualitätsanforderungen - Teil 1: Qualitäts-Meta-Modell [29]
- DIN SPEC 92001-2:2020-12, Künstliche Intelligenz - Life Cycle Prozesse und Qualitätsanforderungen - Teil 2: Robustheit [30]
- Leitfaden zur Gestaltung vertrauenswürdiger Künstlicher Intelligenz [31]
- DIN SPEC 13266:2020-04, Leitfaden für die Entwicklung von Deep-Learning-Bildererkennungssystemen [32]
- ISO/IEC Technical Report 29119-11, Software and systems engineering - Software testing - Part 11: Guidelines on the testing of AI-based systems [33]
- ISO/IEC International Standard 22989: Information Technology - Artificial Intelligence - Artificial Intelligence Concepts and Terminology [34]
- ISO/IEC International Standard 23053:2022-06, Framework for Artificial Intelligence (AI) Systems Using Machine Learning (ML) [35]
- ISO/IEC Technical Specification 4213, Information technology - Artificial intelligence - Assessment of machine learning classification performance [36]
- ISO/IEC Technical Report 24030, Information Technology - Artificial Intelligence (AI) - Use Cases [37]

Diese Dokumente bearbeiten das Themenfeld KI aus unterschiedlichen Perspektiven und betrachten Fragestellung allgemein ohne konkreten Kontext- oder Anwendungsbezug bis hin zu sehr spezifisch im Bereich Objekterkennung und -segmentierung unter Verwendung von mit DL entwickelten Modellen. In Abbildung 1 auf der nächsten Seite wird die Ontologie dieser Dokumente dargestellt.

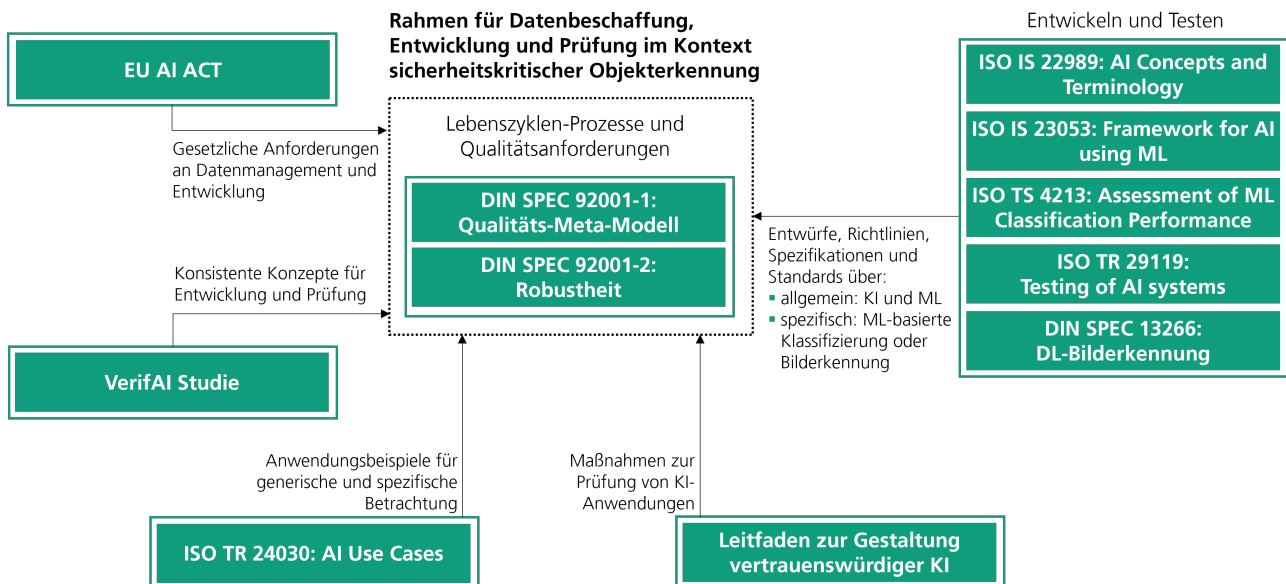


Abbildung 1: Ontologische Darstellung zu gesichteten Dokumenten.

In der Studie VerifAI werden in den aktuellen Prüf- und Zulassungsprozessen für KI-basierte Schiffstechnologien Handlungsbedarfe identifiziert [28]. Diese Handlungsbedarfe zielen darauf ab die Lücke zu schließen, welche durch den Fortschritt von KI-basierten Schiffstechnologien auf der einen Seite und fehlender Prüf- und Zulassungsprozesse auf der anderen Seite entsteht. Aus der Studie geht insbesondere hervor, dass die Verwendung von Kamerasystemen in KI-basierten Schiffstechnologien weit verbreitet ist, wenngleich hierfür noch keine dedizierten Prüf- und Zulassungsprozesse existieren. Damit ist die Relevanz der Bewertung von Algorithmen für nautische Anwendungen (BAnA) offensichtlich. Bei der Prüfung der Schiffstechnologien soll demnach der Fokus auf der Beantwortung der Frage *ob* und nicht *wie* das vorliegende System funktioniert. Die Prüfprozesse hierfür sollen einem modellagnostischen Ansatz folgen, um verschiedenartige KI-Modelle einheitlich prüfen zu können. Ferner wird in VerifAI vorgeschlagen augmentierte oder gänzlich synthetische Testdaten bei der Prüfung zu verwenden. Dies hat den Vorteil, dass Testdaten ziel- und anwendungsorientiert beschaffen werden und so dass keine Testdaten aus der Entwicklung verwendet werden.

In DIN SPEC 92001-1 wird ein Qualitätsmetamodell vorgestellt [29], anhand welchem relevante Qualitätssicherungsprozesse entlang des Lebenszyklus (Life Cycle) einer KI im allgemeinen abgeleitet werden. Dieses Lebenszyklusmodell (LZM) stammt aus dem System- und Software-Ingenieurwesen (Vgl. ISO/IEC/IEEE 12207:2017 [38]). Als Lebenszyklusphasen werden folgende aus dem Englischen übersetzten Begriffe eingeführt:

1. Konzeption (Concept)
2. Entwicklung (Development)
3. Inbetriebnahme (Deployment)
4. Betrieb (Operation)
5. Außerbetriebnahme (Retirement)

Die erste Lebenszyklusphase, Konzeption, beinhaltet alle Schritte, die vor der eigentlichen Entwicklung eines KI-Modells stattfinden. Dies ist insbesondere die Problemdefinition sowie Evaluation geeigneter Modelle. Darüber hinaus sollten für Qualitätssicherungszwecke Akzeptanzkriterien definiert werden. Diese Schritte decken sich mit der Formalisierung als Teil des Sicherheitskonzepts in der Studie VerifAI. In der nächsten Lebenszyklusphase findet die eigentliche Entwicklung statt. Klassische Schritte dieser Phase, die aus dem System- und Software-Ingenieurwesen stammen, sind System-Design, Prototypen-Entwicklung, Implementation und Integration. Im KI-Kontext kommen bei datengetriebenen Lösungen noch spezifische Schritte hinzu, beispielsweise Datenbeschaffung und -analyse. Bei Expertensystemen werden in dieser Phase Regeln oder Regelsysteme formuliert. Am Ende dieser Phase wird ein Modell samt Hyperparametern gewählt, welches das zuvor definierte Problem löst und die Akzeptanzkriterien erfüllt. In der dritten Phase, der Inbetriebnahme, wird das KI-Modell in die vorgesehene Umgebung übertragen und darin gegebenenfalls nachtrainiert. In dieser Phase stellt sich heraus, wie das Modell unter Realbedingungen agiert. Falls die Erwartungen an das Modell nicht erfüllt werden, kann im Lebenszyklus zur Entwicklung zurückgekehrt

werden, um entsprechende Verbesserungen zu tätigen. Beim Betrieb eines KI-Modells spielt die Beobachtung des Modellverhaltens in seiner Umgebung eine zentrale Rolle. Für Modelle, die auf Online-Training setzen gilt dies, weil ihr Modellverhalten mit der Zeit veränderlich ist. Abgesehen davon gilt für alle Modelle, dass ihre Umgebung nicht statisch ist, sondern sich mit der Zeit verändert und dies Auswirkung auf das Modellverhalten haben kann. Am Ende des Lebenszyklus findet die Außerbetriebnahme des Modells statt.

Die Phasen im LZM werden in diesem Qualitätsmetamodell darüber hinaus aus den folgenden Perspektiven betrachtet:

- Risiko (niedrige oder hohe Risikobeurteilung)
- Ebene (Umgebung, Plattform, Daten, Model)
- Qualität (Funktionalität und Performanz, Robustheit, Nachvollziehbarkeit)

Das Dokument DIN SPEC 92001-2 nutzt dieses Qualitätsmetamodell und beschreibt darauf basierend qualitätssichernde Prozesse, die den Lebenszyklusphasen und den betrachteten Perspektiven zugeordnet werden [30].

Vergleichbar mit den DIN SPEC zum Qualitätsmetamodell wird im Leitfaden zur Gestaltung vertrauenswürdiger KI ein Prüfkatalog eingeführt [31]. Dieser Prüfkatalog umfasst Maßnahmen zur Prüfung einer KI-Anwendung hinsichtlich sechs Dimensionen entlang des Lebenszyklus einer KI-Anwendung. Beim Lebenszyklus wird zwischen den folgenden Phasen unterschieden:

- Daten
- Entwicklung und Modellbildung der KI-Komponente
- Einbettung
- Betrieb der KI-Anwendung

Bei den Dimensionen, denen die Maßnahmen zugeordnet werden, handelt es sich um folgende:

- Fairness
- Autonomie und Kontrolle
- Transparenz
- Verlässlichkeit
- Sicherheit
- Datenschutz

Diese Dimensionen lassen sich mit den Qualitätsmerkmalen des LZM aus [29, 30] vergleichen, wobei sie in diesem Leitfaden breiter und detaillierter aufgefasst werden.

Der technische Bericht ISO/IEC TR 29119-11 [33] befasst sich mit den Herausforderungen bei der Prüfung von verschiedenen KI-basierten Systemen. Hierfür werden zunächst die Grundlagen von KI-basierten Systemen eingeführt und anschließend um ihre spezifischen Charakteristika erweitert. Basierend auf den Charakteristika von KI-basierten Systemen, werden ihre Herausforderungen hinsichtlich ihrer Prüfung abgeleitet und mögliche Prüfungsansätze vorgeschlagen.

In DIN SPEC 13266 wird ein Leitfaden spezifisch für die Entwicklung von DL-basierten Bilderkennungssystemen vorgestellt [32]. In Anlehnung an den in DIN SPEC 92001-1 [29] eingeführten Lebenszyklus bildet der Leitfaden Prozesse innerhalb der Phasen Konzeption, Entwicklung und Inbetriebnahme ab.

Der internationale Standard ISO/IEC IS 22989 gibt einen breiten Überblick über KI im Allgemeinen [34]. Dabei werden verschiedene Konzepte behandelt, beispielsweise die Unterscheidung zwischen starker und schwacher KI, Konzepte aus dem Bereich ML und soziale, juristische oder organisatorische Fragestellungen wie beispielsweise Vertrauenswürdigkeit von KI.

Der internationale Standard ISO/IEC IS 23053 gibt einen Überblick über ML-basierte KI [35]. Hierfür werden aus diesem Bereich grundlegende Begriffe eingeführt, um ein entsprechendes System, seine Komponenten sowie Funktionen beschreiben zu können. Dabei wird auf ein Entwicklungs- oder Prüfkonzept verzichtet. Auf Messmethoden der Leistungsfähigkeit wird beschränkt eingegangen.

Bei der technischen Spezifikation ISO/IEC TS 4213 handelt es sich um ein spezifischeres Dokument, als ISO/IEC 23053 [36]. Es werden vorwiegend Methoden zur Messung der Leistungsfähigkeit von ML-basierten Klassifizierungssystemen, zu denen auch Objekterkennungs- und -segmentierungssysteme zählen.

In dem technischen Bericht ISO/IEC TR 24030 werden zahlreiche Anwendungsbeispiele und ihre Ziele sowie Herausforderungen und Bedenken aufgeführt [37]. Unter den thematisierten Branchen befinden sich auch Logistik und Transportwesen. Das Dokument ist dahingehend nützlich, um Anwendungsbeispiele aus naheliegenden Branchen als Referenzen oder Vergleiche für z.B. Standardisierung von Entwicklungs- oder Prüfprozessen heranzuziehen. Das naheliegendste Beispiel sind autonome Züge. Mit diesen werden verschiedene Ziele verfolgt, wie erhöhte Zuverlässigkeit und Sicherheit, Energieverbrauchseinsparungen, Erhöhung der Transportkapazitäten und Personaloptimierung. Ein entsprechendes KI-System wäre für verschiedene Aufgaben zuständig, u.a. Geschwindigkeitsregelung, Türöffnung und -schließung und Hinderniserkennung. Die dafür benötigten Umweltinformationen werden mit Sensoren wie GPS, Lidar, Radar und verschiedenen Kamerasystemen erfasst und entsprechen denen aus der Schifffahrt.

3.2 Qualitätsmanagement-Konzept

Der vorgestellte Leitfaden zu einem Qualitätsmanagement-Konzept orientiert sich im Aufbau an das LZM nach DIN SPEC 92001-1 [29]. Bei dem LZM handelt es sich um ein Metamodell, also ein abstraktes Modell, welches als Vorlage zur Erstellung eines spezifischen Modells dient. Das Metamodell richtet sich bereits speziell an KI-Anwendungen und berücksichtigt Aspekte, die eine solche Anwendung von der Konzeption bis zur Inbetriebnahme und darüber hinaus mit sich bringt. Zum einen wird der Lebenszyklus ganzheitlich betrachtet und in die fünf Lebenszyklus-Phasen Konzeptionierung, Entwicklung, Inbetriebnahme, Betrieb und Außerbetriebnahme gliedert. Die allgemeinen Aufgaben und qualitätssichernden Maßnahmen während der einzelnen Phasen sind in Abbildung 2 dargestellt.

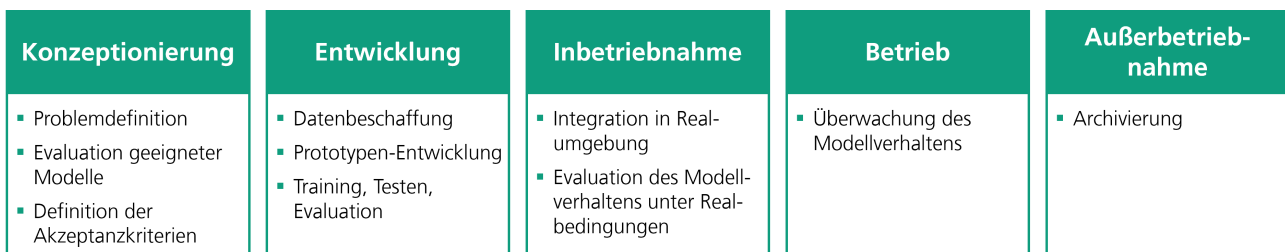


Abbildung 2: Übersicht zu den Phasen des LZM.

Darüber hinaus lassen sich die qualitätssichernde Maßnahmen aus verschiedenen Perspektiven betrachten. Aus dem LZM werden für die vorliegende Studie die Qualitätsmerkmale

- Funktionalität und Performanz,
- Robustheit und
- Nachvollziehbarkeit

übernommen. Dieses Grundgerüst bietet passende Anknüpfungspunkte für die Studie VerifAI, in welcher bereits Maßnahmen für eine zuverlässige Prüfung und Zulassung von KI-basierten Systeme identifiziert worden sind. Darüber hinaus lassen sich hierin die Ergebnisse aus den weiteren gesichteten Dokumenten hervorragend einordnen.

3.2.1 Konzeptionierung

Konzeptionierung	Funktionalität & Performanz	K.F.1	Formalisierung der Anwendungsdomäne
		K.F.2	Charakterisierung von relevanten Datensätzen
		K.F.3	Definition von Performanz-Metriken und Erfolgskriterien
		K.F.4	Definition von Testszenarien
		K.F.5	Auswahl der Hardware
	Robustheit	K.R.1	Auswirkungen bei technischem Versagen
	Nachvollziehbarkeit	K.N.1	Methoden zur Erklärbarkeit des Modellverhaltens

Abbildung 3: Maßnahmen in der Phase Konzeptualisierung.

K.F.1 Formalisierung der Anwendungsdomäne

Die Formalisierung der Anwendungsdomäne wird als erste notwendige Maßnahme betrachtet. Aus ihr lassen sich sowohl Anforderungen an Datensätze und Modelle ableiten. Die Formalisierung wird in [28] behandelt. Das Systemverhalten sollte sich anhand der Formalisierung der Anwendungsdomäne in zumindest die folgenden vier Unterdomänen abgrenzen können [39]:

1. Input-Domäne: Menge an Input, mit welchen das Modell umgehen kann.
2. Betriebsmäßige Domäne: Menge an Input, welches für das Modell im Betrieb erwartet wird.
3. Fehlfunktionale Domäne: Menge an Input, welches das Modell im Falle von Fehlfunktionen innerhalb im gesamten software- und hardware-seitigen System erwarten kann.
4. Feindliche² Domäne: Menge an Input, welches das Modell bei feindlichen Störungen erwarten kann.

Ferner wird mit der Berücksichtigung der Anwendungsdomäne die Brücke zu MASS geschlagen, da hier das Operational Envelope eine wichtige Rolle spielt [40]. Das Operational Envelope wird - vereinfacht formuliert - durch die Erweiterung der Anwendungsdomäne um den Anteil menschlicher operativer Steuerung gebildet.

K.F.2 Charakterisierung von relevanten Datensätzen

Die zur Entwicklung herangezogenen Datensätze müssen sowohl im Umfang als auch in Qualität [30, 39] gegebenen Anforderungen entsprechen. In der Phase der Konzeption wird zunächst der inhaltliche Umfang der notwendigen Datensätze charakterisiert. Aus der formalisierten Anwendungsdomäne (s. Maßnahme K.F.1) der KI-Anwendung lässt sich schließen, welche Daten für die Bedienung welcher Unterdomänen notwendig sind. Die heranzuziehenden Datensätze sollten insbesondere die Input-Domäne und die Betriebsmäßige Domäne bedienen.

K.F.3 Definition von Performanz-Metriken und Erfolgskriterien

Die Definition von Metriken zur Messung der Performanz, sowie zu erreichende Erfolgskriterien dienen als Messung der Zielerreichung eines Modells [28]. Je nach vorliegender Problemstellung eignen sich verschiedene Metriken zur Messung der Performanz eines Modells. In [31, 35] werden Metriken für ML-Modelle i.A. behandelt. In der vorliegenden Studie handelt es sich bei der Problemstellung um ein Klassifikationsproblem. Mögliche Metriken, die sich hierfür im Speziellen eignen, werden in [36] behandelt. Performanz-Metriken im Kontext von Objekterkennung und semantischer Segmentierung, welche auch in dieser Studie herangezogen worden sind, werden in Kapitel 4.1 vorgestellt.

²Aus dem Englischen von "adversarial" im Kontext von "adversarial attacks" abgeleitet.

K.F.4 Definition von Testszenarioszenarien

Zur späteren Untersuchung der Performanz eines Modells kann im Vorfeld eine Menge an Testszenarioszenarien definiert werden. Die Szenarien können gemäß Eingabe, Verarbeitung und Ausgabe (EVA)-Prinzip aus korrespondierenden Eingabe-Ausgabe-Paaren bestehen [39]. Hierbei sollten bewusst Testszenarioszenarien definiert werden, die nicht nur innerhalb, sondern auch außerhalb oder entlang der Grenze der Anwendungsdomäne liegen, um das Modell auf Robustheit zu untersuchen (s. Kapitel B.2.1). Mögliche Herangehensweise zur Beschreibung und Generierung von Szenarioszenarien werden in [28] vorgestellt.

K.F.5 Auswahl der Hardware

Die Auswahl der Hardware sollte für die Entwicklungsphase das Training des Modells auf einem hinreichend leistungsstarken Rechner berücksichtigen. Das Testen des Modells sollte auf einem weiteren System erfolgen, welches wie vorliegend oder vergleichbar im Betrieb zum Einsatz kommt. Als System wird in diesem Fall nicht nur der Rechner an sich, sondern auch das Kamerasystem und andere für die Funktionalität & Performanz entscheidende Systemkomponenten betrachtet.

Unter den gesichteten Dokumenten wird die Notwendigkeit von Hardware-Anforderungen nur in [33] und [38] vage erwähnt, nicht aber inhaltlich bearbeitet. Eine Zusammenfassung möglicher Hardware zum Training von Modellen im Bereich Computer Vision wird in [41] behandelt. Die Hardware zum Testen des Modells ist aufgrund von individuellen Anwendungen spezifisch zu betrachten und kann an dieser Stelle nicht verallgemeinert behandelt werden.

K.R.1 Auswirkungen und Strategien bei technischem Versagen

Technisches Versagen kann unterschiedliche Ursachen haben. Betrachtet man das Modell aus der Perspektive des EVA-Prinzips, so kann es bereits zur Verarbeitung ungeeignete Eingabedaten erhalten, oder die Verarbeitung schlägt fehl und es erfolgt eine fehlerhafte Ausgabe. Welche Eingabedaten geeignet sind und welche nicht, lässt sich der formalisierten Anwendungsdomäne (s. Maßnahme K.F.1) entnehmen. Die korrekte Verarbeitung wiederum lässt sich mit den definierten Performanz-Metriken ermitteln (s. Maßnahme K.F.3). Die Auswirkungen die aus dem technischem Versagen sollten vorab eingeschätzt werden, um mögliche Strategien zu entwickeln. Mögliche Herangehensweisen werden in [30] beschrieben.

K.N.1 Methoden zur Erklärbarkeit des Modellverhaltens

Die Erklärbarkeit des Modellverhaltens ist für Entwickler, Anwender und Prüfer entscheidend. Ein möglicher und insbesondere modell-agnostischer Ansatz der Erklärbarkeit wird in [28] vorgestellt. Dieser basiert grundlegend auf dem EVA-Prinzip, also dem Abgleich zwischen der zu erwarteten Ausgabe bei gegebener Eingabe. Je nach Bedarf und Möglichkeiten kann das Modellverhalten unter zusätzlichem Aufwand spezifischer untersucht werden. Mögliche Ansätze werden in [31] vorgestellt. Hierbei muss grundlegend zwischen intrinsisch-interpretierbaren White-Box-Modellen und nicht intrinsisch-interpretierbaren Black-Box-Modellen unterschieden werden. Bei der Klassifizierung und Segmentierung von Bildinhalten handelt kommen in der Regel Black-Box-Modelle zum Einsatz, also jene, die nicht intrinsisch-interpretierbar sind.

3.2.2 Entwicklung

Entwicklung	Funktionalität & Performanz	E.F.1	Eignung von Datensätzen
		E.F.2	Eignung von Modellen
	Robustheit	E.R.1	Erfassung der Anwendungsgrenzen und Unsicherheiten
	Nachvollziehbarkeit	E.N.1	Beschreibung des Modellverhaltens
		E.N.2	Beschreibung der verwendeten Datensätze

Abbildung 4: Maßnahmen in der Phase Entwicklung.

E.F.1 Eignung von Datensätzen

Aufgrund des entscheidenden Einflusses des verwendeten Datensatzes spielt die Einhaltung von ausreichender Datenqualität eine entscheidende Rolle [28]. In der Konzeptionierungsphase werden basierend auf der formalisierten Anwendungsdomäne (s. Maßnahme **K.F.1**) relevante Datensätze charakterisiert und Anforderungen formuliert (s. Maßnahme **K.F.2**). Datensätze können grundlegend auf folgenden Wegen beschaffen werden [28]:

- Es wird ein bestehender öffentlicher oder privater Datensatz erworben.
- Es wird ein bestehender öffentlicher oder privater Datensatz augmentiert.
- Es wird ein synthetischer Datensatz erzeugt (und augmentiert).

Die Augmentierung von Datensätzen ist bereits eine gängige und effektive Methode zur Erweiterung von Datensätzen. Insbesondere die Synthetisierung von Bilddaten, und die Verwendung dieser zum Training von ML-basierten Modellen wird wegen des vielversprechenden Potenzials aktuell in der Forschung untersucht. Verfügbare öffentliche oder private Datensätze werden in der Entwicklungsphase auf ihre Qualität geprüft. Die problematischen Manifestationen mangelnder Datenqualität werden in [28] angesprochen. Dabei gilt es zu beachten, dass der Datensatz relevant, vollständig, ausgeglichen und präzise ist. Entsprechende Methoden werden in [39] vorgestellt. Methoden zur Untersuchung von Datensätzen auf Bias sind in [42] beschrieben. Wenn zur Verbesserung der Datenqualität notwendig, können verfügbare Datensätze augmentiert oder auch mit synthetischen Datensätzen kombiniert werden.

E.F.2 Eignung von Modellen

Die Untersuchung der Eignung von Modellen verläuft über mehrere Schritte. Für das Lernen des Modells sind mindestens drei Schritte erforderlich [39]:

1. Modellauswahl
2. Modelltraining
3. Wahl der Hyperparameter
4. ggf. Transferlernen

Zunächst werden für die angestrebte Anwendung geeignete Modelle ausgewählt. Im Falle dieser Studie ist die Hauptunterscheidung, ob es sich um Objekterkennung oder -segmentation handelt. Nachfolgend wird das Modell hinsichtlich einer Zielfunktion trainiert, wobei dies iterativ mit der Wahl der Hyperparameter geschieht. Die Hyperparameter wiederum parametrisierung das Training an sich. Sofern notwendig, wird ein Modell nachtrainiert. Dies ist z.B. dann der Fall, wenn ein aufwendig vortrainiertes, allgemeines Modell für spezifische Anwendungsdomänen nachtrainiert wird.

Die Performanz des Modells wird anhand der in der Konzeptionierungsphase definierten Testszenarien (s. Maßnahme **K.F.4**) und Performanz-Metriken (s. Maßnahme **K.F.3**) gemessen und gewertet.

E.R.1 Erfassung der Anwendungsgrenzen und Unsicherheiten

Ein robustes Modell setzt voraus, dass es mit fehlerhaften, verrauschten, unbekanntem und feindlichen Eingabedaten umgehen kann [29]. In [31] werden zur Erfassung der Anwendungsgrenzen und Unsicherheiten unterschiedliche Methoden vorgestellt. Allgemein anwendbar ist die systematische Schwachstellensuche oder die Folgenuntersuchung von erwartbaren Störungen z.B. durch Sensorrauschen und Verzerrung oder Verfremdung von Eingabedaten. Letzteres kann auftreten, wenn z.B. das Kamerasystem widrigen Wetterverhältnissen ausgesetzt ist.

E.N.1 Beschreibung des Modellverhaltens

Zur Nachvollziehbarkeit für die Entwickler, Anwender und Prüfer ist es notwendig, das Modellverhalten zu dokumentieren. Dies sollte nach den in Maßnahmen K.N.1 formulierten Methoden erfolgen.

E.N.2 Beschreibung der verwendeten Datensätze

Eine Beschreibung der verwendeten Datensätze nützt hauptsächlich Entwicklern, welche ein vortrainiertes Modell reproduzieren oder nachtrainieren wollen und dabei offensichtlich nicht auf bereits verwendete Datensätze zurückgreifen können. Mögliche Methoden zur Beschreibung der verwendeten Datensätze werden in [28] vorgestellt.

3.2.3 Inbetriebnahme

Inbetriebnahme	Robustheit	I.R.1	Begutachtung des Modellverhaltens unter Realbedingungen
		I.R.2	Archivierung des Vorgänger-Modells
	Nachvollziehbarkeit	I.N.1	Versionierung der Entwicklungsdatensätze und des Modells

Abbildung 5: Maßnahmen in der Phase Inbetriebnahme.

I.R.1 Begutachtung des Modellverhaltens unter Realbedingungen

Bei der Inbetriebnahme wird das Modell das erste Mal den Realbedingungen ausgesetzt. Dies bedeutet zum einen, dass es in der vorgesehen Systemarchitektur integriert ist und dass das System der realen Umgebung ausgesetzt ist. Im Zuge dessen, erhält das Modell Eingabedaten, die in der Anwendungsdomäne liegen, sich aber im Detail von den Daten aus der Entwicklung unterscheiden. Das Modell muss entsprechend darin begutachtet werden, ob das beschriebene Modellverhalten (s. Maßnahme E.N.1) wiedergegeben wird und die Erfolgskriterien (s. Maßnahme K.F.3) erreicht werden. Dies kann anhand Szenarien erfolgen, wie sie bestenfalls bereits in den Testszenarien definiert und geprüft worden sind. Als Realszenarien können die Testszenarien nachgespielt werden. Darüber hinaus ist es denkbar eine systematische Schwachstellensuche [31] durchzuführen.

I.R.2 Archivierung eines Vorgänger-Modells

Sofern es sich mit dem vorliegenden Modell um eine Aktualisierung und nicht eine Erstbetriebnahme durchgeführt wird, ist es sinnvoll, das lauffähige Vorgängermodell zu archivieren. Diese kann als Rückfall-Option dienen, sofern sich unerwartet technische Probleme beim aktuellen Modell ergeben sollten.

I.N.1 Versionierung und Archivierung der Entwicklungsdatensätze, des Modells und der Systemkonfiguration

Sowohl der Datensatz zur Entwicklung des Modells, das Modell selbst und eine Spezifikation der Systemkonfiguration müssen zur Inbetriebnahme versioniert und archiviert werden.

Zurückverfolgbare verwendete Trainings-, Test-, und Validierungsdaten sind insb. für das Nachtrainieren erforderlich [31, 43, 44]. Eine Versionierung muss auf stringenter Wege erfolgen, was wiederum klare Kriterien voraussetzt. Die Kriterien sollten verschiedene Änderungen in den Datensätzen (z.B. hinsichtlich Format, Struktur, Metadaten, Augmentierung, Bereinigung oder Zusammenführen mit anderen Datensätzen) berücksichtigen [44].

Das Modell selbst kann über die Speicherung von Modellarchitektur und -parametern, z.B. Gewichtungen bei einem Neuronalem Netzwerk [31], versioniert werden. Darüber hinaus sollte bei mit Daten trainierten Modellen

hervorgehen, welche Versionen welcher Datensätze verwendet worden sind. Die Versionierung des Modells ermöglicht es Entwicklern, Anwendern und Prüfern ein Modell und sein Verhalten verlässlich zu reproduzieren und Modellversionen mit spezifischen Features oder Fehlern voneinander abzugrenzen. Die Archivierung kann bei der Auswertung von Hardwarefehlern, -schäden oder bei Sicherheitsvorfällen dienlich sein.

3.2.4 Betrieb

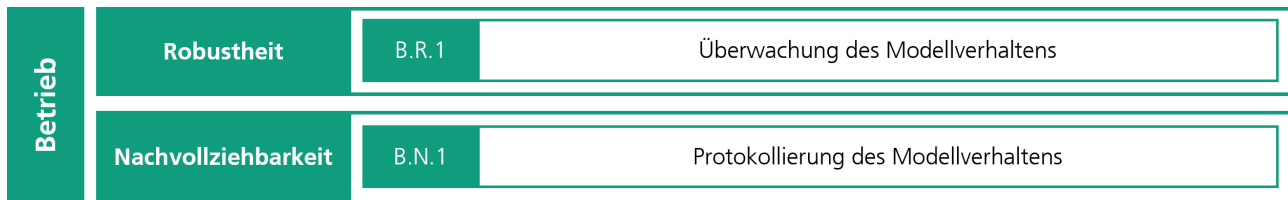


Abbildung 6: Maßnahmen in der Phase Betrieb.

B.R.1 Überwachung des Modellverhaltens

In der Studie werden eingefrorene Modelle betrachtet, also Modelle, die nach ihrer Inbetriebnahme nicht mehr "dazulernen", sondern ihr getestetes Verhalten beibehalten. Allerdings verändert sich die Realumgebung, in welchem sich das Modell befindet, stetig. Dies hat zur Folge, dass die Anwendungsdomäne nicht mehr (vollständig) dem Anwendungskontext, beispielsweise dem bekannten Schiffsaussehen, entspricht [28]. In diesem Fall kommt es zu einem sogenannten Concept Drift. Auch können Umwelteinflüsse Auswirkung auf die Sensorik des Systems haben und die Performanz des Modells signifikant beeinflussen, z.B. aufgrund von verschmutzten Kameralinsen. Daher ist es notwendig die Eingaben und Ausgaben des Modells zu überwachen [28, 39, 31].

B.N.1 Protokollierung des Modellverhaltens

Für spätere Untersuchungszwecke sollte eine Protokollierung des Modellverhalten stattfinden, so dass das Modellverhalten nach dem EVA-Prinzip erklärbar ist. Die Inhalte der Protokolle sollten innerhalb eines realisierbaren Speicherbedarfs die wichtigsten Informationen beinhalten, um (Fehl-)Verhalten eines Modell rekonstruieren zu können. Der einfachste Schritte wäre bei überschaubarem Speicherbedarf die paarweise Erfassung von Eingangs- und Ausgangsdaten. Der Zeitpunkt der Protokollierung kann dabei zufalls- oder ereignisbasiert, letzteres z.B. bei Fehlern, erfolgen [31]. Es ist denkbar, dass für sicherheitsrelevante Funktionen die Protokollierung verpflichtend stattfindet (und nach längerer fehlerfreier Betriebszeit Protokolle zur Speicherbedarfsminderung gelöscht wird). Die Protokolle können den Entwicklern und den Prüfern dienlich sein. Erstere können damit ihr Modell verbessern und letztere können sie im Rahmen einer Prüfung heranziehen.

3.2.5 Außerbetriebnahme

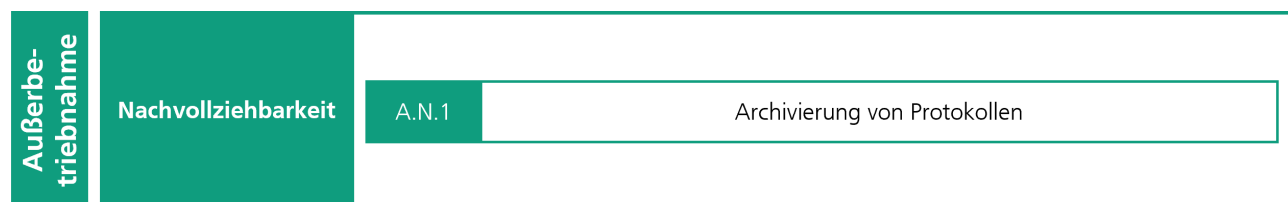


Abbildung 7: Maßnahmen in der Phase Außerbetriebnahme.

A.N.1 Archivierung von Protokollen

Zur Außerbetriebnahme sollten sämtliche aufgezeichnete Protokolle versioniert und archiviert werden [31]. Zur Versionierung der Protokolle könnten analoge Verfahren wie die zur Versionierung von Datensätzen und Modellen (s. Maßnahme [N.1]) verwendet werden [43]. Aus der Versionierung der Protokolle muss hervorgehen, welche Modellversion und wiederum welche Datensatzversion vorliegt.

4 Untersuchung und Vergleich von Objekterkennungsalgorithmen

Objekterkennungsalgorithmen ermöglichen KI-Systemen die Identifizierung und Klassifizierung verschiedener Objekte. Im nautischen Kontext sind das beispielsweise Schiffe, Bojen, Hindernisse und Navigationshilfen. Die genaue Erkennung dieser Objekte ist von grundlegender Bedeutung für die Gewährleistung einer sicheren Navigation, Kollisionsvermeidung und Überwachung des Schiffsverkehrs in Häfen. Dieses Kapitel befasst sich mit der Untersuchung und dem Vergleich von Erkennungsalgorithmen unter Berücksichtigung der Herausforderungen, die die maritime Umgebung mit sich bringt, und dem Vergleich ihrer Leistung mit allgemeineren Anwendungen. Der Hauptaugenmerk der zu erkennenden Objekten gilt Schiffen, da aktuell noch keine hinreichend umfangreiche und annotierte Datensätze für andere nautische Objekte wie Bojen, Hindernissen oder Navigationshilfen existieren. Die Erkennungsalgorithmen werden in der Studie zwischen hinsichtlich der Ziele der Objektdetektion und semantische Segmentierung unterschieden (Vgl. Abbildung 8).

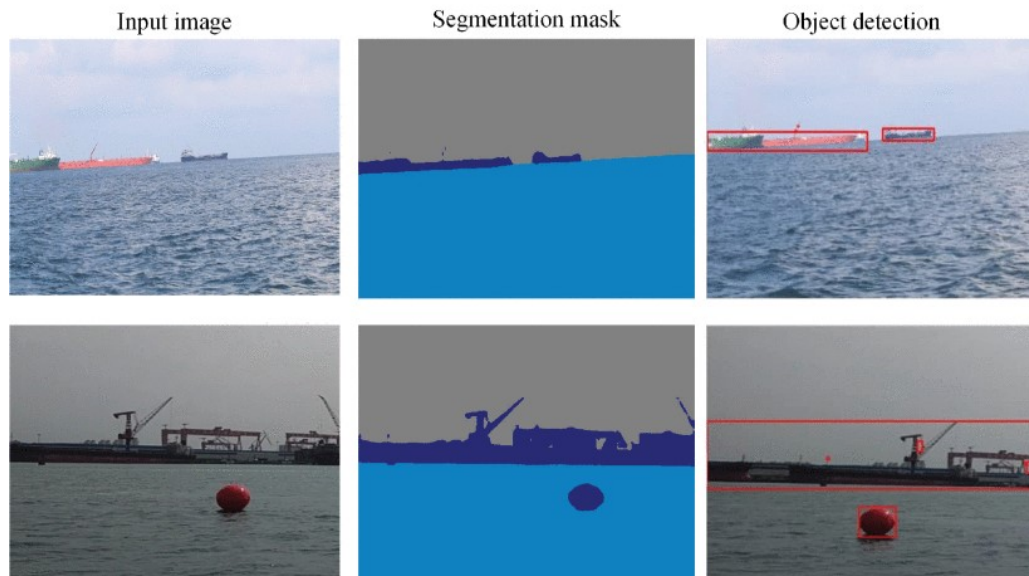


Abbildung 8: Ausgehend von maritimen Bildern (unter „Input image“) Unterscheidung zwischen semantische Segmentierung (unter „Segmentation Mask“) und Objekterkennung (unter „Object detection“) [45].

Im Bereich der Bildverarbeitung ist Objektdetektion eine Aufgabenstellung die sich die Lokalisierung und Klassifizierung mehrerer Objekte innerhalb eines Bildes oder Videoframe konzentriert. Dabei gilt es in der Regel, den Mittelpunkt und die Größe jeder Objektinstanz zu ermitteln, die normalerweise als eine Begrenzungsbox dargestellt wird, und eine entsprechende Klassenbezeichnung zuzuweisen.

Bei der semantischen Segmentierung hingegen handelt es sich um die Aufgabe, jedem Pixel in einem Bild oder Videoframe einer Klasse zuzuweisen und so die Szene in sinnvolle Bereiche zu segmentieren. Semantische Segmentierung ist insbesondere nützlich, um Bereiche statt Objekte zu erkennen, die sich nicht so einfach als eine Begrenzungsbox darstellen lassen. Dieses feinkörnigere Verständnis der Umgebung ermöglicht zusätzliche Anwendungen, wie zum Beispiel die Abschätzung der Horizontlinie und daraus wiederum den Neigungswinkel für den Einsatz in unbemannten Überwasserfahrzeugen.

Beide Aufgabenstellungen stellen ein aktives Forschungsgebiet der Computer Vision dar. Seit einigen Jahren werden diese Aufgabenstellungen von KI-basierten Methoden aus dem Bereich DL dominiert. Insbesondere zwei Klassen von Architekturen haben sich dabei etabliert: CNN und Vision-Transformer.

4.1 Leistungsmetriken

Um zu bewerten wie gut ein KI-System funktioniert, werden Leistungsmetriken verwendet (s. Kapitel 3, Maßnahme K.F.3). Diese Metriken bieten einen standardisierten Rahmen für die quantitative Bewertung und erlauben somit verschiedene Architekturen miteinander zu vergleichen. In ISO/IEC Technical Specification 4213 sind bereits eine Reihe von Metriken definiert, jedoch nur für Varianten der Bild-Klassifikationsaufgabe [36]. Für den Objektdetektionsbereich sind diese nur bedingt einsetzbar, da hier sowohl die Anzahl als auch die Lokalisierung der Objekte einbezogen werden sollte. Hierbei hat sich mean Average Precision (mAP) als wichtigste Metrik etabliert.

Um diese Metrik zu berechnen, müssen Detektionen zuerst in drei Fehlerklassen eingeteilt werden:

- True Positive (TP): Korrekt detektierte Objekte
- False Positive (FP): Redundante Detektionen, d.h. Fälle, in denen der Algorithmus Objekte sieht, die tatsächlich nicht vorhanden sind
- False Negative (FN): Ausgelassene Objekte, d.h. Objekte, die vom Algorithmus nicht erkannt wurden, obwohl sie tatsächlich vorhanden sind.

Diese drei Fälle sind in Abbildung 9 illustriert.

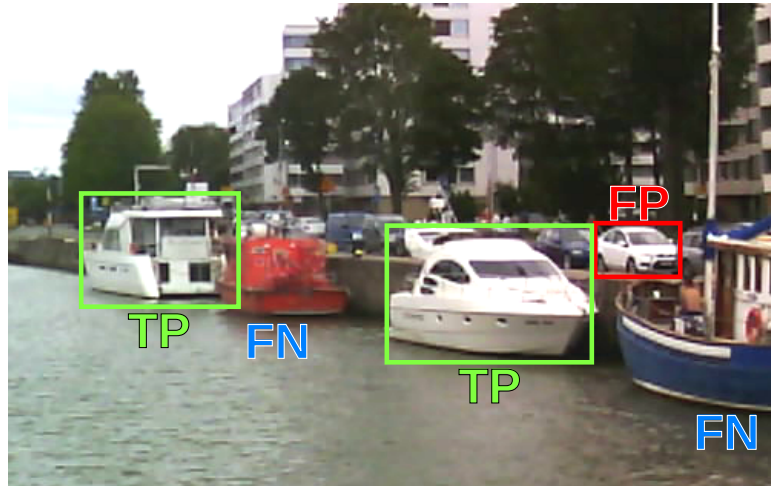
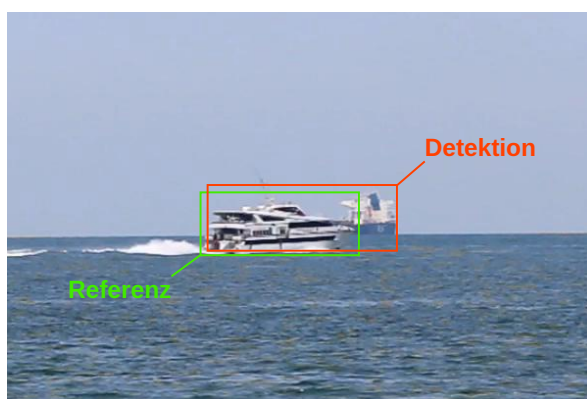


Abbildung 9: Beispielbild mit drei Detektionen im Kontext der Bootsdetektion, dargestellt als Begrenzungsboxen. Es wurden zwei Boote korrekt erkannt (TP, grün) mit einer zusätzlichen inkorrekten Detektion (FP, rot). Zwei Boote wurden übersehen (FN, blau). Die Wiederfindungsrate für diesen Fall ist somit $\frac{2}{4} = 50\%$ und die Präzision ist $\frac{2}{3} \approx 66\%$.

Die Einteilung in eine dieser drei Fälle ist nicht ganz trivial, da die Korrektheit einer Detektion nicht binär ist. Es müssen die Größe und Position der detektierten Objekte mit denen der Referenzboxen mit dem Ähnlichkeitsmaß Intersection over Union (IoU) verglichen werden. Dieses Maß ist auch in ISO/IEC Technical Specification 4213 (6.5.4) als *Jaccard Index* erwähnt, jedoch nur für diskrete Mengen und nicht für Boxen [36]. IoU für Boxen ist definiert als Schnittfläche (Intersection) geteilt durch Vereinigung (Union) von zwei Flächen und drückt aus wie ähnlich sich zwei Boxen sind. Der Wert befindet sich im Bereich zwischen 0 und 1, wobei 1 einer perfekten Übereinstimmung entspricht. Visuell illustriert ist die Berechnung des IoU Ähnlichkeitsmaßes in Abbildung 10. Dieser Wert dient anschließend als Schwellenwert für die Entscheidung ob eine detektierte Box als korrekt oder falsch bewertet wird.



$$\text{IoU} = \frac{\text{Intersection}}{\text{Union}} = \frac{9210 \text{ px}^2}{14064 \text{ px}^2} = 0.65$$

Abbildung 10: Illustration der Berechnung des Ähnlichkeitsmaßes Intersection over Union (IoU). Der Detektor hat das Boot im Vordergrund und das Schiff im Hintergrund als ein einziges Objekt erkannt. Die detektierte Box weicht somit signifikant von der Referenzbox ab und hat eine mittelmäßige IoU-Ähnlichkeit von 0.65.

Der mAP Wert wird letztlich durch die Kombination von Precision (deutsch: Präzision) und Recall (deutsch: Wiederfindungsrate) berechnet. Precision ist das Verhältnis der korrekten Detektionen zur Gesamtanzahl der Detektionen. Anders ausgedrückt, misst Precision, welcher Anteil der detektierten Objekte tatsächlich korrekt sind.

Recall hingegen ist das Verhältnis der korrekten Detektionen zur Gesamtanzahl der tatsächlich vorhandenen Objekte. Recall quantifiziert, wie viele der Objekte vom System erkannt wurden. Mathematisch sind Precision (P) und Recall (R) wie folgt definiert:

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN}$$

Die mAP kombiniert Precision und Recall, indem sie die Precision-Recall-Kurve berechnet und den Durchschnitt der Precision-Werte für verschiedene Recall-Schwellenwerte bildet. Diese Kurve zeigt, wie sich Precision und Recall in Abhängigkeit voneinander verhalten. mAP entspricht letztlich der Fläche unter dieser Kurve (Area Under Precision-Recall Curve (AUPRC), siehe auch ISO/IEC Technical Specification 4213 (6.3.7) [36])

Die mAP-Metrik bietet eine ganzheitliche Bewertung der Leistung von Objektdetektionsalgorithmen, die sowohl die Genauigkeit der Detektionen als auch deren Positionierung und Größenanpassung berücksichtigt. Je nach Anwendung, kann auch die umgekehrte Metrik mean Average Recall (mAR) relevant sein.

Für die semantische Segmentierung ist mean Intersection over Union (mIoU) die am meisten genutzte Metrik zur Bewertung der Leistung von Algorithmen. Das m in mIoU steht für das Mittel einzelner IoU Werte über die vorhandenen Objektklassen. Die Berechnung von IoU ist ähnlich zu der im Objektdetektionsbereich aber basiert auf Mengen von einzelnen Pixeln und nicht auf Flächeninhalten von Boxen. Dieses Maß entspricht dem aus ISO/IEC Technical Specification 4213 (6.5.4) [36]. lässt sich mathematisch wie folgt ausdrücken:

$$\text{IoU} = \frac{A \cap B}{A \cup B}$$

wobei A und B Mengen von detektierten und tatsächlichen Pixeln einer Objektklasse sind.

4.2 Vergleich von Architekturen

Im folgenden werden fünf Architekturen aus dem Objektdetektionsbereich vorgestellt und verglichen, die sich in allgemeinen Anwendungen als erfolgreich erwiesen haben. Dieser Vergleich kann als Teil der Maßnahme E.F.2 (s. Kapitel 3) betrachtet werden.

Objektdetektion

- **Faster R-CNN** [46]
Faster Region-CNN (Faster R-CNN) ist eine fest etablierte und weitverbreitet eingesetzte Architektur für Objektdetektion und häufig als Ausgangspunkt für Vergleiche verwendet. Sie besteht aus zwei Bearbeitungsstufen. Die erste Stufe, das Region Proposal Network (RPN), schlägt potenzielle Positionen für Objekte vor. Anschließend prüft der zweite Teil, ein Klassifikator und ein Bounding-Box-Regressor, ob sich tatsächlich ein Objekt an den vorgeschlagenen Stellen befindet, und verfeinert die Position des Objekts. Dieser zweistufige Prozess ermöglicht Faster R-CNN, Objekte im Vergleich zu vorherigen Architekturen genauer und effizienter zu finden.
- **RetinaNet** [47]
RetinaNet ist eine Verbesserung von Faster-R-CNN, die darauf abzielt, das Klassen-Ungleichgewicht zu lösen. Das bedeutet, dass i.d.R. einige Arten von Objekten häufiger vorkommen als andere, was ein Neuronales Netzwerk (NN) verwirren kann. RetinaNet behandelt dieses Problem, indem es besonderen Fokus auf seltene Objekte legt, sodass sie nicht übersehen werden.
- **DETR** [48]
Detection Transformer (DETR) ist die erste Architektur im Bereich der Objektdetektion, die auf *Transformer*-Netzwerken basiert. Transformer wurden erstmals im Jahr 2017 für die Sprachverarbeitung eingeführt und seitdem diesen Bereich komplett revolutioniert. Seit wenigen Jahren werden sie auch in der Bildverarbeitung zunehmend verwendet. Anders als CNN, die nur einen Bildausschnitt nach dem anderen betrachten und somit ein begrenztes Sichtfeld haben, können Transformer globale Zusammenhänge und Abhängigkeiten im ganzen Bild auf einmal erfassen. Ein großer Vorteil von DETR besteht darin, dass es *end-to-end* trainiert werden kann. Das bedeutet, dass das Modell vom Eingabebild bis hin zur Ausgabe keine manuellen Zwischenschritte oder Nachbearbeitung benötigt, wodurch Bedienung und Programmieraufwand deutlich erleichtert werden. Im Gegensatz zu vorherigen Ansätzen verwendet DETR keine von Hand entworfene Komponenten wie Non-Maximum Suppression (NMS) oder eine Ankererzeugung, sondern behandelt die Objekterkennung als ein Set-Vorhersageproblem.

- **FocalTransformer** [49]

FocalTransformer ist eine weitere Architektur aus der Klasse der Vision Transformer, die insbesondere den Rechenaufwand bei hochauflösenden Bildern verringert. Er führt einen neuen Mechanismus ein, der sowohl feine lokale als auch grobe globale Interaktionen umfasst, um visuelle Abhängigkeiten sowohl im Nah- und Fernbereich effizienter zu erfassen. FocalTransformer ist an sich kein eigenständiger Objektdetektor, sondern eine generische Backbone, die für Objektdetektion erweitert werden kann.

- **YOLOv7** [50] ³

YOLOv7 ist die neueste Version aus der YOLO-Familie von Objektdetektionsarchitekturen. YOLO zielt auf Echtzeitdetektion ab, insbesondere für Anwendungen bei denen eine hohe Bildrate oder Reaktionszeit wichtig ist, beispielsweise beim autonomen Fahren. Während frühere Versionen aus dieser Familie, den größeren und langsameren Architekturen in der Erkennungsleistung deutlich unterlegen waren, konnten nachfolgende Iterationen diesen Nachteil verringern und mittlerweile sogar die größeren Architekturen übertreffen.

Die Rechenaufwand eines Algorithmus spielt eine wichtige Rolle bei der Eignung von Algorithmen. Er wird in der Regel in Floating Point Operations per Second (FLOPs) angegeben und bezieht sich auf die Anzahl der Berechnungen, die ein NN in einer Sekunde mit Gleitkommazahlen durchführt. Es ist eine wichtige Messgröße, um die Leistung und den Ressourcenbedarf eines NN zu bewerten und gibt Aufschluss über die Berechnungskomplexität des Modells und seine Effizienz. Je höher die Anzahl der FLOPs ist, desto mehr Rechenleistung wird benötigt, um das Netzwerk zu trainieren oder Vorhersagen zu treffen. Für Echtzeitanwendungen relevant, beeinflusst das auch die Latenzzeiten.

Die zuvor aufgeführten Architekturen sind in Tabelle 2 mit Metriken evaluiert auf dem Datensatz COCO [51] gegenübergestellt.

Tabelle 2: Vergleich von verbreiteten Architekturen im Objektdetektionsbereich bei allgemeinen Anwendungen.

Architektur	Jahr	Art	Parameter	FLOPs	mAP (COCO)
Faster-R-CNN	2015	CNN	42M	134G	37.0
RetinaNet	2017	CNN	38M	152G	36.4
DETR	2020	Transformer	41M	187G	42.0
FocalTransformer	2021	Transformer	39M	265G	45.5
YOLOv7	2023	CNN	37M	105G	51.4

Semantische Segmentierung

- **DeepLabV3** [52]

DeepLabV3 ist eine Architektur für semantische Bildsegmentierungsaufgaben. Es baut auf dem Erfolg früherer Versionen auf und kombiniert atrous (erweiterte) Konvolution und die Verwendung eines leistungsstarken Encoder-Decoder-Frameworks. DeepLabV3 verwendet ASPP-Module (Atrous Spatial Pyramid Pooling), um mehrskalige Kontextinformationen und -funktionen zu erfassen und so hochpräzise und feinkörnige Segmentierungsergebnisse für eine Vielzahl von Bildern und Szenen zu erzielen. DeepLabV3 ist vergleichsweise einfach aufgebaut und liefert solide Ergebnisse, weswegen es mittlerweile zu einer Standardarchitektur für eine Vielzahl von Anwendungen geworden ist.

- **LR-ASPP** [53]

LR-ASPP (Lite Reduced Atrous Spatial Pyramid Pooling) ist eine auf Echtzeitanwendungen und mobile Endgeräte optimierte Architektur. Das Modell ermöglicht deutlich schnellere Verarbeitung von Bildern und deutlich reduzierte Latenzzeiten bei ähnlicher Genauigkeit als größere Systeme.

- **OCR** [54]

Diese Architektur ist die Erste, die teilweise Transformer für Semantische Segmentierung einsetzt. Im Gegensatz zu CNN sind Transformer gut geeignet um globale Abhängigkeiten innerhalb eines Bildes zu erfassen, was gerade bei Segmentierungsaufgaben von Vorteil ist. Anstatt jeden Pixel einzeln zu klassifizieren, versucht diese Methode erst andere Pixel zu finden die vermutlich zum selben Objekt oder Region gehören. Erst dann werden sie gemeinsam einer Klasse zugeordnet.

³Der offizielle Quellcode für You Only Look Once (YOLO), einschließlich YOLOv7, ist unter den Nutzungsrechten der GNU General Public License (GPL) veröffentlicht.

- **SegFormer** [55]
Anders als OCR, besteht die SegFormer Architektur vollständig aus Transformerbauteilen. Sie führt insbesondere viele Vereinfachungen ein, die zuvor nur umständlich gelöst wurden. Das wirkt sich zudem positiv, sowohl auf die Erkennungsleistung als auch auf die Geschwindigkeit, aus. SegFormer zeigt ein hohes Grad an Robustheit auf, d.h. es kann mit störenden Einflüssen, wie z.B. wechselnden Wetterbedingungen gut umgehen, was auf die Transformerarchitektur zurückgeführt werden kann.
- **ViT-Adapter** [56]
Wie bereits oben erwähnt, haben Vision Transformer (ViT) gegenüber CNN mehrere Vorteile, aber diese kommen auf Kosten von deutlich längerem Trainingaufwand. ViT-Adapter ist eine Methode, die das Training von ViTs erleichtert, indem sie bildbezogenes induktives Vorwissen ausnutzt. Insbesondere ermöglicht ViT-Adapter *multimodales* Training, d.h. kombiniert Bilder mit anderen Arten von Daten, z.B. Text oder Audio.

Die folgende Tabelle 3 stellt diese Architekturen gegenüber, mit Metriken evaluiert auf dem generischen Datensatz ADE20K [57] und dem Datensatz Cityscapes [58] aus dem Automobilbereich.

Tabelle 3: Vergleich von verbreiteten Architekturen im Bereich der semantischen Segmentierung bei allgemeinen Anwendungen.

Architektur	Jahr	Art	Parameter	FLOPs	mIoU (ADE20K)	mIoU (Cityscapes)
DeepLabV3	2017	CNN	42M	10.4G	44.1	80.9
LR-ASPP	2019	CNN	3.2M	3.2G	-	72.4
OCR	2020	Transformer	10.5M	340G	45.3	82.4
SegFormer	2021	Transformer	44.0M	79G	50.0	83.3
ViT-Adapter	2023	Transformer	133.9M	403G	52.5	-

4.3 Training

Alle oben genannten Architekturen basieren auf NN. Bevor sie verwendet werden können, müssen sie durch einen iterativen Lernprozess auf einen Datensatz trainiert werden. Dieser Prozess zielt darauf ab, die Fehlerabweichung zwischen der vorhergesagten Ausgabe des Netzwerks und der tatsächlich gewünschten Ausgabe zu minimieren. Effektives Training neuronaler Netze erfordert die Einhaltung mehrerer Best Practices und Vermeidung einiger Tücken. Die hier durchgeführten Prozesse können als Teile der Maßnahmen E.F.1 und E.F.2 (s. Kapitel 3) betrachtet werden.

Beispielsweise tritt *Overfitting* (Überanpassung) auf, wenn ein ML-Modell die Trainingsdaten auswendig lernt und somit die Fähigkeit verliert, auf neuen, unbekanntenen Daten korrekte Vorhersagen zu treffen. Dies passiert, wenn das Modell irrelevante Muster aus den Trainingsdaten erfasst und diese Muster nicht auf andere Daten verallgemeinert werden können. Overfitting beeinträchtigt die Performanz unter Real-Bedingungen (s. Kapitel 3, Maßnahme R.1).

Um das Ausmaß des Overfitting abzuschätzen, werden Daten in ein Trainings- und Validierungs- und Testset aufgeteilt (s. International Standard ISO/IEC IS 22989 Kapitel 5.11.8 [34]). Das Trainingsset wird verwendet, um das Modell zu trainieren. Die Validierungsdaten werden während und nach dem Training verwendet, um Hyperparameter (z.B. die Auswahl der Architektur) zu optimieren. Das Testset sind Daten, die das Modell während des Trainings nicht gesehen hat. Es wird letztlich verwendet, um die Leistung des Modells zu evaluieren. Durch die Aufteilung der Daten kann sichergestellt werden, dass die Genauigkeit eines Modells dem Einsatz unter Realbedingungen entspricht.

Um Overfitting in Grenzen zu halten, ist es etablierte Praxis *Datenaugmentierung* anzuwenden. Dabei werden die vorhandenen Trainingsdaten modifiziert oder umgeformt, um neue Variationen zu erzeugen. Häufig eingesetzte Transformationen sind hier z.B. Drehen, Skalieren, Spiegeln oder Hinzufügen von Rauschen. Datenaugmentierung hilft dabei, Overfitting zu reduzieren, indem es dem Modell mehr Beispiele aus verschiedenen Blickwinkeln und Variationen bietet und somit robustere Muster antrainiert.

Transfer Learning ist eine Technik, bei der das Wissen eines vorab trainierten Modells auf eine neue, verwandte Aufgabe angewendet wird. Dieses vortrainierte Modell wird in der Regel auf einen großen, generischen Datensatz wie ImageNet [59] oder COCO [51] trainiert und anschließend auf den kleineren Zieldatensatz für die neue Aufgabe geringfügig angepasst. Anstatt bei Null anzufangen, nutzt das Modell die gelernten Darstellungen aus der ursprünglichen Aufgabe und passt sie anhand der neuen Daten an. Transfer Learning kann erheblich Zeit und

Datenverbrauch reduzieren und führt oft zu einer verbesserten Leistung, da das Modell ein besseres Verständnis der neuen Aufgabe gewinnt, indem es auf seinem vorhandenen Wissen aufbaut. Diese Technik wird auch in ISO/IEC IS 22989 (5.11.5) [B4] erwähnt und in ISO/IEC 23053 [B5] genauer beschrieben

4.4 Auswertung für Anwendungen im maritimen Umfeld

In diesem Kapitel werden die gewonnenen Ergebnisse der Objektdetektion und semantischer Segmentierung unter Verwendung einzelner vorgestellter Architekturen (s. Kapitel 4.2) und Datensätzen diskutiert. Dieser Teil der Arbeit kann als Teil der Maßnahmen E.R.1 und E.N.1 (s. Kapitel 3) gesehen werden.

Objektdetektion

Für die Objektdetektion wurden die Architekturen Faster-R-CNN, DETR und YOLOv7 ausgewählt um ein breites Spektrum zu erfassen und zum Teil aus Gründen der Verfügbarkeit des Quellcodes.

Der Singapore Maritime Dataset (SMD) [50] ist ein öffentlich zugänglicher Datensatz für maritime Anwendungen. Er umfasst insgesamt 81 Videos von Schifffahrtsaktivitäten, aufgenommen in der Umgebung des Hafens in Singapur. Der Datensatz stellt zudem Annotationen zur Verfügung die für das Training von KI-Systemen verwendet werden können für Einsatzzwecke, wie z. B. Schiffsnachverfolgung, Verkehrsanalyse und Sicherheitsmaßnahmen.

ABOships [51] ist ein weiterer Datensatz, aufgenommen aus einem fahrendem Schiff im Hafen von Turku in Finnland. Er besteht aus etwa 10.000 Bildern und 42.000 annotierten Schiffen. Im Unterschied zu SMD, enthalten die Bilddaten deutlich mehr Schiffe und andere Objekte die Schiffen ähneln (z.B. Hafenkräne oder Straßenfahrzeuge), was zu deutlich gesteigerten Anforderungen an den Objektdetektor führt.

Die meisten Autoren stellen Modelle zur Verfügung die auf dem COCO Datensatz trainiert wurden. Da dieser Datensatz auch eine Klasse für Schiffe enthält, können diese Modelle theoretisch sofort ohne weiteres Training direkt eingesetzt werden. Wie gut das letztlich funktioniert und ob eine Anpassung an das spezifische Einsatzszenario durch Nachtrainieren nötig ist wird hier ebenfalls untersucht.

Die Ergebnisse der Evaluierung sind in der Tabelle 4 zusammengefasst und zeigen, dass YOLOv7 den anderen Algorithmen in den meisten Metriken überlegen ist. Hinzu kommt, dass YOLOv7 im Gegensatz zu Faster-R-CNN und DETR echtzeitfähig ist.

Tabelle 4: Hauptergebnisse der Evaluierung von Objektdetektionsarchitekturen. Aufgeführt sind hierbei die evaluierten Architekturen ohne und mit nachträglichem Training. Nachträgliches Training und die Evaluierungen fanden unter Verwendung der aufgeführten Datensätze SMD oder ABOships statt. mAP_L steht für die mAP Metrik angewandt nur auf große Objekte, mAP_S für kleine Objekte.

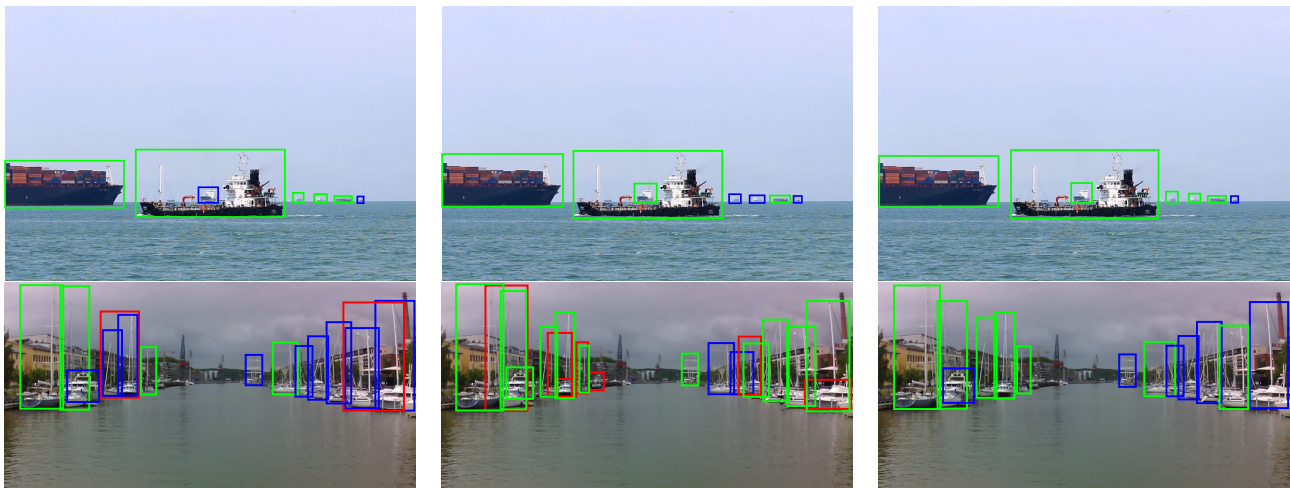
Architektur	Training	SMD				ABOships			
		mAP	mAP_L	mAP_S	mAR	mAP	mAP_L	mAP_S	mAR
Faster-R-CNN	COCO	27.3	55.9	0.1	33.9	13.0	20.9	8.0	24.3
Faster-R-CNN	Nachtrainiert	33.9	56.1	6.7	41.3	29.0	45.6	18.5	37.6
DETR	COCO	21.9	47.2	0.0	29.2	7.9	8.4	7.7	25.6
DETR	Nachtrainiert	34.7	61.2	0.3	39.6	27.1	48.7	13.8	39.6
YOLOv7	COCO	28.8	60.5	0.0	31.6	11.9	26.5	3.4	15.7
YOLOv7	Nachtrainiert	41.0	63.9	7.7	46.5	31.9	52.3	17.6	39.0

Ein Vergleich mit typischen Fehlern ist dargestellt in Abbildung 11.

Semantische Segmentierung

Maritime Semantic Segmentation Training Dataset (Mastr) [52] ist ein Datensatz zur semantischen Segmentierung im Meer, zugeschnitten auf die Entwicklung von Methoden zur Hinderniserkennung in kleinen Küsten-USVs (Unmanned Surface Vehicle). Er enthält 1.325 Bilder, die über einen Zeitraum von zwei Jahren mit einem echten USV an der Küste Sloveniens aufgenommen wurden und eine Reihe realistischer Bedingungen und Hindernissen abdecken, die im Kontext der Küstenüberwachung auftreten können.

Für diese Aufgabenstellung gibt es keine passende vortrainierte Netzwerke, also muss auf das Einsatzszenario trainiert werden. Tabelle 5 zeigt die Ergebnisse der Evaluierung. Einzelne repräsentative Beispielbilder sind dargestellt in Abbildung 12.



(a) Faster-R-CNN

(b) DETR

(c) YOLOv7

Abbildung 11: Visueller Vergleich der Faster-R-CNN, DETR und YOLOv7 Architekturen auf jeweils ein Beispielbild aus dem Singapore Maritime Dataset (oben) und aus AboShips (unten) sowie typische Fehler der Architekturen. Grüne Boxen sind korrekt erkannte Objekte (True Positives), blaue Boxen sind nicht erkannte Objekte (False Negatives) und rote Boxen sind Falsch-Positive. Faster-R-CNN zeigt häufig Schwächen bei überlappenden oder nah beieinander liegenden Objekten und erkennt diese häufig als ein einziges Objekt. DETR neigt zum umgekehrten Problem, d.h. generiert viele Falsch-Positive Detektionen und hat insbesondere bei kleinen, weit entfernten Objekten am Horizont Probleme. YOLOv7 zeigt insgesamt weniger Schwächen, scheitert aber auch an den schwierigsten Stellen.

Alle untersuchten Architekturen liefern eine versprechende Erkennungsleistung für diesen Einsatzzweck. LR-ASPP setzt den Fokus auf Geschwindigkeit und büßt damit an Genauigkeit ein. Falls genug Rechenleistung vorhanden ist, sollte eher auf DeepLabV3 oder SegFormer zurückgegriffen werden.

Tabelle 5: Hauptergebnisse der Evaluierung von Architekturen für Semantische Segmentierung. Neben mIoU ist sind hier auch die Werte für die einzelnen Klassen (Wasser, Himmel, Hindernisse) enthalten.

Architektur	mIoU	IoU _{water}	IoU _{sky}	IoU _{obstacles}
DeepLabV3	95.2	99.3	99.4	88.7
LR-ASPP	94.0	97.7	99.5	86.1
SegFormer	96.4	99.3	99.6	91.2

4.5 Vergleich zu generischen Anwendungen

Verglichen mit dem generischen COCO Datensatz, sind die Leistungsmetriken der Objekterkennung für die maritimen Anwendungen deutlich geringer. Mögliche Gründe liegen in der Natur der Bilder aus diesem Bereich. Maritime Umgebungen unterliegen dynamischen und vielvältigen Wetterbedingungen wie Nebel, Regen und Blendung, die die Bildqualität erheblich beeinträchtigen und häufig nur schwer in einem einzigen Trainingsdatensatz erfasst werden können. Der herausfordernde Charakter fällt umso mehr auf, wenn man bedenkt, dass die Objekte weit entfernt am Horizont liegen und somit sehr klein erscheinen können: Selbst bei einer hohen Bildauflösung in 4K misst ein Schiff am Horizont nur wenige Pixel, was die Identifizierung und Lokalisierung zusätzlich erschwert.

In Bezug auf die semantische Segmentierung zeigt diese Studie signifikant bessere Leistungsindikatoren im Vergleich zu dem generischen Datensätzen ADE20K oder dem aus dem Automobilbereich stammenden Cityscapes-Datensatz. Dies lässt sich zum Teil durch die geringere Anzahl von Klassen in dem maritimen Datensatz erklären, der nur drei Klassen (Wasser, Himmel, Küstenlinie und Hindernisse) enthält. Diese Klassen sind relativ groß und gut definiert in den Bildern, was diese spezifische Segmentierungsaufgabe weniger komplex macht. Dennoch ist zu betonen, dass es herausfordernden Beispiele mit widrigen Bedingungen, wie Nebel, starkem Regen oder starkem Sonnenlicht gibt, die zeigen, dass die erzielten Ergebnisse nicht pauschal auf andere maritime Szenarien übertragen werden können.

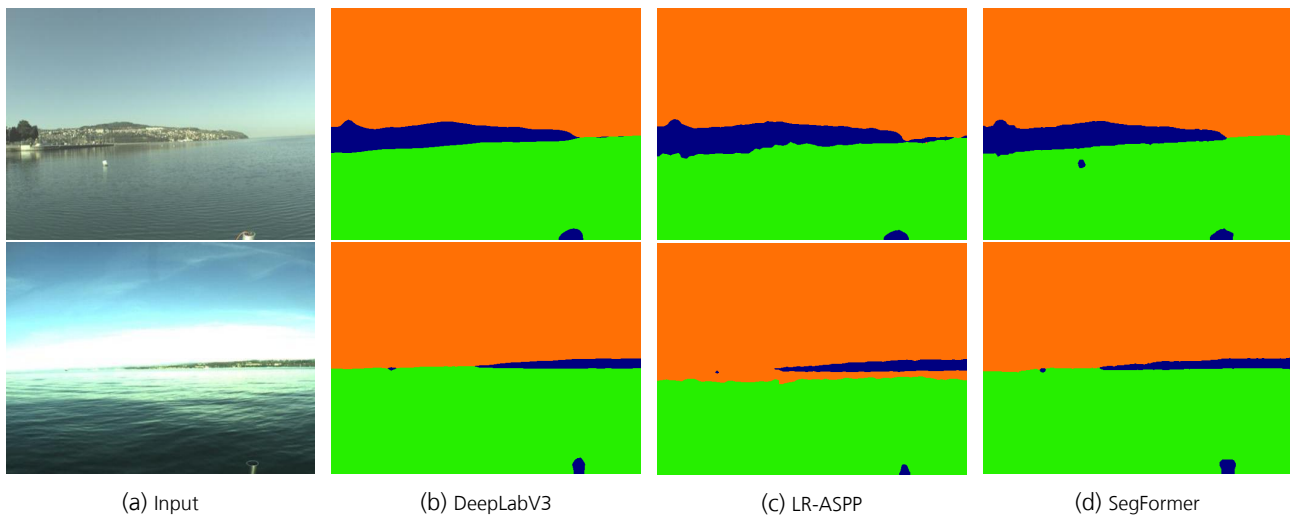


Abbildung 12: Visueller Vergleich der DeepLabV3, LR-ASPP und SegFormer Architekturen auf zwei Beispielbilder aus dem Mastr Datensatz. Die erkannten semantischen Klassen sind visualisiert in den Farben Orange (Himmel), Grün (Wasser) und Blau (Küste oder Hindernisse). DeepLabV3 und SegFormer liefern ähnlich gute Ergebnisse, wobei SegFormer in der Regel etwas genauer ist. Beispielsweise übersieht DeepLabV3 im ersten Bild ein kleines Hindernis. LR-ASPP zeigt insbesondere bei schwierigen Wetterbedingungen (z.B. starkem Lichteinfall) Schwächen, kann aber für Spezialanwendungen ausreichend sein bei denen Geschwindigkeit eine hohe Priorität hat.

Die Resultate verdeutlichen, dass die untersuchten Modelle, obwohl sie primär für andere Anwendungsbereiche entwickelt worden sind, durch gezieltes Training und Anpassungen in der Lage sind, eine akzeptable Leistung für spezialisierte maritime Anwendungen zu erzielen. Dies unterstreicht die Flexibilität und das Potenzial der KI-Methoden, sich an anspruchsvolle Domänen anzupassen, wenn sie richtig bedient werden.

5 Methoden zur Erstellung eines Datensatzes für die Entwicklung von maritimen Objekterkennungsalgorithmen

Für das Training der in Kapitel 4 eingeführten Algorithmen ist die Beschaffung von geeigneten Bilddaten eine grundlegende Voraussetzung. Im Folgenden werden die Art der Daten sowie Anforderungen und Möglichkeiten zur Beschaffung aufgeführt und erläutert. Ziel dieses Kapitels ist es Eigenschaften von Bilddaten (real und synthetisch) sowie Herausforderungen in der Beschaffung und Nutzung dieser Daten herauszuarbeiten. Die Erkenntnisse aus diesem Kapitel sollen helfen Probleme und möglich Schwachstellen in der Nutzung bestimmter Daten frühzeitig zu identifizieren und darauf reagieren zu können.

Realdaten bezeichnen in diesem Zusammenhang Bilddaten, die in der Regel im Rahmen von Testkampagnen (z.B. an Bord eines Schiffes) entweder selbst oder von Dritten erhoben und zur freien Verfügung gestellt werden. Zusammengefasst zeichnen sie sich dadurch aus, dass sie mit Hilfe von Kamerasystemen aufgenommen wurden. Synthetische Daten hingegen sind künstlich erzeugte Bilddaten, die entweder reale Datensätze ergänzen oder eigenständige künstliche Datensätze bilden.

5.1 Einordnung der Datenbeschaffung und -verwendung in Lebenszyklusmodell

Die Betrachtung des LZM (s. Kapitel 3) zeigt, dass Datenbeschaffung und -verwendung insbesondere in den Phasen Konzeption, Entwicklung und Betrieb eine zentrale Rolle spielen. Diese Einordnung ist in Abbildung 13 illustriert.

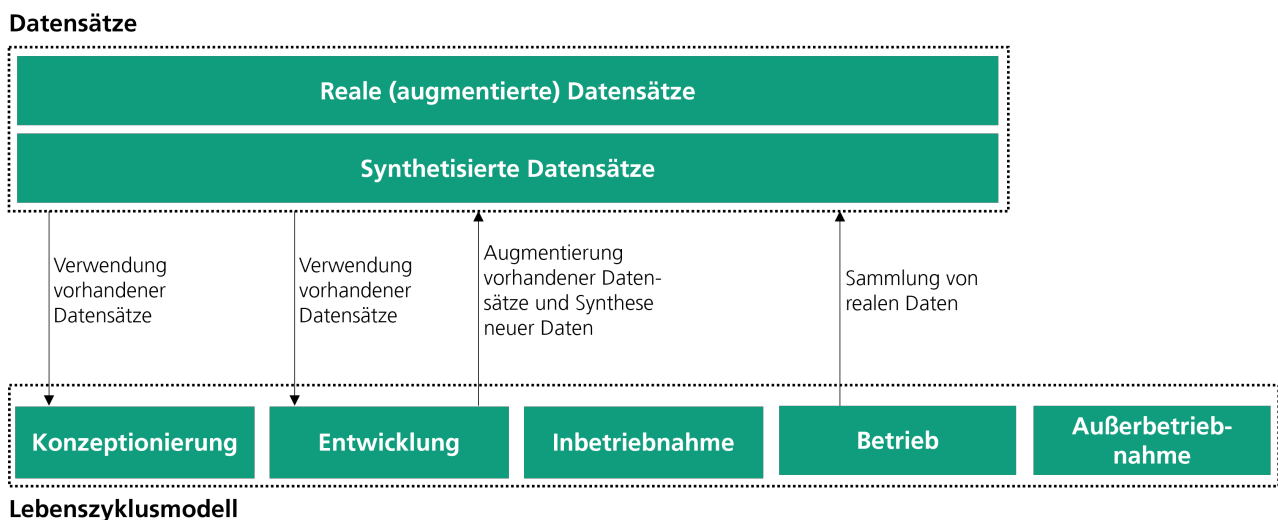


Abbildung 13: Einordnung der Datenbeschaffung und -verwendung in das Lebenszyklusmodell.

Die benötigten Bilddaten dienen in der Konzeptionsphase als Grundlage für die Bewertung der Machbarkeit zur Implementierung eines Objekterkennungs- oder Segmentierungsalgorithmus, welcher auf ML basiert, und in der Entwicklungsphase als Trainingsgrundlage, um das Modell mit den benötigten Bilddaten zu trainieren. Die Phase des Betriebs dient der Sammlung weiterer Bilddaten um Anpassungen an den Modellen für domänenspezifische Problemstellungen durchzuführen oder um sie für andere Anwendungen zu verwenden. Anpassungen hinsichtlich der Verteilung der Objektklassen in den Datensätzen können auch unter Verwendung synthetischer Daten genutzt werden, indem die Datensätze augmentiert werden.

Reale, synthetische oder gemischte Datensätze liefern unterschiedliche Trainingsergebnisse, weshalb sie grundlegend differenziert zu betrachten sind (vgl. [63]). Im Folgenden wird auf systematische Gemeinsamkeiten und spezifische Anforderungen von realen und synthetischen Daten eingegangen.

5.2 Allgemeine Anforderungen an Bilddaten

Bilddaten sind eine grundlegende Form visueller Informationen, die in der Computergrafik und insbesondere bei der Entwicklung von ML-Algorithmen zur Objektdetektion und semantischen Segmentierung als hauptsächliche Datenquelle dienen. Bilddaten bestehen aus einer Sammlung von Datenpunkten, spezifischer Pixelinformationen eines Kamerasystems oder synthetischen Algorithmus, deren Anordnung das Bild darstellen und verschiedene Szenen oder Objekten erfassen und entsprechend abbilden.

Bilddaten können in verschiedenen Formen vorliegen. Zwei wesentliche Darstellungsvarianten sind Graustufenbilder und Farbbilder. Graustufenbilder enthalten nur Helligkeitsinformationen, während Farbbilder zusätzlich Farbinformationen enthalten, die typischerweise in den drei Farbkanälen Rot, Grün, Blau (RGB) kodiert sind. Diese Daten werden in geeigneten Dateiformaten gespeichert und können von Kameras oder Algorithmen erzeugt werden. Im Bereich des ML werden die Farbwerte von Bildern häufig in Grauwertbilder umgewandelt, da Farben in der Regel nur einen geringen Einfluss auf die Objekterkennung haben wohingegen Formen, Kontrastunterschiede und Perspektiven die entscheidenden Faktoren für die Objekterkennung [64] darstellen.

Die wichtigsten Anforderungen an reale und synthetische Bilddaten (vgl. [65]) lassen sich in vier Eigenschaften unterteilen, die nachfolgend aufgeführt und anschließend am Beispiel der realen Bilddaten tiefer erläutert werden:

1. **Repräsentativität:** Die erfassten Bilddaten sollten für die Zielanwendung geeignet sein. Dies bedeutet, dass die Bilddaten ein breites Spektrum von Szenarien, Bedingungen und Variationen abdecken sollten, die der erwarteten Verteilung der Daten innerhalb der gewählten Anwendungsdomäne entsprechen. Dies ermöglicht es den Modellen, Muster und Merkmale zuverlässig zu erkennen. Die Quantifizierung der Repräsentativität erfolgt durch eine Erfassung der Klassenverteilung in den annotierten Bilddaten.
2. **Qualität:** Qualität definiert sich unter anderem durch eine korrekte Belichtung, Schärfe und Farbwiedergabe. Verzerrte oder unscharfe Bilder können die Leistung von maschinellen Lernverfahren beeinträchtigen. Eine Quantifizierung der Qualität kann jedoch helfen ML-Modelle unter Berücksichtigung verschiedener Qualitätsstufen der genutzten Bilddaten robuster zu gestalten.
3. **Metadaten:** Metadaten sind in Bilddaten eingebettete Zusatzinformationen, die wichtige Kontextinformationen wie Aufnahmeort, Aufnahmezeitpunkt oder Kameraeinstellungen enthalten. Metadaten können bei der Organisation von Bilddaten helfen, da sie wichtige Parameter für die Vergleichbarkeit verschiedener Datensätze enthalten.
4. **Annotation:** Annotationen sind hinzugefügte Informationen, die bestimmte Objekte, Regionen oder Merkmale in einem Bild identifizieren oder beschreiben. Dazu gehören z. B. die Position oder Flächenbedeckung von Objekten, ihre Kategorie und weitere anwendungsabhängige Attribute.

5.3 Aggregation und Sammlung von realen Bilddaten

Die in Kapitel 4 betrachteten Architekturen wurden auf Datensätzen trainiert, welche in der maritimen Domäne verbreitet sind und entsprechende Bilddaten und Annotationen enthalten. Die genutzten Datensätze fokussieren sich dabei auf die Objekterkennung und Segmentierung und werden im folgenden kurz vorgestellt. Zusätzlich wird ein Datensatz des Fraunhofer CML eingeführt welcher neben einer hohen Bandbreite an Objekten auch zeigt welche Herausforderungen in der Beschaffung von realen Bilddaten auftreten können.

5.3.1 Evaluierung bestehender Datensätze für die maritime Objekterkennung

Im Rahmen der durchgeführten Untersuchungen für die Nutzbarkeit verschiedener ML Modelle in der maritimen Domäne wurde in Kapitel 4 der Fokus auf drei Datensätze (siehe Tabelle 6) gelegt die über eine ausreichende Anzahl an Annotationen für die Erstellung eines Objektdetektors und Segmentierungsmodells verfügen. Zu Vergleichszwecken ist ein interner Bilddatensatz des Fraunhofer CML *CML Schiffsdatenbank mit AIS-Daten* in der Tabelle mit aufgeführt.

Tabelle 6: Auflistung genutzter Datensätze für den Vergleich der Leistungsfähigkeit untersuchter ML-Modelle.

Name	Inhalt	Verwendung	Aufnahmeort
Singapore Maritime Dataset [60]	81 Videos	Objektdetektion	Singapur
ABOships Inshore and Offshore Maritime Vessel Detection Dataset [61]	ca. 10.000 Bilder	Objektdetektion	Finnland
MaSTr dataset for training deep USV obstacle detection models [62]	1.325 Bilder	Semantische Segmentierung	Slowenien
CML Schiffsdatenbank mit AIS-Daten	ca. 1,2 Millionen Bilder	Objektdetektion	Hamburger Hafen

Die in Kapitel 4 beschriebenen Datensätze wurden bereits im Rahmen der Studie für Untersuchungen der Nutzbarkeit herangezogen und entsprechend ausgewertet. Die Auswertung hat gezeigt das sich entsprechend den

gewählten Use-Cases für die Objekterkennung und semantische Segmentierung die aufgeführten Datensätze die notwendigen Trainingsdaten enthalten um solche Modelle zu entwickeln.

- **Singapore Maritime Dataset:** Dieser Datensatz umfasst insgesamt 81 Videos, die aus verschiedenen Perspektiven und in unterschiedlichen maritimen Szenarien in Singapur aufgenommen wurden. Die Daten enthalten neben RGB Bilddaten auch Infrarot Aufnahmen welche jedoch im Kontext der Untersuchungen dieser Studie nicht weiter betrachtet wurden.
- **ABOships Inshore and Offshore Maritime Vessel Detection Dataset:** Der ABOships-Datensatz enthält ca. 10.000 Bilder. Diese Bilder wurden in den Küsten- und Offshore-Gewässern Finnlands aufgenommen und enthalten eine breite Palette von maritimen Szenarien. Der ABOships-Datensatz enthält Klassen zur Erkennung von Schiffen, Bojen und anderen maritimen Strukturen. [61]
- **MaSTr dataset for training deep USV obstacle detection models:** Der MaSTr Datensatz besteht aus 1325 Bildern und bietet eine Sammlung von maritimen Szenarien in den Gewässern von Slovenien. [62]
- **CML Schiffsbilddatenbank mit AIS Daten:** Der CML Datensatz zur Erstellung von Objekterkennungsalgorithmen besteht aus Bildern verschiedener Webcams die über den Hamburger Hafen verteilt sind. Dieser Datensatz besteht aus Bilddaten und annotierten AIS Daten welche durch einen am CML entwickelt Fusionsalgorithmus erzeugt wurden. Die Daten umfassen etwa 1,2 Millionen Bilder über einen Zeitraum von 6 Monaten. Die Methode zur Erstellung des Datensatzes wird in einem Anfang 2024 veröffentlichten Paper mit dem Titel (eng.) *Image and AIS Data Fusion Technique for Maritime Computer Vision Applications* vorgestellt.

5.3.2 Herausforderungen in der Nutzung von realen Bilddaten

Sowohl Daten aus externen Quellen als auch Daten, die während einer Testkampagne durch einen Entwickler gesammelt werden, sind nicht immer in der erwarteten Form und Qualität verfügbar, wie ursprünglich für den Anwendungsfall spezifiziert oder vom Entwickler erwartet. Nachfolgend wird auf die im vorherigen Abschnitt beschriebenen Qualitätsmerkmale und entsprechende Herausforderungen mit Blick auf die genutzten Datensätze eingegangen.

Repräsentativität

Die Repräsentativität des Datensatzes ergibt sich bei der Erhebung von Daten aus externen Quellen aus der Form der erhobenen Daten und ist bei der Erhebung von Echtzeiten nur schwer zu kontrollieren. Im Beispiel des CML-Schiffsbilddatensatzes zeigen die Kameras alle Schiffsklassen, die im Bereich des Hamburger Hafens unterwegs sind, aus einem erhöhten Blickwinkel, abhängig vom Standort der jeweiligen Webcam. Der CML-Datensatz enthält somit nur Schiffe, die den Hamburger Hafen anlaufen und sich im Sichtbereich der untersuchten Kameras befinden. Für Testkampagnen, deren Auslegung vom Hersteller bestimmt werden kann, ist es sinnvoll, die Repräsentativität in Abhängigkeit vom Anwendungsfall und der gewählten Anwendungsdomäne zu definieren. Grundsätzlich gilt für generische Algorithmen, deren Ziel die Erkennung von Schiffen ist, dass der Datensatz von Fraunhofer CML sowie die im Rahmen der Untersuchung herangezogenen Datensätze eine hohe Eignung aufweisen. Eine entsprechende Untersuchung unter Nutzung von Validierungsdaten durch den Entwickler eines KI-Systems sollte jedoch unbedingt durchgeführt werden. Die in Tab 6 aufgeführten Datensätze enthalten Informationen zur Verteilung der entsprechenden Schiffsklassen in den zugehörigen wissenschaftlichen Veröffentlichungen. Eine in Abbildung 14 aufgeführte Darstellung zeigt die individuellen Objektklassen und ihren Anteil am gesamten Datensatz in Prozent. Angesichts der in Kapitel 4.4 identifizierten Herausforderungen bei der Anwendung dieser Datensätze ist ein Abgleich der geplanten Anwendungsdomäne mit der Verteilung der Objektklassen im verwendeten Datensatz unabdingbar.

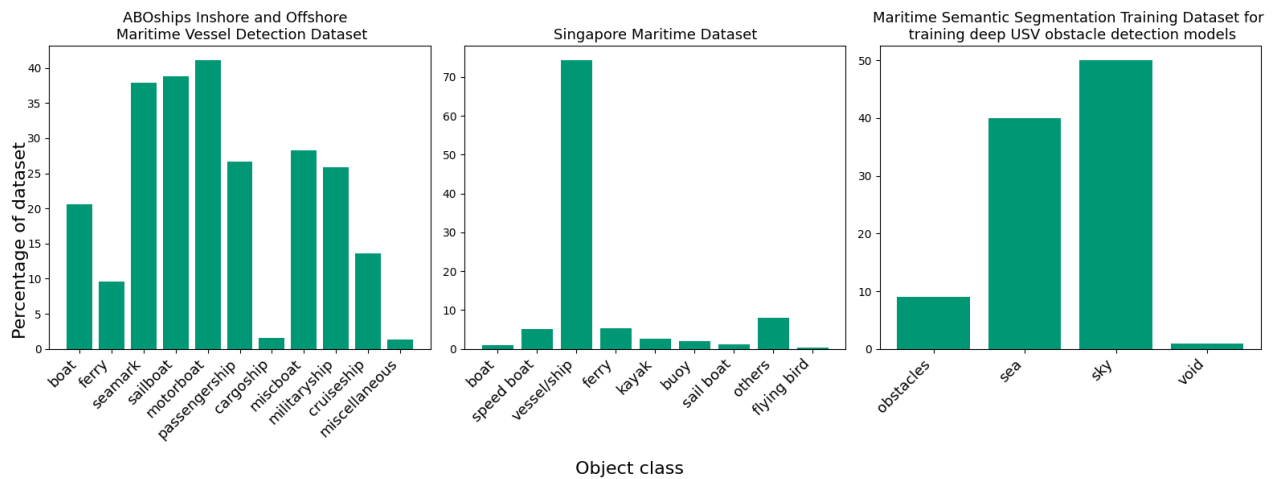


Abbildung 14: Repräsentativität von Datensätzen am Beispiel der externen Datensätze aus Tabelle 6. Aufgetragen nach prozentualen Anteil der jeweiligen Objektklasse am Gesamtanteil des Datensatzes.

Qualität

Bei der Betrachtung der beobachteten Qualität in den verwendeten Datensätzen sind verschiedene Phänomene aufgetreten, die einen entscheidenden Einfluss auf die Verwendbarkeit der Daten haben. Eine genaue Beschreibung der datensatzspezifischen Phänomene findet sich für die externen Datensätze in [50], [61] und [62]. Qualitätseinschränkungen ergeben sich neben den Eigenschaften der Kamera (Sensor, Objektiv und Auflösung) vor allem durch äußere Einflüsse, beispielhaft dargestellt in Abbildung 15. Die beobachteten Effekte können jedoch bei besserer Quantifizierung auch als sinnvolle Ergänzung eines Datensatzes angesehen werden. Wenn die Anwendungsdomäne des entsprechenden KI-Systems diese Störeinflüsse definiert und beschreibt, können Phänomene wie Wetter, Kamerabewegung oder die Reduktion der Auflösung der Bilddaten berücksichtigt und bei entsprechender Datenbasis durch das trainierte KI-Modell robust klassifiziert werden.

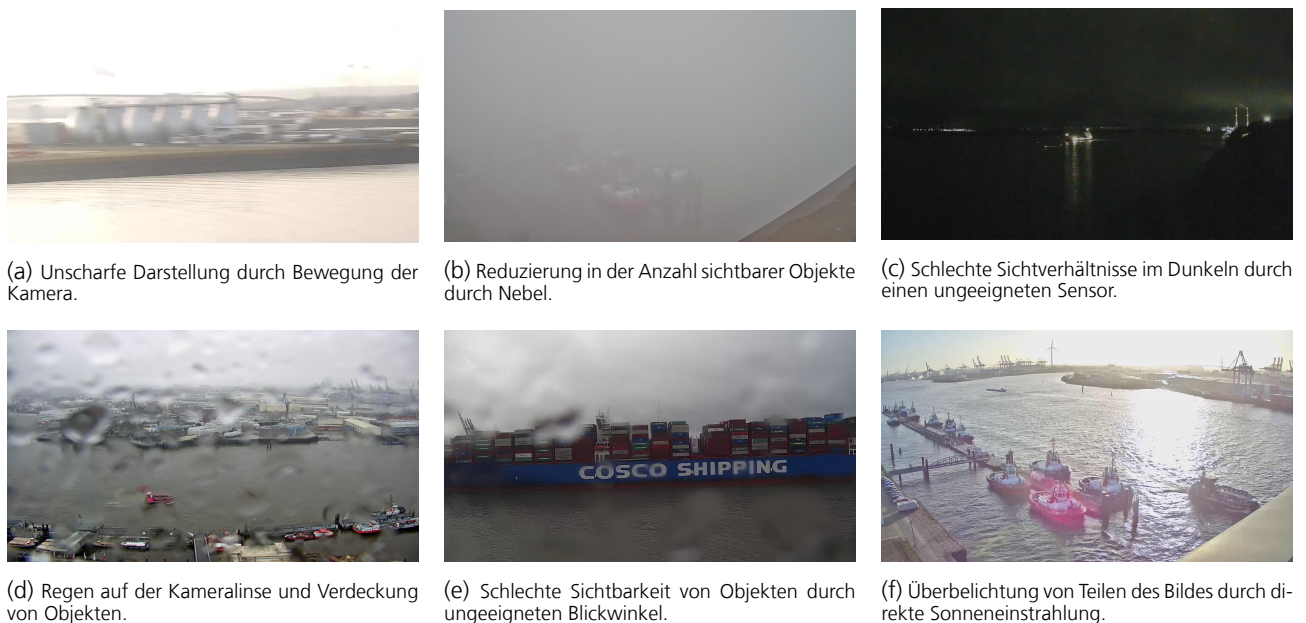


Abbildung 15: Qualitätseinschränkungen in gesammelten Bilddaten am Beispiel des CML Schiffsdatabank Datensatzes auf Basis von Webcams im Bereich des Hamburger Hafen.

Metadaten

Die verwendeten Bilddaten besitzen unterschiedliche Metadaten, je nachdem aus welcher Quelle sie stammen und in welchem Umfang sie nachbearbeitet wurden. Bei allen vorgestellten Datensätzen sind Informationen zur Auflösung, Bildgröße und Dateiformat durch die digitale Bereitstellung inhärent vorgegeben. Informationen zur Hardware des verwendeten Kamerasystems sind nur für das Singapore Maritime Dataset, den ABOships-Datensatz und den MaSTr-Datensatz verfügbar, die vom CML mittels Webcams erhobenen Daten enthalten keine Informationen zur verwendeten Hardware, Aussagen über die Brennweite, das Objektiv oder den Kamerasensor können daher nur geschätzt und nicht näher spezifiziert werden.

Annotation

Bei der Objekterkennung werden üblicherweise Bounding Boxes und zugehörige kategoriale Labels verwendet, um die Position und Identifikation der erkannten Objekte in einer Szene festzuhalten. Im Fall des CML-Datensatzes, der AIS-Daten für Schiffe enthält, gestaltet sich die Annotation als komplex, da es keine eindeutig definierten Objektklassen gibt. Stattdessen werden sämtliche von AIS zur Verfügung gestellten Informationen genutzt. Dies ermöglicht die flexible Erstellung von Annotationen mit einer breiten Palette an Klassen und Labels, um die Vielseitigkeit der Anwendungsfälle bestmöglich zu unterstützen. Dieser Ansatz ermöglicht eine nahtlose Integration komplexer und vielfältiger Schiffsinformationen in den Annotationsprozess.

Concept Drift

Insbesondere bei ML-Modellen, die über einen längeren Zeitraum eingesetzt werden, ist die Veränderlichkeit der umgebenden Objekte ein Kriterium, das kontinuierlich überwacht und beim Betrieb des Systems berücksichtigt werden sollte. Im Gegensatz zu synthetisch erzeugten Daten können reale Daten nur schwer verändert und an neue Anforderungen angepasst werden, die sich aus dieser Veränderlichkeit ergeben. Dementsprechend müssen vorhandene Datensätze kontinuierlich auf ihre Verwendbarkeit für den Anwendungsfall überprüft und ggf. neue Daten beschafft werden, wenn eine Abweichung der im Betrieb beobachteten Daten von den für das Training beobachteten Daten auftritt.

5.3.3 Augmentation von Datensätzen

Die Augmentation von Bilddatensätzen ist ein wichtiges Verfahren im Bereich der Objekterkennung, insbesondere bei im Umfang eingeschränkten Datensätzen. Bei der Datenanreicherung werden vorhandene Bilddaten durch verschiedene Transformationen wie Drehungen, Skalierungen, Spiegelungen, Helligkeitsanpassungen und Rauschanreicherungen künstlich variiert. Dadurch wird die Vielfalt der Trainingsdaten erhöht, was wiederum die Leistungsfähigkeit von Objekterkennungsmodellen verbessert. Ein weiterer Vorteil der Datenanreicherung ist, dass bestehende Annotationen weiter verwendet werden können, da die Änderungen an den Bildern in der Regel die Integrität der Labels nicht beeinträchtigen. Einige der in der Abbildung 116 dargestellten Transformationen zeigen Strategien zur systematischen Veränderung von Bildern und der damit verbundenen Vergrößerung eines Datensatzes. Techniken der synthetischen Datengenerierung finden auch im Bereich der Augmentierung von Datensätzen Anwendung, da ML-basierte Algorithmen auch bestehende Bilder als Input verwenden können und definierte Attribute wie Tageszeit, Wetter oder die Farbe von Objekten verändern können. Eine detaillierte Auflistung, wie die Synthese von Bilddaten den Prozess der Datenbeschaffung unterstützen kann, wird im folgenden Kapitel näher erläutert.

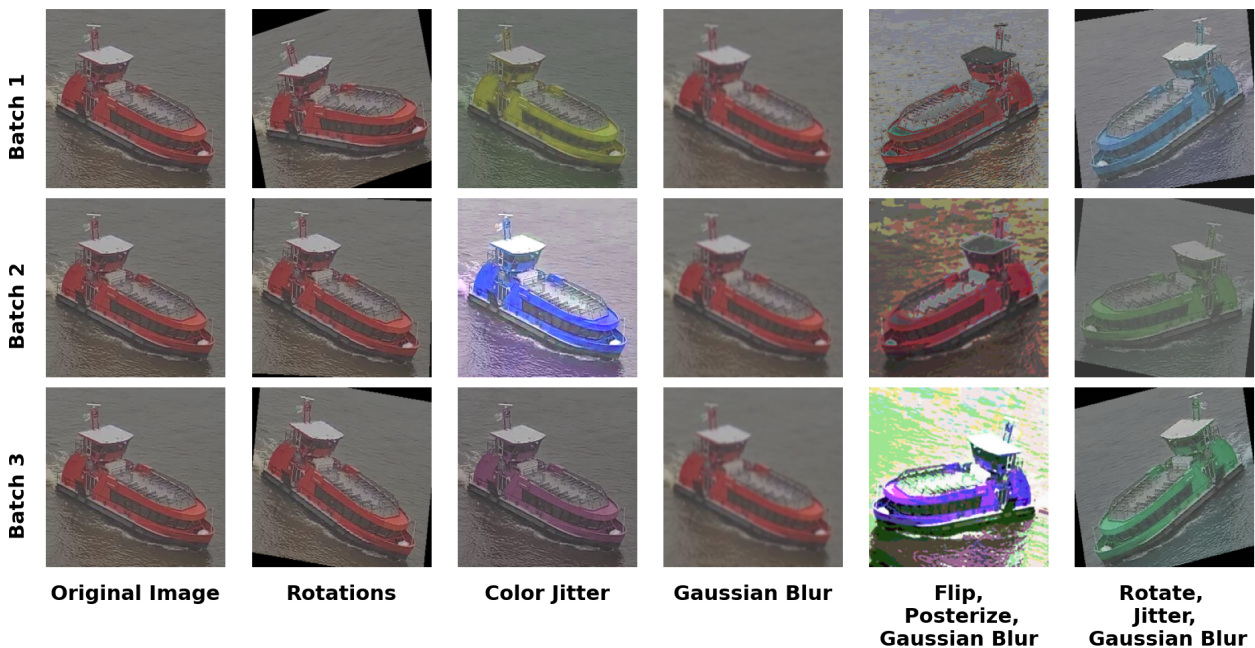


Abbildung 16: Transformationen zur Augmentierung von Bilddaten unter Anwendung verschiedener Strategien (Policies) (Auszug [66])

5.3.4 Annotation von Echtzeiten an Bord eines Schiffes

Die Annotation von Bilddaten spielt eine zentrale Rolle beim Training und bei der Entwicklung von Modellen des überwachten maschinellen Lernens, insbesondere im Bereich der Objekterkennung und der semantischen Segmentierung. In der Regel erfolgt die Annotation durch das Hinzufügen von Bounding Boxes, Masken oder anderen Markierungen, um Objekte in den Bildern zu identifizieren. Die Anwendung von maschinellem Lernen in maritimen Szenarien, insbesondere an Bord von Schiffen, bietet jedoch eine Vielzahl von Datenströmen, die für die Entwicklung von Annotations- und Trainingsprozessen genutzt werden können. Dabei spielen insbesondere IEC 61162-1 (NMEA-0183) und IEC 61162-450 (NMEA-450) eine zentrale Rolle bei der Erfassung und Übertragung von Navigations- und Betriebsdaten. In Verbindung mit ENC-Daten bieten sie die Möglichkeit, Bilddaten mit georeferenzierten Informationen aus der Seekarte zu versehen. Dies ermöglicht eine präzise Referenzierung der visuell erfassten Umgebung, die für Anwendungen in der Navigation und Positionserfassung zur Erstellung umfassender Trainingsdatensätze genutzt werden kann.

Für die Integration von Automatisierungsdaten bieten sich Protokolle wie Modbus, OPC-UA und MQTT an. Sie ermöglichen die Erfassung von Informationen über den Zustand der automatisierten Systeme auf dem Schiff. Die Korrelation dieser Daten mit den Bilddaten erlaubt eine umfassende Analyse und eine genauere Zuordnung der Zustände des Schiffes. Eine Harmonisierung der Schnittstellen ist derzeit jedoch nicht gegeben, so dass eine mögliche Integration dieser Daten im Einzelfall auf ihre Anwendbarkeit und Korrektheit geprüft werden muss.

Zusätzlich ermöglicht der Einsatz von Asterix CAT-240 die Erfassung von Radar-Rohbilddaten. Diese Daten sind besonders wertvoll, da sie als Ground Truth für kamerabasierte Systeme dienen können, um die Leistungsfähigkeit der Objekterkennung in Bezug auf die räumliche Verortung zu bewerten. Beispielhafte Implementierungen von Verfahren zur Objekterkennung in Radarroh Bildern sind in [67] zu finden, beschränken sich jedoch zunächst auf die Erkennung von Objekten in den Radarbildern selbst. Die Detektionen können aber als Hilfsmittel zur Korrelation und räumlichen Zuordnung von Objekten in Kamerabildern genutzt werden. Vertiefende Untersuchungen werden als Forschungsgegenstand für zukünftige Untersuchungen im Bereich der Annotation von Echtzeitdaten mit Hilfe von Betriebsdaten an Bord eines Schiffes angesehen.

5.4 Synthese von Bilddaten

Die in der Entwicklung zur Verfügung stehenden realen Bilddaten Datensätze erfüllen häufig nicht die vorgegebenen Anforderungen (s. Kapitel 5.2). Zwar lassen sich Bilddatensätze unter Verwendung (geometrischer) Transformationen augmentieren und so hinsichtlich verschiedener Eigenschaften verbessern, allerdings nur im beschränkten Maße. Mithilfe der Synthese von Bilddaten, also der Erzeugung von künstlichen Bilddaten, können bestehende Datensätze durch synthetische Bilddaten augmentiert werden oder gänzlich eigenständige synthetische Datensätze

erzeugt werden [68]. Dies bringt den Vorteil, dass Bilddatensätze unabhängig beschafft werden könnten und im Falle von nicht verfügbaren repräsentativen Datensätzen kein kosten- und zeitaufwendiges Datensammeln betrieben werden müsste. Darüber hinaus ist es möglich die synthetischen Daten ihren Anforderungen entsprechend zu erzeugen und initial mit Annotationen verschiedener Arten, z.B. Objekt-Begrenzungsboxen oder segmentierten Bereichen, und mit verschiedenen zugehörigen Informationen, z.B. Schiffstypen oder Wetterverhältnisse, zu versehen.

5.4.1 Aktuelle Verfahren

Synthetische Bilddaten können auf verschiedene Weise erzeugt werden. Gängige Verfahren sind Spielengines oder vortrainierte neuronale Netze, letztere bekannt als *Bildgeneratoren*. [68].

Mit Spiele-Engines, wie beispielsweise der Unreal Engine, lassen sich nach Belieben fiktive Welten erzeugen und steuern. Die Bildsynthese ist dabei präzise durch die das eigens gewählte Szenario in der fiktive Welt vorgegeben. Der Nachteil einer Spiele-Engine liegt im Zeitaufwand zur Erstellung der erforderlichen Szenarien, wodurch die Komplexität der möglichen Szenarien eingeschränkt werden könnte.

Sogenannte *Bildgeneratoren* hingegen synthetisieren Bilder nicht auf Grundlage von selbst konfigurierten, fiktiven Welten sondern i.d.R. durch Texteingaben, sogenannten *Prompts*. Grundlage dieser Bildgeneratoren bilden Netzwerke mit unterschiedlichen Architekturen. Bekannte Beispiele sind Generative Adversarial Networks (GAN), Diffusionsmodelle oder Variational Autoencoder (VAE) [68, 69]. Prominente Beispiele für Bildgeneratoren sind Stable Diffusion XL, Midjourney und DALL-E 2 [70]. Das Programm DALL-E 2 ermöglicht es generierte Bilder durch weitere Prompts nachträglich anzupassen oder ähnliche Bilder zu generieren. Die Entwicklungen in diesem Feld schreiten zügig voran, so dass beispielsweise DALL-E 2 im Zeitraum der Fertigstellung der Studie durch den Nachfolger DALL-E 3 ersetzt werden wird [71]. Dieser baut nativ auf ChatGPT auf und soll nachträgliche Anpassungen von generierten Bildern weiter verbessern.

Aktuelle Forschungsentwicklungen gehen in die Richtung, dass Bilder nicht nur (einmalig) basierend auf Grundlage von Prompts erzeugt werden können, sondern andere Modalitäten herangezogen werden können [69]. Hierbei wird die Möglichkeit von visuellen Eingaben, z.B. semantischen Karten, Szenario-Aufbauten oder Skizzen, erforscht. Gerade semantische Karten oder Szenario-Aufbauten ließen sich mit der Anwendungsdomäne aus dem LZM in Verbindung bringen [28, 72].

Bereits mit den aktuellen Versionen der Bildgeneratoren sind die Ergebnisse bei der Verwendung von verschiedenen szenariobeschreibenden Ausdrücken wie *bewölkt* oder *wolkenlos* deutlich erkennbar. Diese sind in den Abbildungen 17 und 18 unter Verwendung von DALL-E 2 und in den Abbildungen 19 und 20 unter Verwendung von Stable Diffusion dargestellt.



Abbildung 17: Synthetische Bilder erzeugt mit DALL-E 2 bei Verwendung der Texteingabe *multiple container vessels in coastal waters in cloudless weather during daytime*. Das erste Bild wurde zuerst erzeugt und die nachfolgenden drei nachträglich als Variationen des ersten.



Abbildung 18: Synthetische Bilder erzeugt mit DALL-E 2 bei Verwendung der Texteingabe *multiple container vessels in coastal waters in cloudy weather during daytime*. Das erste Bild wurde zuerst erzeugt und die nachfolgenden drei nachträglich als Variationen des ersten.

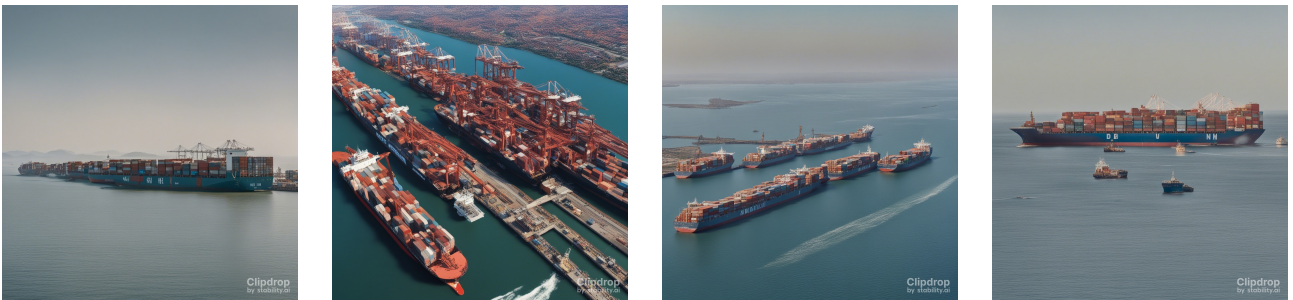


Abbildung 19: Synthetische Bilder erzeugt mit Stable Diffusion XL bei Verwendung der Texteingabe *multiple container vessels in coastal waters in cloudless weather during daytime*.

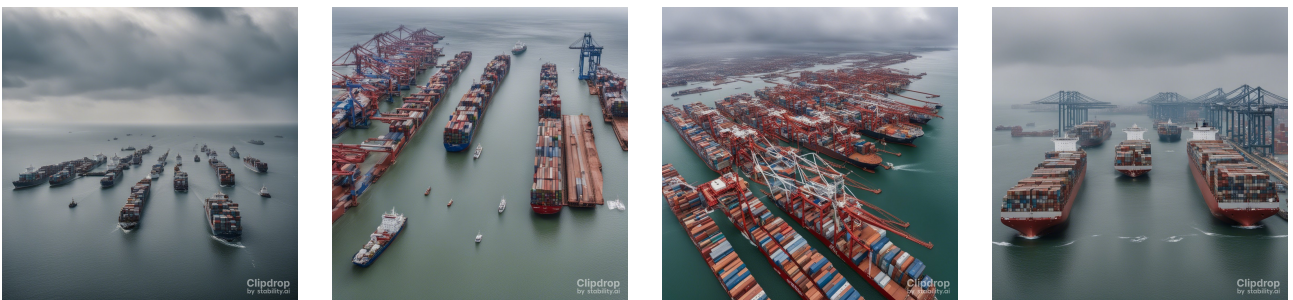


Abbildung 20: Synthetische Bilder erzeugt mit Stable Diffusion XL bei Verwendung der Texteingabe *multiple container vessels in coastal waters in cloudy weather during daytime*.

Synthetisch gewonnene Bilddaten können entsprechend initial mit notwendigen Annotationen ausgestattet werden. Im obigen Fall (s. Abbildungen [17](#) und [18](#)) würde sich die Annotation insbesondere hinsichtlich der Wetterbedingungen (*wolkenlos* bzw. *bewölkt*) unterscheiden. Neben diesen Chancen liegen Herausforderungen in der präzisen Steuerung der Bildgeneratoren durch geeignete Prompts und Konfigurationsparameter. So ist an den beispielhaft generierten Bildern zu erkennen, dass trotz der gleichen Prompts, DALL-E 2 und Stable Diffusion XL deutlich unterschiedliche Ergebnisse liefern. Ersteres generiert Bilder aus einer Perspektive auf Schiffshöhe und in Richtung offenes Wasser, wohingegen letzteres vornehmlich Bilder aus der Vogelperspektive auf offenem Wasser und auch in unmittelbarer Hafennähe generiert. Eine Präzisierung des Prompts für DALL-E 2 zu *aerial view of multiple container vessels in coastal waters in cloudless weather during daytime* liefert hingegen vergleichbare Ergebnisse (s. Abbildung [21](#)).



Abbildung 21: Synthetische Bilder erzeugt mit DALL-E 2 bei Verwendung der Texteingabe *aerial view of multiple container vessels in coastal waters in cloudless weather during daytime*.

Erste Forschungsergebnisse zeigen, dass NN, die ausschließlich mit synthetischen Bilddaten trainiert worden sind, nicht ganz so gute, aber vergleichbare Ergebnisse liefern, wie NN, die ausschließlich mit echten Bilddaten trainiert worden sind [68]. Es wird vermutet, dass eine Erhöhung des Realismus der synthetischen Bilddaten diese Lücke verkleinern könnte. Ebenfalls sei anzumerken, welchen Nutzen bereits verfügbare synthetische Bilddaten gegenüber der kosten- und zeitaufwendigen Datensammlung mit sich bringen.

6 Zusammenfassung

In diesem Kapitel werden die wichtigsten Ergebnisse dieser Studie zusammengefasst. Diese sind wie die Studie selbst in drei wesentliche Abschnitte eingeteilt: Leitfaden zur Entwicklung sicherheitskritischer Deep-Learning-Anwendungen, der Anwendung von Objekterkennungsalgorithmen im nautischen Kontext sowie den Anforderungen an Bilddaten und ihre Beschaffung.

Leitfaden zur Entwicklung sicherheitskritischer Deep-Learning-Anwendungen

Bisher fehlt ein spezifischer Leitfaden zur Entwicklung sicherheitskritischer Deep-Learning-Anwendungen im nautischen Kontext. Stattdessen existieren lose zusammenhängende Normen, Richtlinien und Standards, die unterschiedliche Teilbereiche von spezifischen bis sehr allgemein abbilden (s. Kapitel [3.1](#)). In dieser Studie wurde ein erster Leitfaden auf Grundlage dieser Dokumente erarbeitet (s. Kapitel [3.2](#)), der diese Dokumente kohärent und umfassend zusammenführt. Um ein breites Anwendungsfeld zu bedienen, ist dieser Leitfaden generisch und im Sinne von Maschinellem Lernen (ML) unabhängig vom algorithmischen Ansatz gehalten. Der Leitfaden basiert im Wesentlichen auf dem Lebenszyklusmodell (LZM) und berücksichtigt in jeder LZM-Phase Schritte, die es zur Entwicklung einer sicheren und zuverlässigen Anwendung bedarf. Um den fachlichen Kontext zu schaffen, wurden Methoden und Verfahren aus der maritimen Welt übernommen. Als zentrales Beispiel sei die Formalisierung der Anwendungsdomäne nach dem Ansatz des Operational Envelope (s. Kapitel [3.2.1](#)) erwähnt, um auch die fortschreitende Entwicklung von Maritime Autonomous Surface Ships (MASS) zu berücksichtigen. Der Leitfaden dient damit nicht nur unterstützend bei der Entwicklung solcher Anwendungen, sondern auch als Motivation für die Standardisierung ausgewählter, zum Beispiel besonders sicherheitskritischer, Schritte.

Anwendung von Objekterkennungsalgorithmen im nautischen Kontext

Die Evaluierung der Algorithmen (hier: Modelle) zur Detektion und semantischer Sementierung von Objekten hat gezeigt, dass diese Modelle (mit Nachtraining und Anpassungen) eine akzeptable Leistung für maritime Anwendungen liefern (s. Kapitel [4](#)) erbringen. Es zeigt sich jedoch, dass aufgrund der Herausforderungen an das Bildmaterial im nautischen Kontext durch Wetter, Bewegung und unterschiedlicher Lichtverhältnisse ein Training auf domänenspezifischen Daten notwendig ist. Weiterhin erschwert die geringe Größe maritimer Objekte selbst auf hochaufgelösten Bild- oder Videoaufnahmen ihre (frühzeitige) Erkennung. Die Lösung dieser Herausforderungen wird als separater Forschungsgegenstand betrachtet.

Anforderungen an Bilddaten und ihre Beschaffung

Um die Entwicklung von sicheren und zuverlässigen Anwendungen zu ermöglichen wurden Qualitätsmerkmale als Anforderungen an Bilddaten identifiziert (s. Kapitel [5.3.2](#)). Anhand dieser lässt sich beurteilen, ob Bilddaten(-sätze) zum Training von DL-basierten Modellen hinreichend geeignet sind. Dies ist notwendig um die spezifischen Anforderungen nautischer Anwendungen gerecht zu werden. Hier seien spezielle Objekte wie Navigationshilfen, Sichtzeichen oder Leuchttürme, aber auch unterschiedlichste, einflussnehmende Umweltbedingungen erwähnt. Zur Erhöhung der Sicherheit und Zuverlässigkeit ist eine Fusion mit weiteren sensorischen Quellen wie Radar nahelegend.

Zur Verbesserung der Qualitätsmerkmale von Bilddaten(-sätzen) lassen sich Augmentierung oder Synthetisierung verwenden. Während die Augmentierung von Bilddatensätzen bereits ein gängiges Verfahren darstellt (s. Kapitel [5.3.3](#)), bedarf die sichere und zuverlässige Synthetisierung von Bilddaten weitere Forschungsarbeiten. Erste Ergebnisse unter der Verwendung der sich dauerhaft verbessernden Bildgeneratoren wirken vielversprechend (s. Kapitel [5.4](#)). Es wird vermutet, dass die zielgerichtete Synthese von Bilddatensätzen eine zentrale Rolle bei der Entwicklung (und Prüfung) von leistungsfähigen Objekterkennungsmodellen spielen könnte. Die Synthese könnte unter Verwendung der Semantik aus der Formalisierung der Anwendungsdomäne erfolgen.

7 Literatur

- [1] *Avikus HiNAS 2.0*. <https://avikus.ai/eng/product/hinas>, Juni 2023.
- [2] *Avikus NeuBoat*. <https://avikus.ai/eng/product/neuboot>, Juli 2023.
- [3] *Awarion*. <https://cra.com/awarion-autonomous-lookout-system/>, Juli 2023.
- [4] *Captain AI*. <https://www.captainai.com/technology/>, Juli 2023.
- [5] *Groke Pro*. <https://www.groke-tech.com/en/products>, Juli 2023.
- [6] *Infoceanus Command*. <http://infoceanus.com/en/>, Juli 2023.
- [7] *Kongsberg Maritime Autonomous Shipping*. <https://www.kongsberg.com/maritime/support/themes/autonomous-shipping/>, Mai 2023.
- [8] *Kongsberg Maritime Intelligent Awareness*. <https://www.kongsberg.com/maritime/about-us/news-and-media/our-stories/intelligent-awareness/>, Mai 2023.
- [9] *Marine AI Guardian Autonomy*. <https://marineai.co.uk/>, März 2020.
- [10] *Mayflower Autonomous Ship*. <https://mas400.com/technology>, Juli 2023.
- [11] *MEGURI2040 Fully Autonomous Ship Program*. <https://www.nippon-foundation.or.jp/en/what/projects/meguri2040>, Juli 2023.
- [12] *Mtu NautIQ CoPilot*. <https://www.mtu-solutions.com/eu/de/automation-solutions/ship-automation-systems/mtu-nautiq-copilot.html>, Dezember 2020.
- [13] *Orca AI*. <https://www.orca-ai.io/>, Juli 2023.
- [14] Thornell, John: *Rite Solutions*. <http://www.rite-solutions.com/innovation/ship-recognition-system/>, Juli 2023.
- [15] *Roboat*. <https://roboat.org/technology>, 3.
- [16] *Robopec*. <https://www.robopec.com/en/products/maritime-object-detection-and-tracking-mdt/>, Juli 2023.
- [17] *Saildrone*. <https://www.saildrone.com/solutions/maritime-domain-awareness>, Juli 2023.
- [18] *Sea Machines*. <https://sea-machines.com/>, Mai 2018.
- [19] *SEA.AI*. <https://sea.ai/how-it-works/>, Juli 2023.
- [20] *Seadronix*. <https://www.seadronix.com/?lang=en>, Juli 2023.
- [21] *Seafar*. <https://seafar.eu/services/>, Juli 2023.
- [22] *SeaSight - Situational Awareness and Collision Avoidance*. <https://www.maritimerobotics.com/post/seasight>, November 2022.
- [23] *SEDOR AI Ship Detection*. <https://d-components.com/aufzeichnung/zubehoer-digitalrecorder/lizenzen/2928/dlc-sedor-ai-ship-detection-p-s/w-8>, Juli 2023.
- [24] *Svessel Samsung Smart Ship*. https://shi.svessel.com/?page_id=298, Juli 2023.
- [25] *Wärtsilä Voyage Autonomy Solutions*. <https://www.wartsila.com/insights/whitepaper/the-future-of-smart-autonomy-is-here>, Juli 2023.
- [26] *Yara Birkeland*. <https://www.kongsberg.com/maritime/about-us/news-and-media/our-stories/autonomy-is-here-powered-by-kongsberg/>, Juli 2023.
- [27] Thombre, Sarang, Zhao, Zheng, Ramm-Schmidt, Henrik, Vallet García, José M., Malkamäki, Tuomo, Nikolskiy, Sergey, Hammarberg, Toni, Nuortie, Hiski, H. Bhuiyan, M. Zahidul, Särkkä, Simo und Lehtola, Ville V.: *Sensors and AI Techniques for Situational Awareness in Autonomous Ships: A Review*. IEEE Transactions on Intelligent Transportation Systems, 23(1):64–83, Januar 2022, ISSN 1558-0016.

- [28] Koch, Paul, Stach, Thomas, Constapel, Manfred und Burmeister, Hans Christoph: *VerifAI, Studie Zur Zielbasierten Standardisierung in Der Prüfung Und Zulassung Intelligenter Entscheidungseinrichtungen von Teilautonomen Überwasserfahrzeugen*. Studie, 2023.
- [29] *DIN SPEC 92001-1:2019-04, Künstliche Intelligenz - Life Cycle Prozesse Und Qualitätsanforderungen - Teil 1: Qualitäts-Meta-Modell*, April 2019.
- [30] *DIN SPEC 92001-2:2020-12, Künstliche Intelligenz - Life Cycle Prozesse Und Qualitätsanforderungen - Teil 2: Robustheit*, Dezember 2020.
- [31] Poretschkin, Maximilian, Schmitz, Anna, Akila, Maram, Adilova, Linara, Becker, Daniel, Cremers, Armin, Hecker, Armin, Houben, Sebastian, Mock, Michael, Rosenzweig, Julia, Sicking, Joachim, Schulz, Elena, Voss, Angelika und Wrobel, Stefan: *Leitfaden Zur Gestaltung Vertrauenswürdiger Künstlicher Intelligenz*. Technischer Bericht, Fraunhofer-Institut für Intelligente Analyse und Informationssysteme IAIS, Juli 2021.
- [32] *DIN SPEC 13266:2020-04, Leitfaden Für Die Entwicklung von Deep-Learning-Bilderkennungssystemen*, April 2020.
- [33] *ISO/IEC Technical Report 29119-11, Software and Systems Engineering - Software Testing - Part 11: Guidelines on the Testing of AI-based Systems*, November 2020.
- [34] *ISO/IEC International Standard 22989, Information Technology - Artificial Intelligence - Artificial Intelligence Concepts and Terminology*, Juli 2022.
- [35] *ISO/IEC International Standard 23053:2022-06, Framework for Artificial Intelligence (AI) Systems Using Machine Learning (ML)*, Juni 2022.
- [36] *ISO/IEC Technical Specification 4213, Information Technology - Artificial Intelligence - Assessment of Machine Learning Classification Performance*, Oktober 2022.
- [37] *ISO/IEC Technical Report 24030 Information Technology - Artificial Intelligence (AI) - Use Cases*, Mai 2021.
- [38] *ISO/IEC/IEEE 12207:2017. Systems and Software Engineering - Software Life Cycle Processes*, November 2017.
- [39] Ashmore, Rob, Calinescu, Radu und Paterson, Colin: *Assuring the Machine Learning Lifecycle: Desiderata, Methods, and Challenges*. *ACM Computing Surveys*, 54(5):111:1–111:39, Mai 2021, ISSN 0360-0300.
- [40] *ISO Technical Specification 23860, Ships and Marine Technology - Vocabulary Related to Autonomous Ship Systems*, Mai 2022.
- [41] Feng, Xin, Jiang, Youni, Yang, Xuejiao, Du, Ming und Li, Xin: *Computer Vision Algorithms and Hardware Implementations: A Survey*. *Integration*, 69:309–320, November 2019, ISSN 0167-9260.
- [42] *ISO/IEC Technical Report 24027 Information Technology - Artificial Intelligence (AI) - Bias in AI Systems and AI Aided Decision Making*, November 2021.
- [43] Zhao, Yizhen: *MLOps and Data Versioning in Machine Learning Project*. Technischer Bericht, Universität Amsterdam, Amsterdam, 2020.
- [44] Klump, Jens, Wyborn, Lesley, Wu, Mingfang, Martin, Julia, Downs, Robert R. und Asmi, Ari: *Versioning Data Is About More than Revisions: A Conceptual Framework and Proposed Principles*. *Data Science Journal*, 20:12, März 2021, ISSN 1683-1470.
- [45] Chen, Xiang, Liu, Yuanchang und Achuthan, Kamalasudhan: *WODIS: Water Obstacle Detection Network Based on Image Segmentation for Autonomous Surface Vehicles in Maritime Environments*. *IEEE Transactions on Instrumentation and Measurement*, 70:1–13, 2021.
- [46] Ren, Shaoqing, He, Kaiming, Girshick, Ross B. und Sun, Jian: *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015. <https://api.semanticscholar.org/CorpusID:10328909>.
- [47] Lin, Tsung Yi, Goyal, Priya, Girshick, Ross B., He, Kaiming und Dollár, Piotr: *Focal Loss for Dense Object Detection*. 2017 IEEE International Conference on Computer Vision (ICCV), Seiten 2999–3007, 2017. <https://api.semanticscholar.org/CorpusID:47252984>.

- [48] Carion, Nicolas, Massa, Francisco, Synnaeve, Gabriel, Usunier, Nicolas, Kirillov, Alexander und Zagoruyko, Sergey: *End-to-End Object Detection with Transformers*. European conference on computer vision (pp. 213-229). Springer International Publishing., abs/2005.12872, 2020. <https://api.semanticscholar.org/CorpusID:218889832>.
- [49] Yang, Jianwei, Li, Chunyuan, Zhang, Pengchuan, Dai, Xiyang, Xiao, Bin, Yuan, Lu und Gao, Jianfeng: *Focal Attention for Long-Range Interactions in Vision Transformers*. In: *Neural Information Processing Systems*, 2021. <https://api.semanticscholar.org/CorpusID:245011146>.
- [50] Wang, Chien Yao, Bochkovskiy, Alexey und Liao, Hong Yuan Mark: *YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors*. 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seiten 7464–7475, 2022. <https://api.semanticscholar.org/CorpusID:250311206>.
- [51] Lin, Tsung Yi, Maire, Michael, Belongie, Serge J., Hays, James, Perona, Pietro, Ramanan, Deva, Dollár, Piotr und Zitnick, C. Lawrence: *Microsoft COCO: Common Objects in Context*. In: *European Conference on Computer Vision*, 2014. <https://api.semanticscholar.org/CorpusID:14113767>.
- [52] Chen, Liang Chieh, Papandreou, George, Schroff, Florian und Adam, Hartwig: *Rethinking Atrous Convolution for Semantic Image Segmentation*. ArXiv, abs/1706.05587, 2017. <https://api.semanticscholar.org/CorpusID:22655199>.
- [53] Howard, Andrew G., Sandler, Mark, Chu, Grace, Chen, Liang Chieh, Chen, Bo, Tan, Mingxing, Wang, Weijun, Zhu, Yukun, Pang, Ruoming, Vasudevan, Vijay, Le, Quoc V. und Adam, Hartwig: *Searching for MobileNetV3*. 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seiten 1314–1324, 2019. <https://api.semanticscholar.org/CorpusID:146808333>.
- [54] Yuan, Yuhui, Chen, Xilin und Wang, Jingdong: *Object-Contextual Representations for Semantic Segmentation*. In: *European Conference on Computer Vision*, 2019. <https://api.semanticscholar.org/CorpusID:202734362>.
- [55] Xie, Enze, Wang, Wenhai, Yu, Zhiding, Anandkumar, Anima, Alvarez, Jose M und Luo, Ping: *SegFormer: Simple and efficient design for semantic segmentation with transformers*. *Advances in Neural Information Processing Systems*, 34:12077–12090, 2021.
- [56] Chen, Zhe, Duan, Yuchen, Wang, Wenhai, He, Junjun, Lu, Tong, Dai, Jifeng und Qiao, Y.: *Vision Transformer Adapter for Dense Predictions*. The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023, abs/2205.08534, 2023. <https://api.semanticscholar.org/CorpusID:248834106>.
- [57] Zhou, Bolei, Zhao, Hang, Puig, Xavier, Fidler, Sanja, Barriuso, Adela und Torralba, Antonio: *Scene Parsing through ADE20K Dataset*. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Seiten 5122–5130, 2017.
- [58] Cordts, Marius, Omran, Mohamed, Ramos, Sebastian, Rehfeld, Timo, Enzweiler, Markus, Benenson, Rodrigo, Franke, Uwe, Roth, Stefan und Schiele, Bernt: *The Cityscapes Dataset for Semantic Urban Scene Understanding*. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [59] Russakovsky, Olga, Deng, Jia, Su, Hao, Krause, Jonathan, Satheesh, Sanjeev, Ma, Sean, Huang, Zhiheng, Karpathy, Andrej, Khosla, Aditya, Bernstein, Michael S., Berg, Alexander C. und Fei-Fei, Li: *ImageNet Large Scale Visual Recognition Challenge*. *International Journal of Computer Vision*, 115:211 – 252, 2014. <https://api.semanticscholar.org/CorpusID:2930547>.
- [60] Prasad, Dilip K., Rajan, Deepu, Rachmawati, Lily, Rajabally, Eshan und Quek, Chai: *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, 2016. <https://api.semanticscholar.org/CorpusID:11905372>.
- [61] Iancu, Bogdan, Soloviev, Valentin, Zelioli, Luca und Lilius, Johan: *ABOShips - An Inshore and Offshore Maritime Vessel Detection Dataset with Precise Annotations*. ArXiv, abs/2102.05869, 2021. <https://api.semanticscholar.org/CorpusID:231879722>.

- [62] Bovcon, Borja, Muhovič, Jon, Perš, Janez und Kristan, Matej: *The MaStr1325 dataset for training deep USV obstacle detection models*. In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019.
- [63] Ljungqvist, Martin Georg, Nordander, Otto, Skans, Markus, Mildner, Arvid, Liu, Tony und Nugues, Pierre: *Object Detector Differences When Using Synthetic and Real Training Data*. *SN Computer Science*, 4(3):302, März 2023, ISSN 2661-8907.
- [64] Kasaei, S. Hamidreza, Ghorbani, Maryam, Schilperoort, Jits und Rest, Wessel van der: *Investigating the Importance of Shape Features, Color Constancy, Color Spaces and Similarity Measures in Open-Ended 3D Object Recognition*. *CoRR*, abs/2002.03779, 2020. <https://arxiv.org/abs/2002.03779>.
- [65] Cromey, Douglas: *Digital Images Are Data: And Should Be Treated as Such*. *Methods in molecular biology* (Clifton, N.J.), 931:1–27, Januar 2013.
- [66] Cubuk, Ekin D., Zoph, Barret, Mane, Dandelion, Vasudevan, Vijay und Le, Quoc V.: *AutoAugment: Learning Augmentation Policies from Data*, 2019.
- [67] Siegert, Gregor, Hoth, Julian, Banyś, Paweł und Heymann, Frank: *Generic Framework for Vessel Detection and Tracking Based on Distributed Marine Radar Image Data*. *CEAS Space Journal*, 11(1):65–79, März 2019, ISSN 1868-2510.
- [68] Man, Keith und Chahl, Javaan: *A Review of Synthetic Image Data and Its Use in Computer Vision*. *Journal of Imaging*, 8(11):310, November 2022, ISSN 2313-433X.
- [69] Zhan, Fangneng, Yu, Yingchen, Wu, Rongliang, Zhang, Jiahui, Lu, Shijian, Liu, Lingjie, Kortylewski, Adam, Theobalt, Christian und Xing, Eric: *Multimodal Image Synthesis and Editing: A Survey and Taxonomy*. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Seiten 1–20, 2023, ISSN 1939-3539.
- [70] Borji, Ali: *Generated Faces in the Wild: Quantitative Comparison of Stable Diffusion, Midjourney and DALL-E 2*, Juni 2023.
- [71] Mewes, Bernd: *Dall-E 3: OpenAI verbessert Ergebnisse dank nativer Chat-GPT-Unterstützung*. <https://www.heise.de/news/Dall-E-3-OpenAI-verbessert-Ergebnisse-dank-nativer-Chat-GPT-Unterstuetzung-9312381.html>, September 2023.
- [72] Burmeister, H. C., Constapel, M., Ugé, C und Jahn, C: *From Sensors to MASS: Digital Representation of the Perceived Environment Enabling Ship Navigation*. Band 929 der Reihe *IOP Conference Series: Materials Science and Engineering*. IOP Publishing, 2020.

8 Würdigung und abschließende Anmerkung

Die Inhalte der vorliegenden Studie repräsentieren den aktuellen Wissenstand der Autoren des Fraunhofer CML und keine subjektive Einschätzung des BSH.

Die Recherchen, Erkenntnisse und Betrachtungen basieren auf dem Stand vom Dezember 2023.