Seabed Fingerprinting for Maritime Navigation in GNSS-Denied Environments

SAND-E: Seabed-Aided Navigation Using Classical and Learned Image Matching

J. Pille

TU Delft

CGI

Seabed Fingerprinting for Maritime Navigation in GNSS-Denied Environments

SAND-E: Seabed-Aided Navigation Using Classical and Learned Image Matching

by

J. Pille

to obtain the degree of Master of Science
in Geomatics at Delft University of Technology.
Defended on Thursday June 11, 2026 at 10:00.

The work in this thesis was carried out in the:

3D Geoinformation Group
Delft University of Technology

in collaboration with:

CGI

Royal Netherlands Navy

Thesis Committee:	Dr. L. Nan	TU Delft
	Dr. R.C. Lindenbergh	TU Delft
	Ir. R.L. Voûte	CGI
Co-reader:	Dr. F. Dahle	TU Delft

Preface

As I complete my six years at TU Delft, I look back with great appreciation on the people who have supported me throughout this journey. These years have been an important period of learning, growth, and development, both academically and personally. I am grateful to everyone who contributed to this journey in their own way.

First, I would like to thank my supervisors, Liangliang Nan, Roderik Lindenbergh, and Robert Voûte, for their guidance throughout this thesis. Our biweekly meetings, their thoughtful feedback, and their willingness to answer my questions quickly helped me greatly, even when those questions arrived late on a Friday afternoon.

I am also very grateful to CGI for warmly welcoming me and supporting me throughout the entire thesis period. The many conversations, harbor visits, and practical suggestions I received were extremely valuable. A special word of thanks goes to Wouter Dobbenberg for his enthusiasm, and close involvement, and for teaching me to write code more efficiently, a skill I wish I had developed in year one rather than year six.

My sincere appreciation also goes to the Royal Netherlands Navy, and in particular to the Hydrographic Service. I would especially like to thank Thijs Ligteringen and Michael Knoop for their generous support and the considerable effort they put into providing the data needed for this research. I greatly appreciated the many visits to the Frederikkazerne and the time they spent helping me. Their contribution, from sharing high-quality maps of the North Sea to giving me the opportunity to visit a ship and see the vessels in the harbor, was a great experience and made this project even more valuable and meaningful.

On a personal level, I am deeply grateful to my family for their constant support throughout my studies and thesis. I especially want to thank my parents and my boyfriend for encouraging me, listening to me, and making the effort to understand my work, which was sometimes a task harder than the thesis itself.

I would also like to thank my fellow Geomatics students. Working alongside them made the thesis enjoyable and helped me stay motivated. Even the informal rule that being late meant paying for good-quality coffee from the Espressobar turned out to be a useful incentive.

Finally, I want to thank my roommates and friends for patiently listening to my stories, presentations, and doubts during this period, even when the topic was far removed from their own interests. You now know more about seabed fingerprinting than you ever wanted to.

Julia Pille
Delft, June 2026

Abstract

Maritime navigation relies heavily on Global Navigation Satellite Systems (GNSS), yet military surface vessels must remain operational when satellite signals are unavailable, degraded, or denied. In such GNSS-denied environments, Inertial Navigation Systems (INS) accumulate unbounded drift, while existing Terrain-Aided Navigation (TAN) methods remain sensitive to terrain distinctiveness and are rarely evaluated for surface vessels. We present SAND-E, a particle-filter framework for seabed-aided maritime navigation that treats seabed fingerprinting as an image matching problem. Near real-time Multibeam Echosounder (MBES) measurements are matched against bathymetric reference maps using Normalized Cross-Correlation (NCC), SuperPoint+LightGlue (SP+LG), or a combined prior-gated method, and the resulting position fixes are fused into the particle filter for recursive state estimation. Evaluated on North Sea and Atlantic Ocean bathymetry, NCC outperforms SP+LG across all metrics, achieving an RMSE of 92.1 m, a 100% fix rate, 92.6% of runs within 500 m, and a runtime of 0.5 ms per fix. The combined method matches NCC under nominal conditions but provides additional robustness with an outdated reference map, where the prior gate rejects degraded NCC fixes and falls back to SP+LG. The framework generalizes across three geographically distinct test areas, remains viable with a three-year-old reference map, and reduces average final position error from Dead Reckoning (DR) to 115.3 m, demonstrating seabed fingerprinting as a viable infrastructure-independent navigation solution for GNSS-denied military surface vessels.

Keywords: GNSS-Denied Maritime Navigation, Seabed-Aided Navigation, Bathymetric Image Matching, Particle Filter, Multibeam Echosounder.

Nomenclature

Acronyms

Acronym	Definition
GNSS	Global Navigation Satellite Systems
INS	Inertial Navigation System
TAN	Terrain-Aided Navigation
SAND-E	Seabed-Aided Navigation in GNSS-Denied Environments
MBES	Multibeam Echosounder
NCC	Normalized Cross-Correlation
SP+LG	SuperPoint+LightGlue
PNT	Position, Navigation, and Timing
DR	Dead Reckoning
LORAN	Long Range Navigation
SONAR	Sound Navigation and Ranging
SIFT	Scale-Invariant Feature Transform
SURF	Speeded Up Robust Features
ORB	Oriented FAST and Rotated BRIEF
CNN	Convolutional Neural Network
PF	Particle Filter
KF	Kalman Filter
SBES	Single-Beam Echosounder

Symbols

Symbol	Definition	Unit	Domain
Problem Description			
R	Bathymetric reference map	-	$\Omega_R \subset \mathbb{R}^2 \to \mathbb{R}$
Ω_R	Spatial domain of the reference map R	$[\mathrm{m}]$	$\subset \mathbb{R}^2$
M_k	Onboard MBES measurement at time step k	-	$\Omega_{M_k} \subset \mathbb{R}^2 \to \mathbb{R}$
Ω_{M_k}	Spatial domain of the measurement M_k	$[\mathrm{m}]$	$\subset \mathbb{R}^2$
$\boldsymbol{z}_k$	Position fix at time step k	$[\mathrm{m}]$	$\mathbb{R}^2$
$\boldsymbol{x}_k$	Full vessel state vector $\left(p_x, p_y, \psi, v\right)^\mathsf{T}$ at time step k	[mixed]	$\mathbb{R}^4$
p_x, p_y	Horizontal position components	$[\mathrm{m}]$	$\mathbb{R}$
ψ	Vessel heading (from North)	$[^\circ]$	$\mathbb{R}$
v	Constant speed over ground	$[\mathrm{kn}]$	$\mathbb{R}_{>0}$
$\widetilde{M}_k$	North-aligned (rotated) measurement at time step k	—	$\mathbb{R}^2 \to \mathbb{R}$
S_k	Local search region extracted from $\hat{R}$ at time step k	—	$\subset \hat{R}$

Symbol	Definition	Unit	Domain
Data Preparation			
d_{nodata}	NoData cell indicator value in the raster	—	$\mathbb{R}$
i, j	Row and column index of a raster cell	—	$\mathbb{Z}_{\geq 0}$
$\mathcal{M}_{\text{valid}}(i, j)$	Binary mask: 1 if cell (i, j) has a valid depth value, 0 otherwise	—	$\{0, 1\}$
$\mathcal{M}_{\text{border}}(i, j)$	Binary mask: 1 if NoData cell (i, j) is border-connected	—	$\{0, 1\}$
$\mathcal{M}_{\text{interior}}(i, j)$	Binary mask: 1 if NoData cell (i, j) is interior (not border-connected)	—	$\{0, 1\}$
(i^*, j^*)	Nearest valid neighbour of an interior NoData cell (by Euclidean distance)	[px]	$\mathbb{Z}^2$
$\hat{R}$	Preprocessed reference map with interior NoData cells filled	—	$\mathbb{R}^2 \to \mathbb{R}$
Particle Filter Representation			
N	Number of particles	—	$\mathbb{Z}_{>0}$
$\boldsymbol{x}_k^{(i)}$	State of particle i at time step k	[mixed]	$\mathbb{R}^4$
$w_k^{(i)}$	Normalised weight of particle i; $\sum_i w_k^{(i)} = 1$	—	$[0, 1]$
$p(\boldsymbol{x}_k \mid \boldsymbol{z}_{1:k})$	Posterior probability density over vessel state given all position fixes up to time step k	—	$\mathbb{R}_{\geq 0}$
$\boldsymbol{z}_{1:k}$	Sequence of all position fixes from time step 1 to k	—	$\mathbb{R}^{2\times k}$
$\delta(\cdot)$	Dirac delta function	—	—
Motion Model			
$\boldsymbol{u}_k$	Target waypoint position (control input)	[m]	$\mathbb{R}^2$
$f(\cdot)$	Nonlinear state-transition (dynamics) function	—	$\mathbb{R}^4 \to \mathbb{R}^4$
$\boldsymbol{\eta}_k$	Process noise vector, $\boldsymbol{\eta}_k \sim \mathcal{N}(\mathbf{0}, \boldsymbol{Q})$	[mixed]	$\mathbb{R}^4$
$\mathcal{N}(\mu, \sigma^2)$	Normal distribution with mean μ and variance σ^2	—	—
$\boldsymbol{Q}$	Process noise covariance matrix, $= \text{diag}(\sigma_p^2, \sigma_p^2, 0, 0)$	[m²; m²; —; —]	$\mathbb{R}^{4\times 4}$
Δt	Time interval between consecutive time steps	[s]	$\mathbb{R}_{>0}$
$\psi_{k'}$	Updated heading after applying control change and noise	[°]	$\mathbb{R}$
$\Delta\psi_k$	Bounded heading change toward the target waypoint	[°]	$[-\dot{\psi}_{\max}, \dot{\psi}_{\max}]$
$\dot{\psi}_{\max}$	Maximum allowed heading-change rate per time step	[°/k]	$\mathbb{R}_{>0}$
ε_ψ	Zero-mean Gaussian heading noise, $\varepsilon_\psi \sim \mathcal{N}(0, \sigma_\psi^2)$	[°]	$\mathbb{R}$
σ_ψ	Standard deviation of heading noise (compass/INS uncertainty)	[°]	$\mathbb{R}_{>0}$
σ_p	Standard deviation of position process noise (sea current etc.)	[m]	$\mathbb{R}_{>0}$

Symbol	Definition	Unit	Domain
Measurement Model			
$h(\boldsymbol{x}_k)$	Measurement function: extracts $(p_x, p_y)^\top$ from the state	—	$\mathbb{R}^4 \to \mathbb{R}^2$
$\boldsymbol{\nu}_k$	Measurement noise, $\boldsymbol{\nu}_k \sim \mathcal{N}(\mathbf{0}, \boldsymbol{R})$	[m]	$\mathbb{R}^2$
$\boldsymbol{R}$	Measurement noise covariance matrix, $= \sigma_r^2 \boldsymbol{I}_2$	[m²]	$\mathbb{R}^{2\times2}$
$\boldsymbol{I}$	Identity matrix (2×2)	—	$\mathbb{R}^{2\times2}$
σ_r	Measurement noise standard deviation (method-specific)	[m]	$\mathbb{R}_{>0}$
$\hat{\boldsymbol{z}}_k^{(i)}$	Predicted position fix for particle i: $h\left(\boldsymbol{x}_k^{(i)}\right)$	[m]	$\mathbb{R}^2$
$p\left(\boldsymbol{z}_k \mid \boldsymbol{x}_k^{(i)}\right)$	Gaussian likelihood of the measurement given particle i's state	[m⁻²]	$\mathbb{R}_{\geq0}$
$\exp(\cdot)$	Exponential function	—	$\mathbb{R} \to \mathbb{R}_{>0}$
Weighting			
$\propto$	Proportional to	—	—
Resampling			
N_{eff}	Effective sample size $= 1/\sum_i \left(w_k^{(i)}\right)^2$	—	$[1, N]$
U	Single uniform draw used in systematic resampling	—	$\mathcal{U}[0, 1)$
u_i	Resampling position for particle i: $(i - 1 + U)/N$	—	$[0, 1)$
σ_{pos}	Std. dev. of Gaussian roughening noise added to resampled positions	[m]	$\mathbb{R}_{>0}$
Measurement Preprocessing			
w, h	Width and height of the measurement	[px]	$\mathbb{Z}_{>0}$
$C = (C_x, C_y)$	Centre of the measurement $= \left(\frac{w-1}{2}, \frac{h-1}{2}\right)$	[px]	$\mathbb{R}^2$
t_x, t_y	Translation components of the rotation transform about C	[px]	$\mathbb{R}$
x, y	Input pixel coordinates in M_k	[px]	$\mathbb{Z}^2$
x', y'	Output pixel coordinates in $\widetilde{M}_k$ after rotation	[px]	$\mathbb{Z}^2$
$\boldsymbol{R}_{\text{rot}(\psi_k)}$	2D rotation matrix by $-\psi_k$ used to produce north-aligned $\widetilde{M}_k$	—	$\mathbb{R}^{2\times2}$
Search Region Extraction			
$\overline{\boldsymbol{p}}_k$	Weighted mean particle position (centre of search region S_k)	[m]	$\mathbb{R}^2$
$\overline{x}_k, \overline{y}_k$	x- and y-components of the weighted mean position	[m]	$\mathbb{R}$
$p_{x,k}^{(i)}, p_{y,k}^{(i)}$	Position components of particle i after prediction at time step k	[m]	$\mathbb{R}$
$\sigma_{x,k}, \sigma_{y,k}$	Weighted standard deviations of the particle cloud in x and y	[m]	$\mathbb{R}_{\geq0}$

Symbol	Definition	Unit	Domain
σ_k^{sr}	Scalar positional spread = $\max(\sigma_{x,k}, \sigma_{y,k})$	[m]	$\mathbb{R}_{\geq 0}$
r_k	Search region radius, clamped to $[r_{\min}, r_{\max}]$	[m]	$[r_{\min}, r_{\max}]$
$r_{\min}$	Minimum search radius	[m]	$\mathbb{R}_{>0}$
$r_{\max}$	Maximum search radius	[m]	$\mathbb{R}_{>0}$
NCC Matching			
$\widetilde{M}'_k$	Mean-subtracted north-aligned measurement	[m]	$\mathbb{R}^2 \to \mathbb{R}$
$S'_k(x, y)$	Locally mean-subtracted search region at candidate position (x, y)	[m]	$\mathbb{R}^2 \to \mathbb{R}$
$\overline{\widetilde{M}}_k$	Global mean depth of $\widetilde{M}_k$	[m]	$\mathbb{R}$
$\overline{S}_k(x, y)$	Local mean of S_k over the window at (x, y)	[m]	$\mathbb{R}$
$\rho(x, y)$	Normalised cross-correlation score at candidate position (x, y)	—	$[-1, 1]$
(x^*, y^*)	Pixel position of the best NCC match = $\arg\max_{(x,y)} \rho(x, y)$	[pixels]	$\mathbb{Z}^2$
$\boldsymbol{z}_k^{\mathrm{NCC}}$	Position fix produced by the NCC method	[m]	$\mathbb{R}^2$
SP+LG Matching			
s	Upscaling factor applied before feature extraction	—	$\mathbb{Z}_{>0}$
p_2, p_{98}	2nd and 98th depth percentiles used for joint normalisation	[m]	$\mathbb{R}$
$\mathcal{F}$	Feature extractor (SuperPoint)	—	—
$\mathcal{K}_M$	Set of N_m keypoints detected in $\widetilde{M}_k$	[px]	$\subset \mathbb{R}^2$
$\mathcal{K}_R$	Set of N_r keypoints detected in S_k	[px]	$\subset \mathbb{R}^2$
$\boldsymbol{k}_a^{M_k}, \boldsymbol{k}_b^R$	Individual keypoint coordinates in $\widetilde{M}_k$ and S_k respectively	[px]	$\mathbb{R}^2$
N_m, N_r	Number of keypoints detected in $\widetilde{M}_k$ and S_k respectively	—	$\mathbb{Z}_{\geq 0}$
$\mathcal{L}$	Keypoint matcher (LightGlue)	—	—
$\mathcal{M}_k$	Set of N_c keypoint correspondences with confidence scores	—	—
a_l, b_l	Indices into $\mathcal{K}_M$ and $\mathcal{K}_R$ for the l-th matched pair	—	$\mathbb{Z}_{\geq 0}$
c_l	Confidence score of the l-th match	—	$[0, 1]$
N_c	Total number of correspondences returned by LightGlue	—	$\mathbb{Z}_{\geq 0}$
$\boldsymbol{m}_l, \boldsymbol{r}_l$	Matched keypoint coordinates in $\widetilde{M}_k$ and S_k for match l	[px]	$\mathbb{R}^2$
$\boldsymbol{\delta}_l$	Displacement vector for match l: $\boldsymbol{r}_l - \boldsymbol{m}_l$	[px]	$\mathbb{R}^2$
δ_l^x, δ_l^y	x- and y-components of the l-th displacement vector	[px]	$\mathbb{R}$
$N_m^{\max}$	Maximum keypoint limit for $\widetilde{M}_k$	—	$\mathbb{Z}_{>0}$
$N_r^{\max}$	Maximum keypoint limit for S_k	—	$\mathbb{Z}_{>0}$
$\hat{\boldsymbol{\delta}}_k$	Estimated translation: component-wise median of $\{\boldsymbol{\delta}_l\}$	[px]	$\mathbb{R}^2$
$\boldsymbol{c}$	Centre of the upscaled measurement = $\frac{s}{2}(W_m, H_m)^\top$	[px]	$\mathbb{R}^2$
W_m, H_m	Width and height of $\widetilde{M}_k$ in original pixels	[px]	$\mathbb{Z}_{>0}$

Symbol	Definition	Unit	Domain
$\boldsymbol{o}_k = (c_0, r_0)^\top$	Pixel origin (top-left corner) of S_k in $\hat{R}$	[px]	$\mathbb{Z}^2$
$\boldsymbol{p}_k$	Estimated position of the measurement centre in $\hat{R}$ (pixel coords)	[px]	$\mathbb{R}^2$
$N_{\min}$	Minimum matches required for a valid SP+LG fix	—	$\mathbb{Z}_{>0}$
$\boldsymbol{z}_k^{\text{SP+LG}}$	Position fix produced by the SuperPoint+LightGlue method	[m]	$\mathbb{R}^2$
Combined Matching			
μ_k	Weighted mean of particle states after prediction	[mixed]	$\mathbb{R}^4$
σ_k^{prior}	Weighted RMS positional spread of particles from μ_k	[m]	$\mathbb{R}_{\geq 0}$
d_k	Euclidean distance from NCC fix to prior mean μ_k	[m]	$\mathbb{R}_{\geq 0}$
τ_k	Adaptive NCC rejection threshold = $\max\left(\gamma \cdot \sigma_k^{\text{prior}}, \tau_{\min}\right)$	[m]	$\mathbb{R}_{>0}$
γ	Sigma factor for the rejection gate	—	$\mathbb{R}_{>0}$
$\tau_{\min}$	Minimum gate floor	[m]	$\mathbb{R}_{>0}$

List of Figures

Contents

1. Introduction

1.1. Motivation

Society relies on well-functioning Global Navigation Satellite Systems (GNSS), such as GPS, Galileo, GLONASS, and BeiDou [5]. GNSS provides continuous and accurate Positioning, Navigation, and Timing (PNT) services and is used across many sectors, including transport, energy, and military operations [5], [6], [7], [8], [9], [10]. As society becomes increasingly dependent on these services, disruptions can have far-reaching consequences and may pose a direct threat to society [11].

This dependence is particularly important in the military and maritime domains. Military operations rely on GNSS for activities such as guided ammunition, the movement of people and objects, and transport logistics. In maritime operations, GNSS supports situational awareness, safe and efficient route planning, navigation, and collision avoidance [5], [12], [13], [14]. The need for GNSS-independent navigation applies to both underwater and surface vessels. Underwater vehicles cannot receive GNSS signals while submerged, whereas surface vessels may lose access to GNSS because of jamming, spoofing, interference, or other disruptions, particularly in contested environments. When positioning and timing data are unavailable, delayed, or unreliable, vessel navigation can be compromised and military operations can be severely affected. These vulnerabilities have therefore received explicit attention at both national and international levels, underlining their urgency [5].

Despite this critical role, GNSS is inherently vulnerable. Its low-power signals are susceptible to degradation or denial from ionospheric disturbances, space weather, satellite clock errors, and both unintentional and intentional interference such as jamming and spoofing [5], [6], [7], [10], [12]. This makes it particularly unreliable in contested environments, where accurate positioning is most critical.

Therefore, robust maritime navigation alternatives for maritime navigation that operate independently of satellite infrastructure must be explored. In operational practice, GNSS outages in maritime navigation are often mitigated by Dead Reckoning (DR), celestial navigation, visual observations, magnetic compasses, Long Range Navigation (LORAN), and Inertial Navigation Systems (INS). However, these approaches are time-consuming, depend on environmental conditions and human interpretation, or often lack the accuracy required for near real-time and high-precision navigation. INS can provide short-term autonomy, but its errors accumulate over time without external correction. LORAN and similar alternatives depend on fixed infrastructure that must be present in the operational area, which is costly to deploy and maintain.

Together, these limitations highlight the need for an accurate, autonomous, and infrastructure-independent maritime navigation solution that remains operational in GNSS-denied environments. This need is directly relevant in the context of CGI's collaboration with the Royal Netherlands Navy, which aims to support the development of robust maritime navigation solutions for military operations. Within this broader context, this thesis investigates seabed fingerprinting as a potential solution for GNSS-independent maritime navigation.

Seabed fingerprinting is well suited to this problem because it uses the seabed itself as a natural navigation reference. The seabed is widely available and contains distinctive bathymetric patterns that can serve as location-specific fingerprints over operational timescales. By matching near real-time Multibeam Echosounder (MBES) measurements against a pre-existing bathymetric reference map, a vessel's position can be estimated repeatedly during navigation. These repeated position fixes can be used to limit INS drift without relying on satellite signals or externally deployed infrastructure. Unlike LORAN, seabed fingerprinting does not require fixed infrastructure in the operational area, and unlike GNSS, it is not directly vulnerable to satellite-signal jamming or spoofing. Its main onboard sensing requirement is an MBES. Although this thesis focuses on military surface vessels, the proposed seabed-fingerprinting approach is applicable to both surface vessels and underwater vehicles, provided that suitable bathymetric measurements, heading information, and a reference map are available.

1.2. Objective and Research Questions

The primary objective of this thesis is to develop and evaluate a framework for maritime navigation in GNSS-denied environments. The framework uses near real-time MBES measurements and bathymetric reference data to estimate vessel position at each time step, enabling continuous navigation without relying on satellite signals. To be practically applicable, the positioning methods within the framework must be accurate, robust, and computationally efficient enough to support near real-time maritime operation. Based on these objectives, the following main research question is formulated:

Main Research Question
How can seabed fingerprinting, using near real-time MBES measurements matched against bathymetric reference data, enable position estimation for maritime vessel navigation in GNSS-denied environments, while ensuring robustness, accuracy, and computational efficiency?

Research Sub-questions
To systematically address the main research question, the study is structured around three sub-questions, each targeting a critical aspect of the thesis:

Research Sub-question 1
How can an end-to-end seabed-aided maritime navigation framework be designed for GNSS-denied environments?

Research Sub-question 2
How can seabed fingerprinting be used for vessel position estimation within this framework?

Research Sub-question 3
How does the proposed framework perform when applied to realistic bathymetric data, in terms of accuracy, robustness, and computational efficiency?

1.3. Thesis Outline

The remainder of this thesis is organized as follows.

- Section 2 provides the theoretical background needed to understand the paper in Section 3. It introduces GNSS-denied environments, maritime navigation, DR, INS, Terrain-Aided Navigation (TAN), seabed fingerprinting, bathymetry, MBES, image matching, and particle filtering. Together, these concepts form the foundation needed to understand the paper in Section 3.
- Section 3 presents the paper, which introduces SAND-E: an end-to-end particle filter framework for seabed-aided maritime navigation in GNSS-denied environments. The paper describes the framework, the three positioning methods evaluated (NCC, SP+LG, and a combined prior-gated approach), and the fusion of the position fixes into a continuous navigation solution. Experiments are conducted on North Sea and Atlantic Ocean bathymetry across three geographically distinct test areas, evaluating positioning accuracy, robustness to a three-year old reference map, generalization beyond the primary test area, and computational efficiency.
- Section 4 provides a closing discussion on the relevance of this work to MSc Geomatics for the Built Environment, and outlines recommendations for future work and operational deployment.

2. Theoretical Background

Section 1.1 established that reliable maritime navigation in GNSS-denied environments requires an alternative that is accurate, autonomous, and independent of external infrastructure. Seabed fingerprinting addresses this by matching near real-time MBES measurements against a pre-existing bathymetric reference map to estimate vessel position without relying on satellite signals. This chapter introduces the key concepts needed to understand how this works and to follow the paper in Section 3, covering GNSS-denied environments, DR, INS, TAN, seabed fingerprinting, bathymetry, MBES, image matching, and the particle filter.

2.1. GNSS-Denied Environments

GNSS are collections of satellites, called constellations, that broadcast radio signals enabling receivers anywhere on earth to determine their PNT data with high accuracy [15]. GNSS has become foundational across a broad range of applications, from surveying and agriculture to autonomous vehicles and maritime navigation [16], the focus of this thesis.

When a vessel operates at sea, it relies on continuous and accurate positioning to navigate safely. In certain situations, however, satellite signals may be unavailable, unreliable, or deliberately disrupted. A military vessel operating in a conflict zone may face deliberate jamming, where an adversary broadcasts interference signals that overwhelm the GNSS receiver and prevent accurate positioning, or spoofing, where false satellite signals are transmitted to mislead the receiver into reporting an incorrect position [6]. Such situations are collectively referred to as GNSS-denied environments (Definition 1), motivating the need for alternative, infrastructure-independent navigation solutions such as seabed fingerprinting.

Definition 1 — GNSS-Denied Environment
An environment where GNSS signals are unavailable or unreliable, preventing systems from receiving accurate PNT data from satellites.

2.2. Maritime Navigation

The concept of navigation originated at sea, where it referred to guiding a vessel from origin to destination. With advances in science and technology, the term has since broadened to encompass aircraft, missiles, rockets, satellites, and spacecraft [16]. In this thesis, the focus is narrowed to maritime navigation, as defined in Definition 2.

Definition 2 — Maritime Navigation
The process of determining and maintaining the position, heading, and speed of a vessel at sea, requiring continuous and accurate PNT data to ensure safe and effective operation.

In practice, maritime navigation is executed by following a sequence of predetermined waypoints, intermediate target positions defining the planned route to the destination. Maintaining accurate positioning throughout this route is critical, and any disruption to PNT data, such as that caused by a GNSS-denied environment, directly compromises navigational safety and operational effectiveness.

In GNSS-denied environments, vessels can no longer rely on satellite signals for positioning. Within this thesis, positioning refers to the process of determining a vessel's position in two dimensions, referenced to a known coordinate system, as defined in Definition 3.

Definition 3 — Positioning
The process of determining a moving vessel's position in two dimensions, referenced to a standard geodetic coordinate system and expressed as a set of numerical coordinates.

2.3. DR and INS

In GNSS-denied environments, vessels fall back on DR, a self-contained navigation method that estimates the current position by integrating measured speed and heading forward from a known starting position [16], [17]. A key input to DR is heading, the compass direction in which the vessel's bow is pointed, typically measured by a gyrocompass or magnetic compass. A gyrocompass is a navigation instrument that uses a spinning gyroscope,

a rapidly rotating wheel that resists changes in orientation due to angular momentum, to find true north and maintain a stable heading reference independent of magnetic fields and vessel motion [17]. An INS combines gyroscopes with accelerometers, sensors that measure the specific force acting on a vessel along one or more axes. By integrating the measured accelerations twice over time, the INS derives velocity and position. Combined with the gyroscope measuring changes in orientation, the INS can continuously estimate the full motion state of the vessel without any external signals [17]. Both DR and INS require a known position at initialization and accumulate errors over time, causing the position estimate to drift progressively further from the true position, as illustrated in Figure 2.1, making them insufficient for long-term navigation without external correction [17], [18], [19]. In the framework presented in this thesis, heading from a gyrocompass or a magnetic compass is used as input at each time step, while position is estimated entirely by the seabed fingerprinting framework.

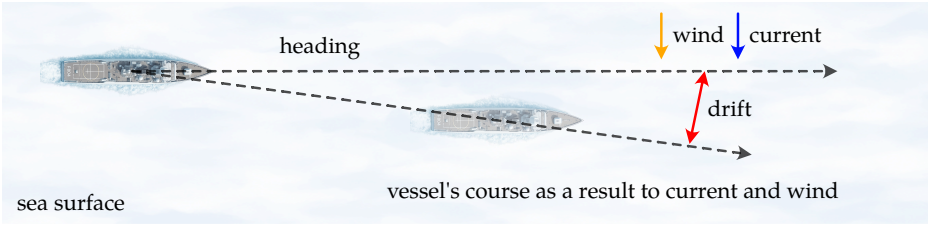


Figure 2.1: Vessel's course as a result of current and wind.

2.4. TAN and Seabed Fingerprinting

TAN offers a solution to this problem [18], [19]. Rather than estimating position from movement alone, TAN corrects accumulated drift by matching near real-time sensor measurements against a pre-existing reference map of the environment [18], [19]. The seabed is a particularly suitable terrain reference, as it is universally accessible, requires no external infrastructure, and exhibits unique spatial patterns that can be used for position estimation. Examples of the unique patterns are illustrated in Figure 2.2. Unlike TAN applied to aerial or land vehicles, which matches against topographic elevation, seabed-based TAN matches against water depth as measured by MBES, treating the problem as image matching. When applied to seabed terrain, this approach is referred to in this thesis as seabed fingerprinting.

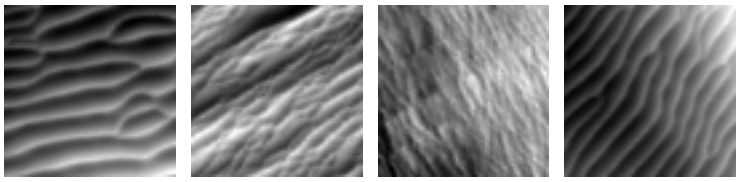


Figure 2.2: Four bathymetric measurements from different locations in the reference map. Each measurement exhibits a unique spatial pattern, illustrating why seabed features can serve as natural reference points for position estimation.

The term fingerprinting is borrowed from its most literal use: the practice of recording somebody's fingerprints, often used by the police to identify criminals, where a fingerprint is the mark made by the pattern of lines on the tip of a person's finger [20]. The concept of fingerprinting extends far beyond biometric and forensic practice. Across fields as diverse as molecular chemistry, audio/music processing, cybersecurity, and indoor positioning, fingerprinting consistently refers to the same idea: deriving a stable, distinctive signature from complex data and matching it against stored signatures for identification [21], [22], [23], [24], [25].

While classical fingerprinting relies on a pre-built database of stored fingerprints, the approach proposed in Section 3 adapts the principle to a near real-time context. Geometric features are extracted directly from a bathymetric measurement and matched against a search area within an existing reference map. No separate fingerprint database is constructed, as the matching process is sufficiently fast for the marine navigation task at hand. The conceptual basis nonetheless remains the same: stable, distinctive signatures are derived from complex data and matched against known references for identification.

In this thesis, the term seabed fingerprinting is introduced to distinguish this approach from conventional stored-signature methods such as Wi-Fi or audio fingerprinting, and it is formally defined in Definition 4.

Definition 4 — Seabed fingerprinting
A positioning method in which near real-time bathymetric MBES measurements are treated as raster images and matched against a pre-existing bathymetric reference map using image matching, exploiting distinctive geometric characteristics of the seabed to estimate vessel position.

2.5. Bathymetry and Multibeam Echosounder

Bathymetry is the study of seabed depth and morphology, the underwater equivalent of topography. Bathymetric surveys measure and map the shape and depth of the seabed using instruments such as an MBES, producing bathymetric maps that form the basis of nautical charts used for safe maritime navigation [26]. An MBES is an active SONAR (Sound Navigation and Ranging) system that emits acoustic pulses and measures the time for the echo to return after reflecting off a surface. Since sound travels at a known speed through water, this travel time is converted directly to depth. An MBES specifically emits multiple simultaneous pulses in a fan-shaped pattern perpendicular to the vessel's track, covering the seafloor both directly beneath the vessel and to each side, producing high-resolution bathymetric data [27], as illustrated in Figure 2.3. This data can be organized into a raster, a regular grid of depth values in which each cell encodes seafloor depth at a known geographic location, and aligning a near real-time bathymetric measurement against a pre-existing reference map can therefore be treated as an image matching problem. In the framework presented in this thesis, a pre-existing bathymetric reference map surveyed by the Hydrographic Service [28] serves as this reference. The following section introduces image matching, the specific techniques evaluated in Section 3, and their application within the particle-filter framework.

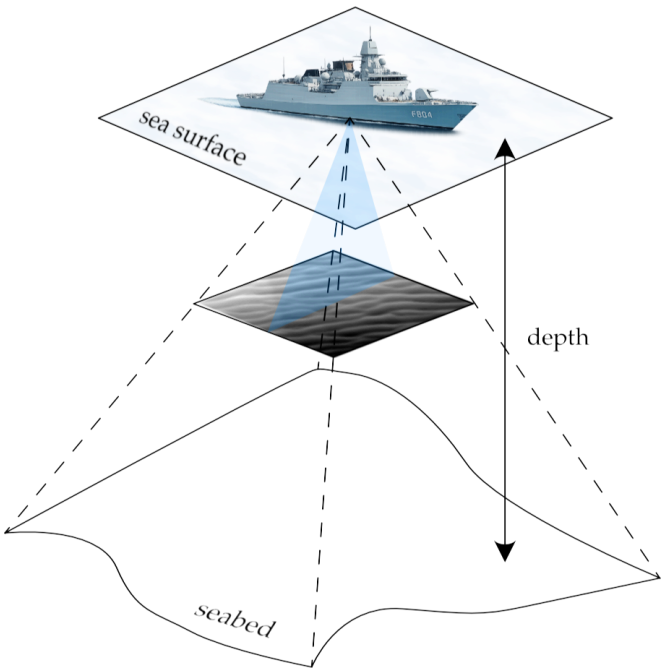


Figure 2.3: Fan-shaped MBES beam pattern, covering the seafloor directly beneath the vessel and to each side, producing a swath of depth measurements.

2.6. Image Matching

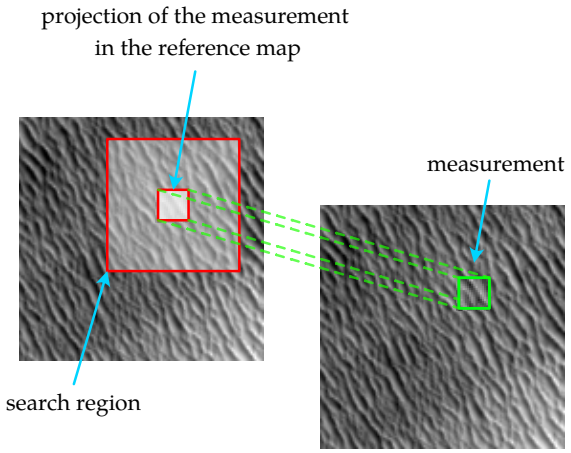
Image matching is a fundamental problem in computer vision concerned with finding correspondences between two or more images representing the same underlying structure [1]. Images may differ in viewpoint, scale, and illumination, all of which make matching challenging. In this thesis, only rotational differences are considered, as the bathymetric measurement and the reference map share the same spatial resolution of 2 m per pixel and are both acquired directly from above. The matching problem therefore reduces to finding the corresponding pixel in the search region for each pixel in the measurement. This pixel location is converted to real-world coordinates to produce a position fix, which is then used by the particle filter in Section 3.

Image matching has historically been dominated by handcrafted methods such as SIFT (Scale-Invariant Feature Transform), SURF (Speeded Up Robust Features), and ORB (Oriented FAST and Rotated BRIEF), which perform well under moderate conditions but struggle with low-texture regions and repetitive patterns, both of which are common in seabed terrain [1]. With the advent of deep learning, the field has shifted toward data-driven

approaches leveraging convolutional neural networks and transformer-based attention mechanisms, achieving more robust feature detection and matching. Both classical and learned methods are evaluated in Section 3: NCC as a classical template matching baseline, SP+LG as a learned feature matching method, and a combination of both, within a particle filter framework for seabed-aided navigation.

2.6.1. Classical Image Matching

Template matching is the earliest image-matching approach and operates by sliding a template over all candidate positions within a search window and computing a similarity score at each position [21]. The classical positioning method evaluated in Section 3 is NCC. In this context, the template corresponds to a near real-time bathymetric measurement and the search window is a restricted region extracted from the bathymetric reference map. The similarity score is computed as the correlation between the mean-subtracted measurement and the overlapping mean-subtracted region of the reference map, normalized by the product of their standard deviations. This normalization ensures that the score is based on relative depth variations rather than absolute depth values, making it invariant to constant depth offsets between the measurement and reference map. The resulting score ranges from -1 to 1, where 1 indicates a perfect match. NCC is straightforward to implement and requires no training data. In Section 3, rotation is handled prior to matching by north-aligning the measurement, reducing the problem to a pure translation search.



$$R(x,y) = \frac{\sum_{x',y'}(T'(x',y') \cdot I'(x+x',y+y'))}{\sqrt{\sum_{x',y'} T'(x',y')^2 \cdot \sum_{x',y'} I'(x+x',y+y')^2}}$$

Figure 2.4: NCC template matching: the measurement (right) is slid over the search region (left) and a similarity score is computed at each position. The highest-scoring position is returned as the position fix. Adapted from [1].

2.6.2. Learned Image Matching

Compared to classical template matching, learned feature-based image matching uses neural networks to detect distinctive feature points and compute descriptors encoding their local appearance. Two key advantages over classical methods are that: (1) feature representations are learned from data rather than manually designed, and (2) self- and cross-attention mechanisms build globally context-aware descriptors, making them better suited for ambiguous and low-texture environments [1]. Matching occurs in three steps: feature points are detected in both the measurement and the search region; a descriptor encoding the local appearance is computed for each point; and the descriptors are matched across the two images to establish corresponding point pairs with confidence scores.

The positioning method evaluated is SP+LG. SuperPoint [2] is a deep learning model for keypoint detection and descriptor extraction. A keypoint is a distinctive location in an image, such as a distinctive local pattern or corner, that can be reliably detected. A descriptor encodes what that location looks like as a 256-dimensional vector representing the local appearance around the keypoint, enabling matching across images. Its architecture (shown in Figure 2.5) consists of a shared convolutional encoder, a keypoint detection head, and a descriptor head. It is trained using self-supervision by enforcing consistency of keypoints and descriptors across geometric transformations of real images, requiring no manually labelled data [2]. While SuperPoint outperforms classical methods such as SIFT and SURF in robustness and accuracy, it relies on local Convolutional Neural Network (CNN) descriptors, which allows for fast computation, but it also limits its ability to capture global context,

meaning it needs to be paired with a matcher such as LightGlue or SuperGlue to handle ambiguous or repetitive scenes effectively [1].

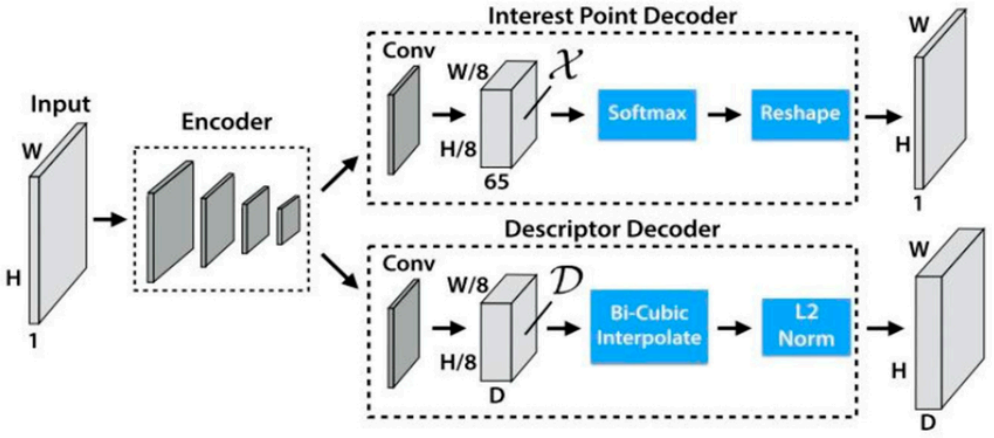


Figure 2.5: The SuperPoint pipeline applied to an image pair, extracting keypoints and descriptors from each image independently, which are subsequently passed to LightGlue for matching. Adopted from [2].

In Section 3, LightGlue is used as the matcher paired with SuperPoint. The SP+LG combination is widely used in computer vision as it balances matching accuracy with computational efficiency, making it suitable for real-time applications [29]. LightGlue [3] is a lightweight attention-based matcher that takes keypoints and descriptors from both images as input and establishes correspondences using self- and cross-attention mechanisms. LightGlue has a simple design with fewer transformer blocks and optimized parallel self- and cross-attention mechanisms compared to SuperGlue [30]. It incorporates early stopping when high-confidence matches are already found, avoiding unnecessary computation. This reduces inference time and memory consumption while retaining comparable matching accuracy, making it suitable for real-time navigation applications.

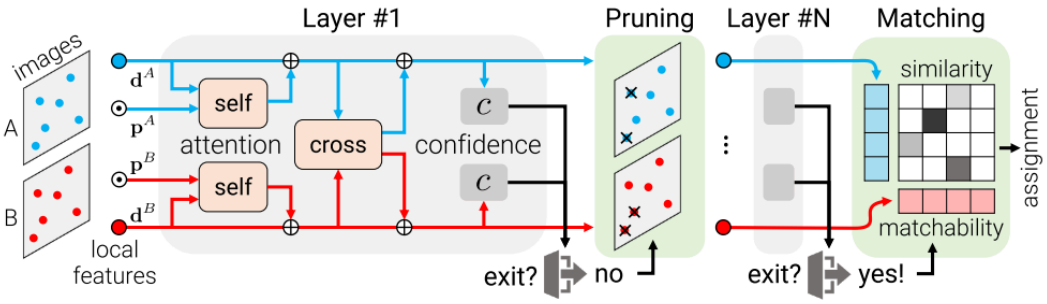


Figure 2.6: The LightGlue matching pipeline, taking keypoints and descriptors from two images as input, applying self- and cross-attention across multiple layers, and establishing correspondences through similarity and matchability scoring. Adopted from [3].

2.7. Particle Filter

A Particle Filter (PF) is widely used in navigation applications, from robotics to indoor positioning and TAN [18], [19], because it performs well under the nonlinear motion and high uncertainty characteristic of GNSS-denied environments. Image matching alone produces a position fix at each time step but cannot maintain a continuous and accurate position estimate over time. The framework presented in Section 3 therefore uses a particle filter to fuse successive position fixes with a motion model, correcting accumulated drift and maintaining a probabilistic estimate of the vessel state, which is a distribution over possible states rather than a single value, reflecting the uncertainty in the estimate. For nonlinear systems such as vessel navigation, the posterior cannot be computed exactly. The particle filter solves this by approximating it with a set of weighted samples, requiring no assumptions about the shape of the distribution.

The particle filter generates so-called particles. Each particle represents a hypothesis about the current vessel state, comprising its position, heading, and speed, with an associated weight reflecting how likely that hypothesis is. Together, the weighted particles approximate the full probability distribution over possible vessel states. The more particles are concentrated in a region, the more likely it is that the vessel is located there. Unlike a Kalman Filter (KF), which assumes that the uncertainty follows a Gaussian distribution, the particle filter makes no

such assumption. This makes it robust to occasional false positive position fixes, which can temporarily shift the distribution but are corrected by subsequent measurements [31].

Particle Initialization. Before the filter begins, all particles are initialized at the known vessel position at the point of GNSS loss, each with equal weight.

Prediction. At each time step, every particle is propagated forward using a motion model. The vessel moves according to its current heading and speed, with small random noise added to account for unmodelled dynamics such as sea currents. This spreads the particle cloud slightly, reflecting growing uncertainty between measurements.

Weighting. When a position fix arrives from image matching, each particle is reweighted based on how close its predicted position is to the fix. Particles near the fix receive high weights, whereas particles farther away receive low weights. The weights are then normalized so they sum to one [31]. The updated weighted particle set now represents the posterior, which is the belief about the vessel state after incorporating a new measurement. This contrasts with the prior, which is the predicted belief before the measurement arrives, produced by the prediction step [31]. The effect of measurement noise on the posterior is illustrated in Figure 2.7.

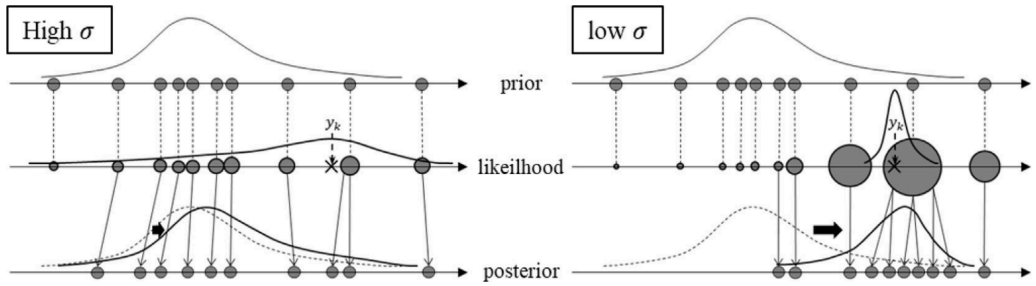


Figure 2.7: Effect of measurement noise on the posterior. A high σ (left) produces a broad, uncertain posterior, while a low σ (right) produces a sharp, concentrated posterior. Adopted from [4].

Resampling. Over time, most particles receive near-zero weight while a few dominate, a problem known as degeneracy. Resampling addresses this by drawing a new set of particles in proportion to their weights. High-weight particles are duplicated, whereas low-weight particles are discarded. Small roughening noise is added to prevent particle collapse. This cycle is illustrated in Figure 2.8, where the particle cloud expands during prediction, particles are reweighted based on the measurement likelihood, and the distribution is refocused through resampling.

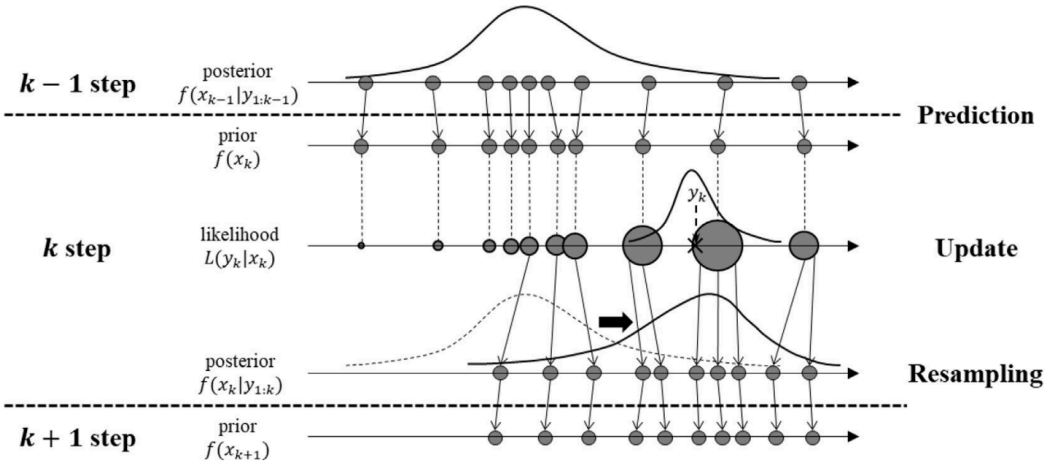


Figure 2.8: The particle filter cycle showing prediction, weighting, and resampling steps. At each time step, particles are propagated forward, reweighted based on the measurement likelihood, and resampled to focus on the most likely states. Adopted from [4].

3. Paper

Page left intentionally blank.

Seabed Fingerprinting for Maritime Navigation in GNSS-Denied Environments

SAND-E: Seabed-Aided Navigation Using Classical and Learned Image Matching

Julia Pille[1], Liangliang Nan[1], Roderik C. Lindenbergh[1] & Robert L. Voûte[1,2]
[1] Delft University of Technology [2] CGI

Abstract

Maritime navigation relies heavily on Global Navigation Satellite Systems (GNSS), yet military surface vessels must remain operational when satellite signals are unavailable, degraded, or denied. In such GNSS-denied environments, Inertial Navigation Systems (INS) accumulate unbounded drift, while existing Terrain-Aided Navigation (TAN) methods remain sensitive to terrain distinctiveness and are rarely evaluated for surface vessels. We present SAND-E, a particle-filter framework for seabed-aided maritime navigation that treats seabed fingerprinting as an image matching problem. Near real-time Multibeam Echosounder (MBES) measurements are matched against bathymetric reference maps using Normalized Cross-Correlation (NCC), SuperPoint+LightGlue (SP+LG), or a combined prior-gated method, and the resulting position fixes are fused into the particle filter for recursive state estimation. Evaluated on North Sea and Atlantic Ocean bathymetry, NCC outperforms SP+LG across all metrics, achieving an RMSE of 92.1 m, a 100% fix rate, 92.6% of runs within 500 m, and a runtime of 0.5 ms per fix. The combined method matches NCC under nominal conditions but provides additional robustness with an outdated reference map, where the prior gate rejects degraded NCC fixes and falls back to SP+LG. The framework generalizes across three geographically distinct test areas, remains viable with a three-year-old reference map, and reduces average final position error from Dead Reckoning (DR) to 115.3 m, demonstrating seabed fingerprinting as a viable infrastructure-independent navigation solution for GNSS-denied military surface vessels.

Keywords: GNSS-Denied Maritime Navigation, Seabed-Aided Navigation, Bathymetric Image Matching, Particle Filter, Multibeam Echosounder.

1. Introduction

The need for Global Navigation Satellite System (GNSS)-independent navigation applies to both underwater and surface vessels. Underwater vehicles cannot receive GNSS signals while submerged and therefore depend on inertial navigation and occasional external position updates. Surface vessels normally have access to GNSS, but may lose it because of jamming, spoofing, interference, or other disruptions, particularly in contested environments. Military surface vessels operating under such conditions rely primarily on Inertial Navigation Systems (INS), a form of Dead Reckoning (DR). However, INS suffers from unbounded position error growth over time, causing position estimates to drift progressively farther from the true position and making it insufficient for long-term navigation [1, 2].

Terrain-Aided Navigation (TAN) addresses this limitation by matching real-time sensor measurements against a pre-existing reference map to bound accumulated drift without relying on external systems such as satellites or ground-based beacons [3, 4]. Originally developed for aircraft and cruise missiles, TAN has since found broader applications, including underwater navigation [5]. A Multibeam Echosounder (MBES) is the primary sensor for acquiring bathymetric measurements in TAN, providing high-resolution depth data suitable for real-time implementation [6, 7]. The seabed is a particularly suitable terrain reference because it is widely accessible and exhibits distinctive spatial patterns that can support infrastructure-independent position estimation.

Although existing TAN methods can bound INS drift, their performance depends strongly on terrain distinctiveness [1, 3]. Seabed-aided navigation is relevant to both underwater and surface vessels. However, most existing research focuses on underwater vehicles, leaving GNSS-independent navigation for military surface vessels less explored. Moreover, learned image-matching methods have not been systematically compared with classical handcrafted methods within a particle-filter framework for seabed-aided navigation.

To address this gap, we introduce SAND-E (Seabed-Aided Navigation in GNSS-Denied Environments), an end-to-end particle filter framework for seabed-aided

maritime navigation. SAND-E uses seabed fingerprinting, in which near real-time MBES measurements and a pre-existing bathymetric reference map are both treated as raster images and matched against each other to estimate vessel position. Because individual image-matching results provide only discrete position fixes, they are not sufficient for continuous navigation on their own. SAND-E therefore embeds these fixes in a particle filter, which propagates the vessel state between measurements, represents uncertainty, and recursively corrects accumulated DR drift. Within this framework, we design and evaluate three positioning models that represent complementary image-matching strategies for seabed fingerprinting: Normalized Cross-Correlation (NCC) as a classical correlation-based baseline, SuperPoint + LightGlue (SP+LG) as a modern learned local-feature matcher, and a prior-gated combination of both to test whether classical and learned matching can complement each other within the particle filter. SAND-E is developed and parameterized on North Sea bathymetry.

To summarize, the following contributions are made: (1) We introduce SAND-E, requiring only onboard MBES measurements and a pre-existing bathymetric reference map. (2) We formulate seabed fingerprinting as an image-matching problem and integrate the resulting position fixes into a recursive navigation framework with uncertainty-adaptive search-region extraction. (3) We evaluate SAND-E on North Sea and Atlantic Ocean bathymetry across three geographically distinct test areas and a three-year-old reference map, demonstrating that NCC outperforms SP+LG on every metric with a Root Mean Square Error (RMSE) of 92.1 m, a 100% fix rate, and a runtime of 0.5 ms per fix, that the combined method provides robustness for harder operating conditions through prior-gated outlier rejection, and that SAND-E reduces average final position error from DR to 115.3 m in the Monte Carlo analysis.

The code is publicly available at https://github.com/juliapille/SAND-E as of June 2027, following a one-year embargo, to support reproducibility and facilitate further research on accurate, robust, and computationally efficient seabed-aided maritime navigation in GNSS-denied environments.

2. Related Work

This section reviews the existing literature and identifies the gaps that motivate this work.

2.1. Terrain Matching

Terrain matching is fundamentally an optimization problem that maximizes the similarity between real-time and reference data [3], encompassing correlation-based, filter-based, combined, and deep-learning-based methods.

2.1.1. Correlation-Based Methods

Correlation-based methods estimate position by finding the highest similarity between real-time terrain elevation data and a reference map through continuous search or iterative processes, with Terrain Contour Matching (TERCOM) and Iterative Closest Contour Point (ICCP) being the most widely adopted approaches, where TERCOM uses a grid-based search and ICCP refines position through iterative contour matching. While correlation-based methods are simple and effective on nonlinear terrain, they suffer from poor real-time performance and sensitivity to heading errors, requiring careful path planning to avoid feature-poor terrain regions [1, 3].

2.1.2. Filter-Based Terrain Matching

Filter-based terrain matching algorithms continuously estimate and correct INS position and velocity errors as seabed terrain data is acquired. These methods are rooted in a Bayesian Filter (BF) and include parametric approaches such as Kalman Filter (KF), Extended Kalman Filter (EKF), and Unscented Kalman Filter (UKF), as well as non-parametric approaches like Particle Filter (PF) and Point Mass Filter (PMF) [1]. While EKF offers the best real-time performance, terrain nonlinearity limits its accuracy, and KF's assumptions of linearity and Gaussian noise break down in real conditions [1, 8]. UKF, PF, and PMF achieve better results, with PF being the only method capable of relocating the vehicle under large initial errors [1]. PF has been widely adopted across many navigation domains, including estimating drift tracks of underwater vehicles and vision-based navigation on segmented aerial images, and positioning in flat terrain regions where feature scarcity typically degrades performance [9, 10, 11]. Although BF is theoretically optimal, its computational cost can be prohibitive for long-duration missions [3]. Maximum Likelihood Estimation (MLE) addresses this by converting terrain matching into a maximum likelihood problem, requiring minimal prior information and showing robustness to early localization errors, while Maximum A Posteriori (MAP) extends this by incorporating a prior probability density function, improving accuracy at the cost of increased computation [3].

2.1.3. Combined Terrain Matching

Combined methods have been widely explored, pairing correlation-based and filter-based techniques to exploit the strengths of each. Common combinations include TERCOM with PMF [12], ICCP with KF [13], and the correlator method within a PF framework [14]. To address large initial errors, Particle Swarm Optimization (PSO) and Artificial Bee Colony (ABC) optimization have been integrated with ICCP [4], [15], while [11] incorporates INS error characteristics for smooth terrain regions.

2.1.4. Deep Learning-Based Terrain Matching

Deep learning-based methods have been introduced to overcome the limitations of classical terrain matching. Progressive Learning Network (PL-Net) [16] extracts keypoints and applies a pretrained network with self-attention for robust real-time localization, while [2] generated discriminant descriptors directly from bathymetric point cloud patches, avoiding information loss from image conversion. Image enhancement techniques [17] further improve matching under degraded conditions. More broadly, deep learning has shown strong potential for feature detection and matching across domains [18, 19], with pretrained models improving generalization [20]. SuperPoint [21] is a fast and reliable keypoint detector and descriptor, typically paired with LightGlue [22], an attention-based matcher that reduces inference time and memory consumption for real-time use. SuperPoint + LightGlue achieves near-optimal matching performance across robustness benchmarks [23]. It outperforms handcrafted methods such as SIFT (Scale-Invariant Feature Transform) [24] and generalizes effectively across domains [25, 26].

While TAN has demonstrated the ability to bound INS drift and achieve meter-level accuracy, its performance depends strongly on terrain distinctiveness and may degrade in flat or poorly mapped regions [1, 3]. Seabed-aided navigation is relevant to both underwater and surface vessels. However, most existing research focuses on underwater vehicles, leaving its application to military surface vessels less well established. Classical correlation-based methods are widely used for terrain matching, while learned image-matching methods have shown strong performance in other computer-vision domains. However, it remains unclear how well these learned methods transfer to bathymetric raster data, which often contains gradual depth variations, repetitive patterns, and limited texture. This study therefore investigates the practical suitability of classical and learned image-matching methods as positioning models within a particle-filter framework and examines whether combining them can improve robustness under challenging navigation conditions.

3. Method

We introduce SAND-E, with an overview shown in Figure 3.1. The remainder of this chapter is organized as follows: Section 3.1 defines the problem and notation, Section 3.2 describes reference map preprocessing, Section 3.3 presents the particle filter, and Section 3.4 details the position fix methods.

3.1. Problem Description and Notation

A vessel navigates a GNSS-denied maritime environment using only a pre-existing bathymetric reference map $R : \Omega_R \subset \mathbb{R}^2 \to \mathbb{R}$ and onboard MBES measurements $M_k : \Omega_{M_k} \subset \mathbb{R}^2 \to \mathbb{R}$, where both are 2.5D raster surfaces treated as images with depth values interpreted as pixel intensities. Since position fixes alone are insufficient for continuous navigation, the full vessel state $\boldsymbol{x}_k = \left(p_x, p_y, \psi, v\right)^\mathsf{T} \in \mathbb{R}^4$, comprising horizontal position $\left(p_x, p_y\right)$, heading ψ obtained from a compass or INS, and constant speed over ground v, is maintained by a particle filter. At each time step k, a position fix $\boldsymbol{z}_k \in \mathbb{R}^2$ is derived by matching the north-aligned measurement $\tilde{M}_k$ against a local search region $S_k \subset \hat{R}$ and fused into the particle filter following [8] to maintain a probabilistic estimate of $\boldsymbol{x}_k$ over time. Three positioning methods are evaluated: NCC, SP+LG, and a combined prior-gated approach, described in Section 3.4.

Assumptions. The framework operates under a set of assumptions regarding the reference map, MBES measurements, vessel state, and noise characteristics. Most critically, the vessel position is assumed known at the point of GNSS loss, enabling particle initialization, the reference map is assumed to be expressed in a north-up projected coordinate system, and heading is assumed available from a compass or INS at each time step, enabling north-alignment and motion model propagation. A full list of assumptions with justifications is given in Appendix A.

3.2. Data Preparation

The bathymetric reference map R is preprocessed before the navigation process is started, to remove interior NoData cells. A NoData cell is classified as interior if it is completely surrounded by valid depth cells, in other words it is not connected to the boundary of the raster through other NoData cells. To identify the interior NoData, a valid cell mask $\mathcal{M}_{\text{valid}}(i, j)$ is first constructed:

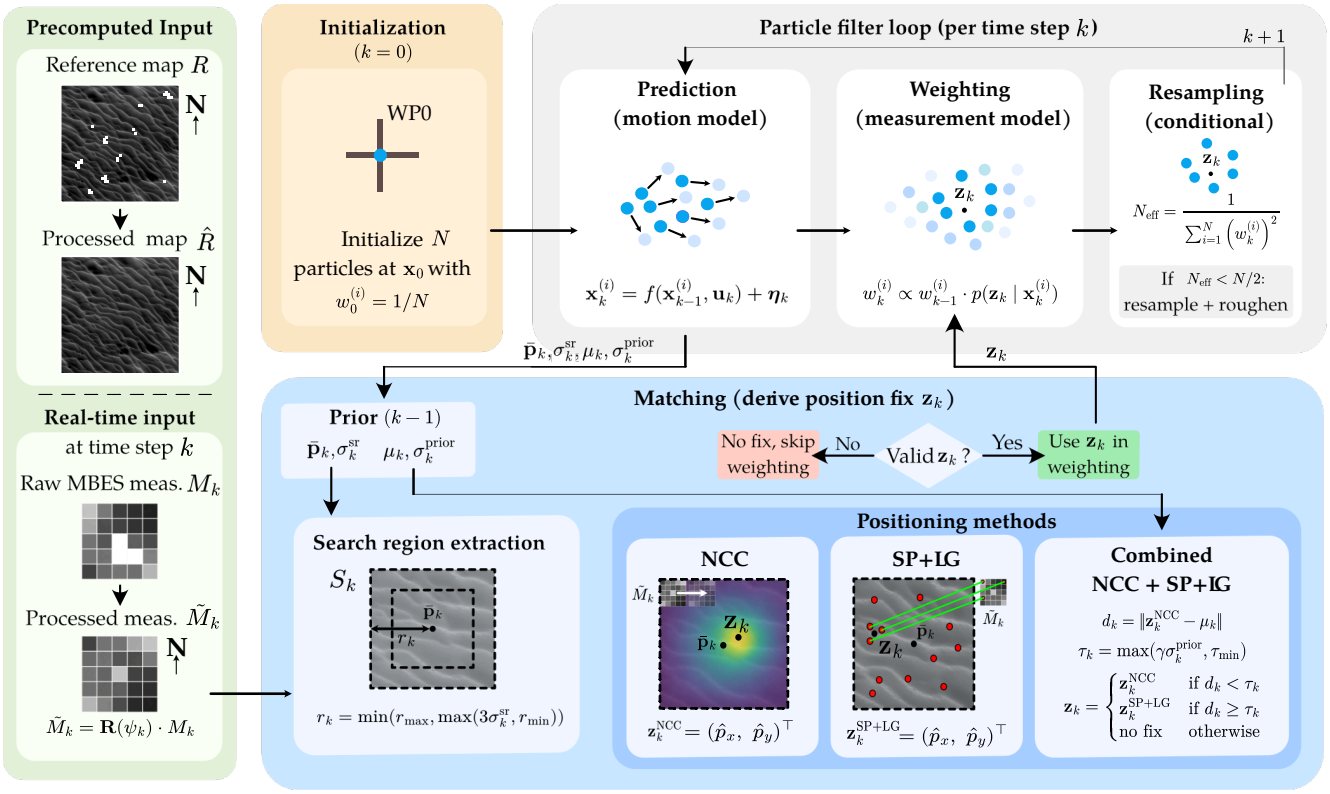


Figure 3.1: Overview of the SAND-E framework. A preprocessed bathymetric reference map and near real-time MBES measurements are used to derive position fixes through NCC, SP+LG, or a combined prior-gated positioning method, which are then fused in a particle filter for recursive state estimation.

$$\mathcal{M}_{\text{valid}}(i,j) = \begin{cases} 1 \text{ if } R(i,j) \neq d_{\text{nodata}} \\ 0 \text{ otherwise} \end{cases} \tag{3.1}$$

Starting from all NoData cells on the four raster borders, a flood-fill expands through 4-connected NoData neighbors, labelling all border-connected NoData cells as $\mathcal{M}_{\text{border}}$. The interior NoData mask is then:

$$\mathcal{M}_{\text{interior}(i,j)} = (1 - \mathcal{M}_{\text{valid}}(i,j)) \cdot (1 - \mathcal{M}_{\text{border}}(i,j)) \tag{3.2}$$

Each interior NoData cell is filled with the depth value of its nearest neighbor, identified via Euclidean distance transform [27]:

$$(i^*, j^*) = \arg\min_{\substack{(m,n): \\ \mathcal{M}_{\text{valid}}(m,n)=1}} \sqrt{(i-m)^2 + (j-n)^2} \tag{3.3}$$

$$\hat{R}(i,j) = \begin{cases} R(i^*, j^*) \text{ if } \mathcal{M}_{\text{interior}}(i,j) = 1 \\ R(i,j) \text{ otherwise} \end{cases} \tag{3.4}$$

Border-connected NoData regions are excluded from interpolation to preserve the original survey extent, while interior NoData cells are interpolated to prevent missing values from being misinterpreted as valid bathymetry during matching (Section 3.4).

3.3. Particle Filter

Particle filters are widely used in navigation and TAN because they can handle nonlinear motion and high uncertainty in GNSS-denied environments [1, 3]. In SAND-E, a particle filter is needed because seabed matching provides only discrete position fixes $\boldsymbol{z}_k$, while navigation requires a continuous estimate of the full vessel state $\boldsymbol{x}_k = (p_x, p_y, \psi, v)^\top$. The particle filter represents the posterior as a weighted set of particles, maintaining uncertainty when fixes are noisy or missing, and the predicted particle distribution defines the prior used to extract the local search region for matching. The filter cycles through prediction (Section 3.3.5), weighting (Section 3.3.6), and resampling with roughening (Section 3.3.7) at each time step, preceded by one-time initialization (Section 3.3.4). The motion and measurement models are defined in Section 3.3.2 and Section 3.3.3.

3.3.1. Particle Representation

We maintain a set of N such particles, each with an associated weight:

$$\left\{\boldsymbol{x}_k^{(i)}, w_k^{(i)}\right\}_{i=1}^{N}, \quad w_k^{(i)} \geq 0, \quad \sum_{i=1}^{N} w_k^{(i)} = 1 \tag{3.5}$$

where $w_k^{(i)}$ reflects how well particle $\boldsymbol{x}_k^{(i)}$ agrees with all past measurements. The weighted particle set approximates the posterior:

$$p(\boldsymbol{x}_k \mid \boldsymbol{z}_{1:k}) \approx \sum_{i=1}^{N} w_k^{(i)} \delta\left(\boldsymbol{x}_k - \boldsymbol{x}_k^{(i)}\right) \tag{3.6}$$

This non-parametric representation allows the filter to capture multimodal uncertainty without assuming a Gaussian distribution

3.3.2. Motion Model

The motion model describes how the state evolves from one time step to the next:

$$\boldsymbol{x}_k = f(\boldsymbol{x}_{k-1}, \boldsymbol{u}_k) + \boldsymbol{\eta}_k \tag{3.7}$$

where $\boldsymbol{u}_k$ is the target waypoint position toward which the vessel steers, $f(\cdot)$ is the nonlinear function describing the system dynamics, and $\boldsymbol{\eta}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q})$ is the process noise vector with covariance matrix $\boldsymbol{Q} \in \mathbb{R}^{4\times4}$. We define $f(\cdot)$ as:

$$f(\boldsymbol{x}_{k-1}, \boldsymbol{u}_k) = \begin{pmatrix} p_{x,k-1} + \upsilon \cos(\psi_{k'}) \cdot \Delta t \\ p_{y,k-1} + \upsilon \sin(\psi_{k'}) \cdot \Delta t \\ \psi_{k'} \\ \upsilon \end{pmatrix} \tag{3.8}$$

where $\psi_{k'} = \psi_{k-1} + \Delta\psi_k + \varepsilon_\psi$ is the updated heading, with $\Delta\psi_k$ the bounded heading change toward the target waypoint limited to $\dot{\psi}_{\max} = 55°$ per minute, and $\varepsilon_\psi \sim \mathcal{N}\left(0, \sigma_\psi^2\right)$ zero-mean Gaussian noise modelling compass or INS uncertainty and maintaining particle diversity. The position update inside f uses the noisy heading ε_ψ directly. The remaining process noise accounts for position uncertainty only:

$$\boldsymbol{Q} = \operatorname{diag}\left(\sigma_p^2, \sigma_p^2, 0, 0\right) \tag{3.9}$$

where σ_p^2 accounts for unmodelled dynamics such as sea current. The heading noise ε_ψ is handled separately inside $f(\cdot)$ and is therefore included in $\boldsymbol{Q}$. Speed has zero noise since υ is a known constant.

3.3.3. Measurement Model

The measurement model links the hidden state to the observable quantities. In vector form:

$$\boldsymbol{z}_k = h(\boldsymbol{x}_k) + \nu_k \tag{3.10}$$

where $\boldsymbol{z}_k \in \mathbb{R}^2$ is a position fix, $h(\boldsymbol{x}_k) = \left(p_x, p_y\right)^\top$ extracts the position components from the state vector, and $\boldsymbol{\nu}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R})$ is the measurement noise with covariance $\boldsymbol{R} = \sigma_r^2 \boldsymbol{I}^2$ where σ_r is the measurement noise standard deviation, which is method-specific and determined in Section 4.

Each particle predicts what the expected position fix would be if the vessel were at that particle's state:

$$\hat{\boldsymbol{z}}_k^{(i)} = h\left(\boldsymbol{x}_k^{(i)}\right) = \begin{pmatrix} p_x^{(i)} \\ p_y^{(i)} \end{pmatrix} \tag{3.11}$$

The likelihood of the actual measurement $\boldsymbol{z}_k$ given each particle's prediction is then computed as:

$$p\left(\boldsymbol{z}_k \mid \boldsymbol{x}_k^{(i)}\right) = \frac{1}{2\pi\sigma_r^2} \exp\left[-\frac{1}{2\sigma_r^2}\left\|\boldsymbol{z}_k - \hat{\boldsymbol{z}}_k^{(i)}\right\|^2\right] \tag{3.12}$$

These likelihoods are used as unnormalized weights in the particle weight update step.

3.3.4. Particle Initialization

Particles are initialized at a known starting position $\boldsymbol{x}_0$ with equal weights $w_0^{(i)} = \frac{1}{N}$, reflecting the realistic scenario where the vessel's position and heading are known at the point of GNSS loss. If the starting position is unknown, a best-guess may be used to initialize the particle cloud. Once the true starting position becomes available, the cloud can be shifted by the positional offset to recover an approximate absolute position estimate, though accumulated drift and heading noise during the interim period will have reduced estimation accuracy. The specific choice of N is given in Section 4.

3.3.5. Prediction

At each time step k, the motion model (Section 3.3.2) is applied to every particle independently, propagating the particle cloud forward in time before the measurement update.

3.3.6. Weighting

When a position fix $\boldsymbol{z}_k$ is returned by one of the positioning methods in Section 3.4, each particle is reweighted according to its likelihood. Each particle predicts the expected position fix if the vessel were at that particle's state (Equation (3.11)). The weight of each particle is then updated as:

$$w_k^{(i)} \propto w_{k-1}^{(i)} \cdot p\left(\boldsymbol{z}_k \mid \boldsymbol{x}_k^{(i)}\right) \tag{3.13}$$

Particles whose predicted position $\hat{\boldsymbol{z}}_k^{(i)}$ is close to $\boldsymbol{z}_k$ receive high weight and particles far from $\boldsymbol{z}_k$ receive low weight. The weights are then normalized so that $\sum_i w_k^{(i)} = 1$. The weighted particle set now approximates the posterior $p(\boldsymbol{x}_k \mid z_{1:k})$ (Equation (3.6)). If no valid position fix is returned, the weight update is skipped and the weights remain unchanged from the previous time step, meaning the particle cloud continues to propagate on the motion model alone.

3.3.7. Resampling

The degree of particle degeneracy is monitored using the effective sample size:

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^{N}\left(w_k^{(i)}\right)^2} \tag{3.14}$$

which equals N when all weights are equal and approaches 1 when a single particle dominates. N_{eff} estimates how many particles effectively contribute to

the posterior estimate. Resampling is only triggered when $N_{\text{eff}} < \frac{N}{2}$, preventing unnecessary resampling when the particle set is still healthy and diverse.

Systematic resampling draws N new particles by sampling positions $u_i = \frac{i-1+U}{N}$ with $U \sim \mathcal{U}[0,1)$ from the cumulative weight distribution, selecting particles with high weight multiple times and discarding those with near-zero weight. To prevent particle collapse, small Gaussian roughening noise with standard deviations σ_{pos} and σ_ψ is added to the resampled positions and headings respectively. All weights are then reset to $\frac{1}{N}$.

3.4. Matching

At time step k, the north-aligned measurement $\tilde{M}_k$ is matched against a search region $S_k \subset \hat{R}$ to derive a position fix $\boldsymbol{z}_k$, which is then fused into the particle filter as described in Section 3.3.6. The matching pipeline consists of three stages: measurement preprocessing (Section 3.4.1), search region extraction (Section 3.4.2), and position estimation.

For position estimation, three methods are evaluated: NCC (Section 3.4.3) as a classical template matching baseline, SP+LG (Section 3.4.4) as a learned feature matching method, and a combined prior-gated approach (Section 3.4.5) that falls back to SP+LG when NCC fixes are rejected.

3.4.1. Measurement Preprocessing

Missing depth values in M_k are filled using nearest-neighbor interpolation (Section 3.2), with no distinction between interior and border cells since the measurement boundary has no physical significance. As NoData values represent at most 2% of the experiment data, any impact on local texture and matching performance is expected to be negligible, and no maximum hole size is applied. However, dense NoData regions may introduce artificial flat areas that reduce local texture, potentially degrading both NCC correlation and SP+LG keypoint detection.

Since M_k is acquired in the vessel-aligned frame, matching it directly against the north-up search region S_k would cause incorrect correlation. M_k is therefore rotated by $-\psi_k$ around its center (as illustrated in Figure 3.2) $C = \left(\frac{w-1}{2}, \frac{h-1}{2}\right)$, where w and h are the measurement dimensions in pixels. Let $t_x = C_x(1-\cos\psi_k) + C_y \sin\psi_k$ and $t_y = C_y(1-\cos\psi_k) - C_x \sin\psi_k$, then:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \boldsymbol{R}_{\text{rot}(\psi_k)} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \quad \tilde{M}_k(x', y') = M_k(x, y) \tag{3.15}$$

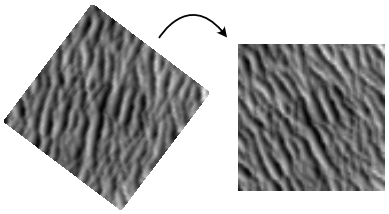


Figure 3.2: The measurement patch M_k acquired in the vessel-aligned frame (left) is rotated to north-up to produce $\tilde{M}_k$ (right), reducing the matching problem to a pure translation search.

where $(x, y)^\top \in \Omega_M$ is the input coordinate corresponding to output pixel $(x', y')^\top$. Reflected boundary padding [28] is applied during rotation to prevent depth discontinuities at measurement edges from corrupting the matching score. The resulting north-aligned measurement $\tilde{M}_k$ aligns the measurement with the reference map coordinate frame, reducing matching to a pure translation search.

3.4.2. Search Region Extraction

At each time step k, a search region S_k is constructed from $\hat{R}$ based on the current particle distribution, restricting matching to the area where the vessel is most likely located, shown in Figure 3.3 .

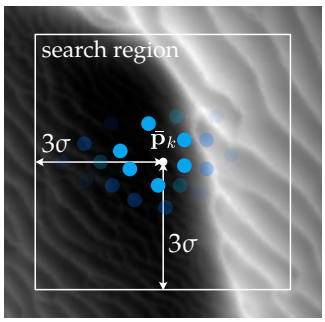


Figure 3.3: At each time step k, a search region S_k is constructed from $\hat{R}$ based on the current particle distribution, restricting matching to the area where the vessel is most likely located.

The center of S_k is the weighted mean particle position $\overline{\boldsymbol{p}}_k$, representing the predicted vessel location based on all measurements up to the time step $k-1$:

$$\overline{\boldsymbol{p}}_k = \begin{pmatrix} \overline{x}_k \\ \overline{y}_k \end{pmatrix} \tag{3.16}$$

$$\overline{x}_k = \sum_{i=1}^{N} w_{k-1}^{(i)} p_{x,k}^{(i)}, \quad \overline{y}_k = \sum_{i=1}^{N} w_{k-1}^{(i)} p_{y,k}^{(i)} \tag{3.17}$$

where $w_{k-1}^{(i)}$ are the particle weights from the previous update step and $p_{x,k}^{(i)}$, $p_{y,k}^{(i)}$ are the positions components of particle i after the prediction step at time step k. Since S_k must be computed before the position fix $\boldsymbol{z}_k$ can be obtained, the weights from the previous time step are used.

The size of S_k adapts to the current positional uncertainty, measured by the weighted standard deviation of the particle cloud:

$$\sigma_{x,k} = \sqrt{\sum_{i=1}^{N} w_{k-1}^{(i)} \left(p_{x,k}^{(i)} - \overline{x}_k\right)^2}$$
$$\sigma_{y,k} = \sqrt{\sum_{i=1}^{N} w_{k-1}^{(i)} \left(p_{y,k}^{(i)} - \overline{y}_k\right)^2} \tag{3.18}$$

The search radius is defined as three times the largest standard deviation, following the 3σ convention that covers 99.7% of a Gaussian distribution, ensuring the true vessel position is almost certainly contained within S_k [3]:

$$\sigma_k^{\text{sr}} = \max\left(\sigma_{x,k}, \sigma_{y,k}\right) \tag{3.19}$$

$$r_k = \min\left(r_{\max}, \max\left(3\sigma_k^{\text{sr}}, r_{\min}\right)\right) \tag{3.20}$$

where $r_{\min}$ is determined by the measurement size to ensure S_k always contains at least one full sliding window position for NCC, and $r_{\max} = 500$ m is set to half the maximum acceptable position error of 1000 m as indicated by the Royal Netherlands Navy. The specific value of $r_{\min}$ is given in Section 4.

A square region S_k is extracted from $\hat{R}$, centered at $\overline{\boldsymbol{p}}_k$ and clipped to the raster boundaries. The pixel coordinate of the top-left corner of S_k is recorded as the region origin for use in the position estimation.

3.4.3. NCC

To derive the position fix $\boldsymbol{z}_k$, the preprocessed measurement $\tilde{M}_k$ is slid over the search region S_k and a similarity score $\rho(x, y)$ is computed at every candidate position (x, y) to find the location that best matches the bathymetric pattern of $\tilde{M}_k$. Before computing the score, the measurement and each candidate search window are mean-subtracted:

$$\tilde{M}'_k(x', y') = \tilde{M}_k(x', y') - \overline{\tilde{M}}_k \tag{3.21}$$

$$S'_k(x + x', y + y') = S_k(x + x', y + y') - \overline{S}_k(x, y) \tag{3.22}$$

where $\overline{\tilde{M}}_k = \frac{1}{w \cdot h} \sum_{x'',y''} \tilde{M}_k(x'', y'')$ is the mean of $\tilde{M}_k$ and $\overline{S}_k(x, y) = \frac{1}{w \cdot h} \sum_{x'',y''} S_k(x + x'', y + y'')$ is the local mean of S_k at position (x, y). The similarity score $\rho(x, y)$ ranges from -1 to 1, where 1 indicates a perfect match, and is computed as:

$$\rho(x, y) = \frac{\sum_{x',y'} \tilde{M}'_k(x', y') \cdot S'_k(x+x', y+y')}{\sqrt{\sum_{x',y'} \tilde{M}'_k(x', y')^2 \cdot \sum_{x',y'} S'_k(x+x', y+y')^2}} \tag{3.23}$$

The position with the highest score is selected as the best match:

$$(x^*, y^*) = \arg\max_{(x,y)} \rho(x, y) \tag{3.24}$$

The pixel coordinate (x^*, y^*) is offset by the search region origin, shifted to the measurement center, and converted to world coordinates using the raster spatial resolution and origin, producing the position fix $\boldsymbol{z}_k^{\text{NCC}}$ in $\mathbb{R}^2$. The fix error $\nu_k = \boldsymbol{z}_k - h(\boldsymbol{x}_k)$ is assumed to follow $\nu_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R})$, enabling the Gaussian likelihood model in the weighting (Section 3.3.6). No confidence threshold is applied to the NCC score, as its magnitude is terrain-dependent. Featureless or repetitive seabed can produce high scores at multiple locations, making the peak an unreliable indicator of match quality. Outlier rejection is instead delegated to the combined gating step (Section 3.4.5). If S_k is smaller than $\tilde{M}_k$, NCC cannot be performed and no position is returned, leaving the particle weights unchanged at that time step.

3.4.4. SP+LG

To derive the position fix $\boldsymbol{z}_k$, the preprocessed measurement $\tilde{M}_k$ and search region S_k are prepared for feature matching as follows. Both depth arrays are upscaled by factor $s = 8$, compensating for SuperPoint's internal encoder stride of 8, which downsamples the input before keypoint detection [21]. Normalization is performed by computing percentiles p_2 and p_{98} over the combined depth values of both arrays, ensuring $\tilde{M}_k$ and S_k are mapped to the same intensity range $[0, 1]$, which is required for meaningful feature comparison. Contrast Limited Adaptive Histogram Equalization (CLAHE) [29] is then applied to enhance local contrast. Bathymetric data exhibits subtle depth variation that results in low contrast when normalized, making contrast enhancement necessary for keypoint detection. A clip limit of 4.0 and tile grid size of 4x4 are used to apply sufficiently strong local contrast to small measurements, making subtle variations detectable as keypoints for feature extractor $\mathcal{F}$.

The feature extractor $\mathcal{F}$ is applied to both $\tilde{M}_k$ and S_k, yielding two sets of keypoint locations:

$$\mathcal{K}_M = \left\{\boldsymbol{k}_a^M\right\}_{a=1}^{N_m}, \quad \boldsymbol{k}_a^M \in \mathbb{R}^2 \tag{3.25}$$

$$\mathcal{K}_R = \left\{\boldsymbol{k}_b^R\right\}_{b=1}^{N_r}, \quad \boldsymbol{k}_b^R \in \mathbb{R}^2 \tag{3.26}$$

where $\boldsymbol{k}_a^M$ and $\boldsymbol{k}_b^R$ are coordinates in scaled pixel space at which a distinctive terrain feature was detected, and N_m and N_r are the number of keypoints in $\tilde{M}_k$ and S_k respectively.

The matcher $\mathcal{L}$ (LightGlue [22]) pairs keypoints across $\mathcal{K}_M$ and $\mathcal{K}_R$, returning a set of correspondences with per-match confidence scores:

$$\mathcal{M}_k = \mathcal{L}(\mathcal{K}_M, \mathcal{K}_R) = \{(a_l, b_l, c_l)\}_{l=1}^{N_c}, \quad c_l \in [0,1] \tag{3.27}$$

where a_l and b_l are the indices into $\mathcal{K}_M$ and $\mathcal{K}_R$ of the l-th matched pair, c_l is the associated confidence score, and N_c is the number of matches. The matched coordinate pairs are:

$$\boldsymbol{m}_l = \boldsymbol{k}_{a_l}^M \in \mathbb{R}^2, \quad \boldsymbol{r}_l = \boldsymbol{k}_{b_l}^R \in \mathbb{R}^2 \tag{3.28}$$

For each correspondence, the displacement vector is:

$$\boldsymbol{\delta}_l = \boldsymbol{r}_l - \boldsymbol{m}_l \in \mathbb{R}^2 \tag{3.29}$$

where $\boldsymbol{r}_l$ and $\boldsymbol{m}_l$ are the matched keypoint coordinates in the upscaled search region and measurement arrays respectively. To ensure uniform keypoint density across both arrays, the keypoint limits are set proportionally to their areas, with $N_m^{\max} = 128$ for $\tilde{M}_k$ and $N_r^{\max} = 2048$ for S_k. The shift estimate is the component-wise median over all N_c matches:

$$\hat{\boldsymbol{\delta}}_k = \begin{pmatrix} \text{median}_l\left(\delta_l^x\right) \\ \text{median}_l\left(\delta_l^y\right) \end{pmatrix} \tag{3.30}$$

The median is preferred over the mean since a single outlier correspondence cannot dominate the estimate.

The estimated position of the measurement center in the reference map is then:

$$\boldsymbol{p}_k = \frac{1}{s}\left(\boldsymbol{c} + \hat{\boldsymbol{\delta}}_k\right) + \boldsymbol{o}_k \tag{3.31}$$

where $\boldsymbol{c} = \frac{s}{2}(W_m, H_m)^\top$ is the center of the upscaled measurement, with W_m and H_m the width and height of $\tilde{M}_k$ in pixels, s is the upscaling factor, and $\boldsymbol{o}_k = (c_0, r_0)^\top$ is the pixel origin of the search region in the reference map $\hat{R}$. The pixel coordinate $\boldsymbol{p}_k$ is then converted to world coordinates using the raster spatial resolution and origin, producing the position fix $\boldsymbol{z}_k^{\text{SP+LG}}$ in $\mathbb{R}^2$.The LightGlue match threshold is set to 0.05 to maximize the number of correspondences, as outlier robustness is delegated to the median estimator rather than the confidence threshold. If fewer than $N_{\min} = 3$ matches are found, the minimum number for which the median can suppress one outlier, the fix is rejected and no position is returned and the particle weights remain unchanged at that time step.

3.4.5. Combined

At each time step k, after the prediction in Section 3.3.5, the particle filter maintains a prior distribution over vessel position. The prior mean and standard deviation are:

$$\mu_k = \sum_{i=1}^{N} w_k^{(i)} \boldsymbol{x}_k^{(i)} \tag{3.32}$$

$$\sigma_k^{\text{prior}} = \sqrt{\sum_{i=1}^{N} w_k^{(i)} \left\| \boldsymbol{x}_k^{(i)} - \mu_k \right\|^2} \tag{3.33}$$

where $w_k^{(i)}$ are the particle weights and $\boldsymbol{x}_k^{(i)} = \left(p_{x,k}^{(i)}, p_{y,k}^{(i)}\right)^\top$ are the particle positions after prediction. NCC is run first, producing a position fix $\boldsymbol{z}_k^{\text{NCC}}$. The Euclidean distance from the fix to the prior mean is computed:

$$d_k = \left\| \boldsymbol{z}_k^{\text{NCC}} - \mu_k \right\| \tag{3.34}$$

The fix is accepted only if it lies within the rejection threshold:

$$\tau_k = \max\left(\gamma \cdot \sigma_k^{\text{prior}}, \tau_{\min}\right) \tag{3.35}$$

where γ is the sigma factor and $\tau_{\min}$ is a minimum floor. The decision rule is:

$$\boldsymbol{z}_k = \begin{cases} \boldsymbol{z}_k^{\text{NCC}} & \text{if } d_k < \tau_k \\ \boldsymbol{z}_k^{\text{SP+LG}} & \text{if } d_k \geq \tau_k \text{ and SP+LG valid} \\ \text{no fix} & \text{otherwise} \end{cases} \tag{3.36}$$

The threshold τ_k adapts with the particle cloud. When the particle filter is well-converged and σ_k^{prior} is small, the gate tightens and becomes more selective. When the filter is uncertain and σ_k^{prior} is large, the gate relaxes to avoid rejecting valid fixes that lie further from the prior mean. Here, $\tau_{\min} = 500$ m ensures the gate remains permissive during early convergence and aligns with the maximum acceptable position error (Section 3.4.2), and $\gamma = 1$ applies a conservative 1σ rejection gate, accepting only NCC fixes that lie close to the prior mean. This is motivated by NCC's tendency to produce occasional large false positives, as observed in the experiments in Section 4.

4. Experiments

This chapter evaluates the performance of SAND-E for seabed-aided maritime navigation in GNSS-denied environments. The experiments assess whether position estimates derived from seabed fingerprinting can correct DR drift and support accurate, robust, and computationally efficient maritime navigation. An overview of the experimental design is shown in Figure 4.1.

First, the experimental setup is described in Section 4.1. The study areas are introduced in Section 4.1.1, followed by the simulation model in Section 4.1.2, and the evaluation metrics in Section 4.1.3. Next, the parameter selection is presented in Section 4.2, where the measurement width, measurement noise covariance, and update interval are determined. Finally, the navigation experiments are discussed in Section 4.3, including the Monte Carlo analysis, representative run, outdated-

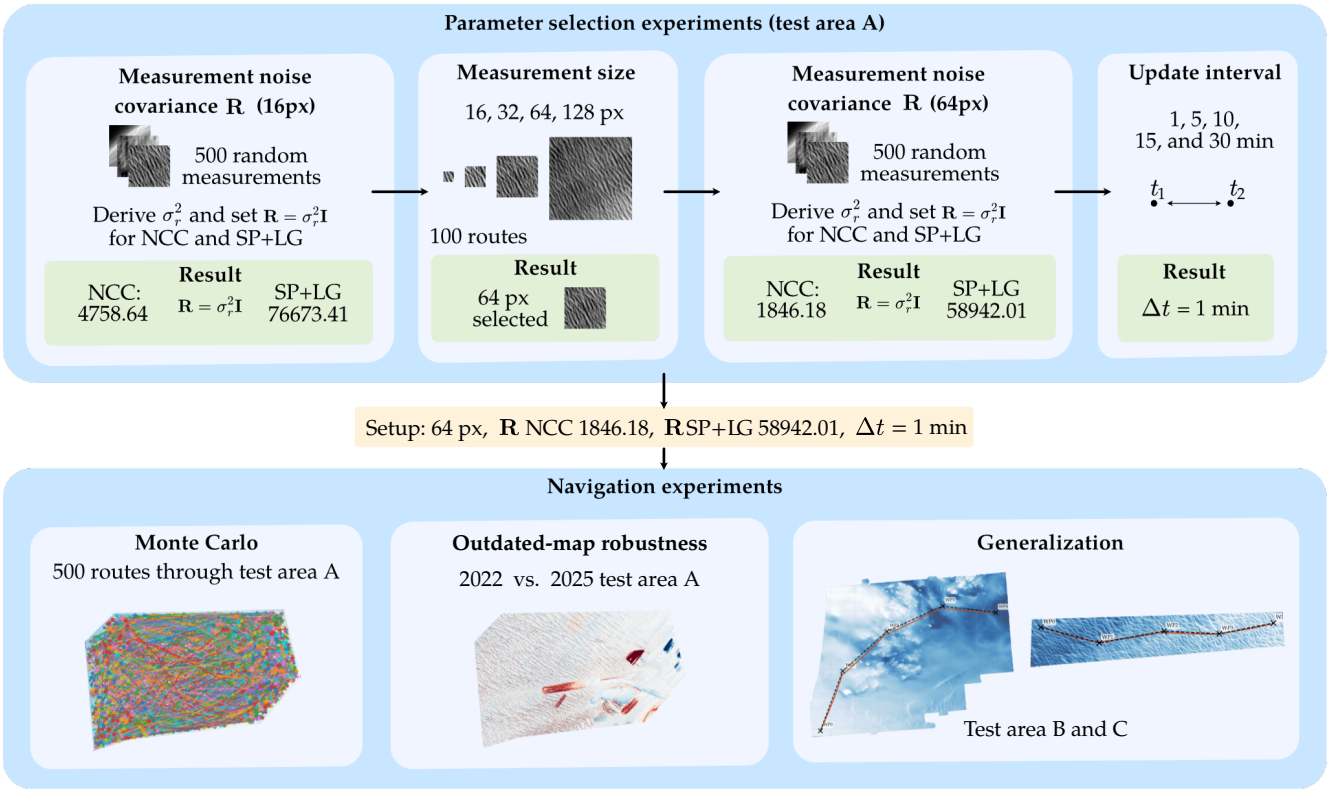


Figure 4.1: Experimental design used to evaluate SAND-E. Parameter selection experiments determine the measurement noise covariance, measurement width, and update interval, after which the selected configuration is used for the Monte Carlo, outdated-map robustness, and generalization experiments.

map robustness test, and generalization to additional test areas.

4.1. Experimental Setup

This section describes the experimental setup used to evaluate SAND-E. It introduces the bathymetric test areas (Section 4.1.1), the vessel simulation model (Section 4.1.2), and the evaluation metrics (Section 4.1.3) used to assess both fix-level positioning accuracy and trajectory-level navigation performance.

Random measurements, navigation measurements, and generated trajectories are used throughout the parameter-selection and navigation experiments. The generation procedure is described in Appendix B.

4.1.1. Test Areas

Three bathymetric test areas are used in the experiments. An overview of the test areas is shown in Figure 4.2. Area A covers the Eurogeul approach to the port of Rotterdam in the North Sea and contains diverse seabed morphology, including sand ripples and dredged channels. It serves as the primary test area on which SAND-E is developed and parametrized, as its morphological variety makes it a challenging baseline for framework design. Reference from both 2022 and 2025 enable the outdated-map robustness test in Section 4.3.3. It should be noted that except for the outdated map experiment, all measurements are extracted directly from the reference map, ensuring a perfect match exists. Area B is a second North Sea location in close proximity to Area A but with more uniform seabed morphology, providing a more challenging test case within the same geographic setting. Area C is located in the Atlantic Ocean near the north coast of Ireland and represents a structurally distinct seabed, characterized by rocky terrain and a larger depth range of 70 m, compared to 14 m for Area A or 5 m for Area B, allowing generalization to be assessed beyond North Sea bathymetry.

4.1.2. Simulation Model

A vessel navigates a five-waypoint route at 10 knots (5.1 m/s). An unknown SSE current of 0.58 knots (0.3 m/s), representative of typical North Sea tidal conditions [30], is applied to the true trajectory but is not known to the particle filter, causing DR error to accumulate. Particles are initialized at the known starting position with no initial uncertainty (Section 3.3.4). To ensure that results are not specific to a single path, the simulation is repeated 500 times over independently generated routes, with heading perturbations introducing minor deviations that simulate realistic vessel dynamics. Simulation parameters are fixed across runs and listed in Table 2, while additional parameters used in the navigation experiments in Section 4.3 are listed in Table D.1. Each positioning method is configured

based on empirical calibration rather than identical parameter settings, enabling a fair assessment of practical performance. The particle count is set to $N = 5000$, as the four-dimensional state space requires thousands of particles for adequate posterior approximation [8].

4.1.3. Evaluation Metrics

Performance is evaluated at two levels: fix-level accuracy and trajectory-level accuracy.

Fix-level accuracy. This characterizes the positioning methods in isolation. The positioning error $e_k = \left\| \hat{\boldsymbol{p}}_k - \boldsymbol{p}_k^{\text{true}} \right\|_2$ measures the Euclidean distance between each position fix and the true vessel position. RMSE:

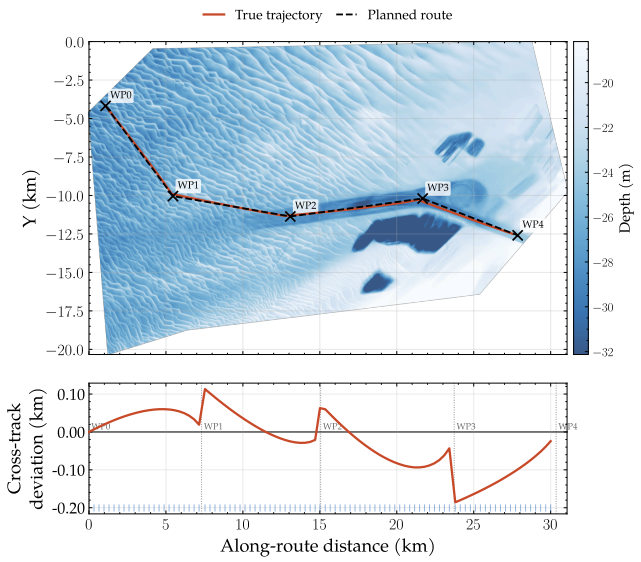
$$\text{RMSE} = \sqrt{\frac{1}{K} \sum_{k=1}^{K} \left\| \hat{\boldsymbol{p}}_k - \boldsymbol{p}_k^{\text{true}} \right\|_2^2} \tag{4.1}$$

where K is the total number of time steps, summarizes the overall positioning accuracy. The mean, median, and standard deviation are reported alongside RMSE. The median is included as it is robust to occasional gross errors. The measurement noise covariance $\sigma_r^2 = \left(\frac{\text{RMSE}}{\sqrt{2}}\right)^2$ is derived from the RMSE and used to parameterize $\boldsymbol{R}$ (Section 3.3.3) in the particle filter. The invalid match rate reports the fraction of time steps where no valid fix was returned, and the fix rate the fraction of fixes within a defined error threshold.

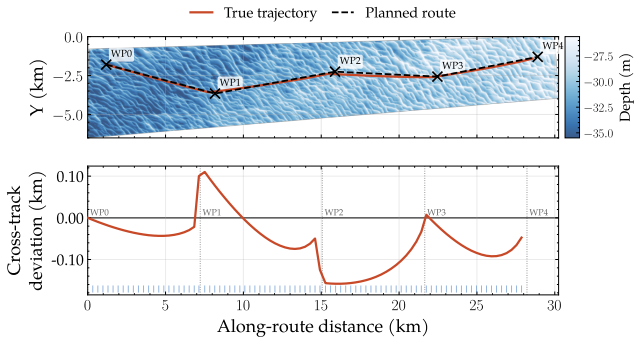
Trajectory-level accuracy. This characterizes the full particle filter. The RMSE and final position error of the particle filter estimate over a complete run are reported, alongside the fix rate, the fraction of time steps where a valid fix was obtained. Runtime is reported as median and standard deviation across fixes to assess computational feasibility. All runtime measurements were obtained on a Dell Pro Max 16 Plus MB16250 laptop, with the full system specifications listed in Table C.1.

4.2. Parameter Selection

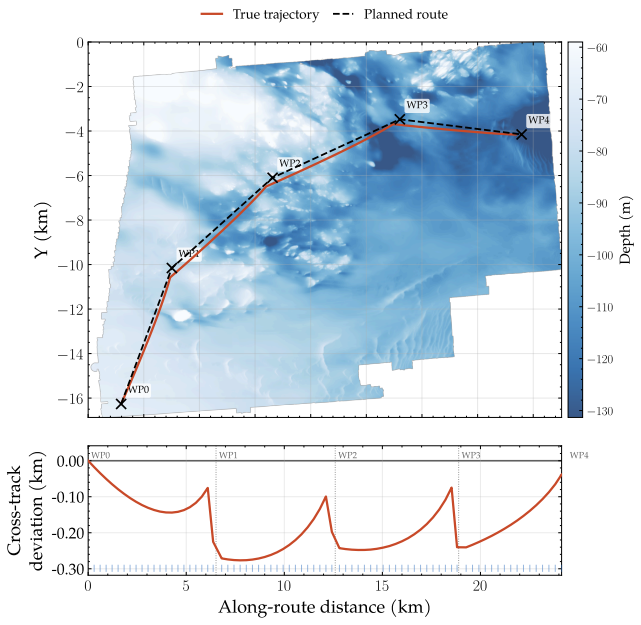
Before the main navigation experiments, three parameters are determined sequentially: measurement width, measurement noise covariance $\boldsymbol{R}$, and update interval Δt. Each parameter is fixed before the next is selected. For this parameter selection, two generated datasets are used: 500 standalone random measurements from test area A to estimate a conservative measurement noise covariance for the measurement width experiment in Section 4.2.1, and 100 independently generated routes through test area A for the route-based experiments. These routes share the same starting point but use different seeds for heading, particle, and process noise. The same route-based setup is used to evaluate measurement width and determine the update interval Δt. The combined method is not included in the parameter-selection experiments and is evaluated only in the final navigation experiments.



(a) Test area A, North Sea

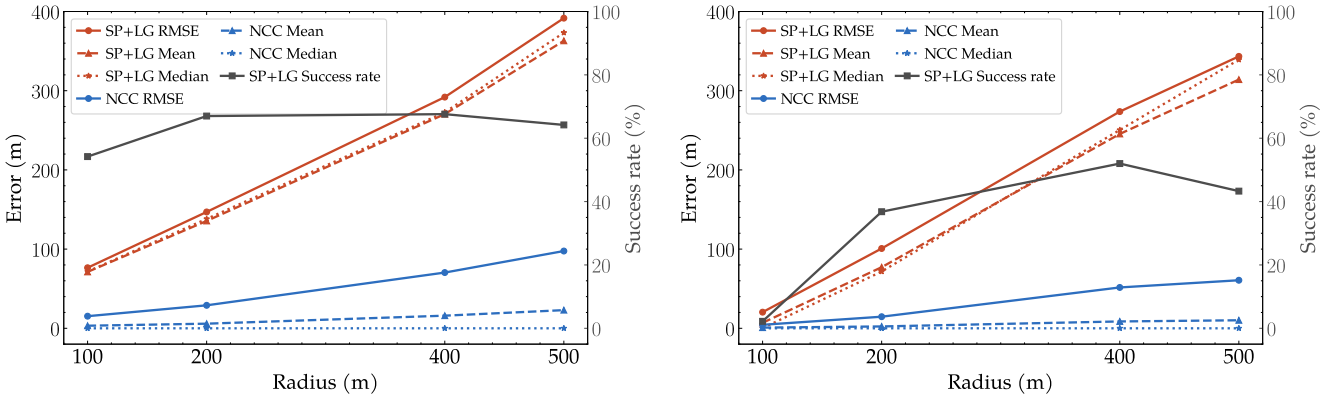


(b) Test area B, North Sea



(c) Test area C, Atlantic Ocean

Figure 4.2: Planned routes and true trajectories overlaid on bathymetric reference maps for the three test areas, with cross-track deviation shown below each map.



(a) NCC (b) SP+LG

Figure 4.3: Positioning accuracy and fix rate as a function of search radius for NCC and SP+LG, evaluated on 500 random measurements from test area A for a measurement width of 16 px (left) and 64 px (right).

4.2.1. Measurement Width

Measurement width controls how much seabed information is available for matching and therefore affects navigation accuracy and robustness. SAND-E is evaluated using measurement widths of 16, 32, 64, and 128 px on 100 randomly generated routes through test area A to determine the minimum information required for reliable navigation. Each route uses a different random seed for heading, particle, and process noise, producing a distribution of outcomes. A conservative measurement noise standard deviation σ_r^2 is derived from the 16 px positioning accuracy tests in Table D.2, where positioning accuracy is evaluated as a function of search radius under ideal conditions. The lowest-information case of 16 px is used solely to derive a conservative upper bound on positioning uncertainty ($\sigma_r^2 = 4758.64\text{m}^2$ for NCC and $\sigma_r^2 = 76673.41\text{m}^2$ at radius $r_{\max} = 500$ m), ensuring the particle filter does not over-trust position fixes in the measurement width experiment. The 16 px results are not discussed further. Positioning accuracy for the selected measurement width is evaluated in Section 4.2.2.

Method	Metric	Measurement width (px)			
		16	32	64	128
NCC	RMSE (m)	97.56	67.69	60.76	63.25
	Median (m)	0.0	0.0	0.0	0.0
	Fix rate (%)	100.0	100.0	100.0	100.0
SP+LG	RMSE (m)	391.60	350.33	343.34	357.83
	Median (m)	373.24	333.95	339.14	337.97
	Fix rate (%)	35.8	37.6	56.6	62.2

Table 1: Navigation accuracy across measurement widths for NCC and SP+LG, evaluated on 100 routes through test area A.

Table 1 shows that NCC is robust across all tested measurement widths, with a median of 0 m and a 100% fix rate. Its RMSE decreases from 97.56 m at 16 px to 60.76 m at 64 px, after which no further improvement is observed. SP+LG benefits from larger measurements, with fix rate increasing from 35.8% for 16 px to 62.2% for 128 px, but its median error remains above 300 m for all measurement widths. This is also visible in the representative run in Section 4.3.2, where NCC remains consistently below the 500 m threshold for all measurement widths, while SP+LG shows substantially higher and more variable errors, reaching approximately 1000 m by the end of the route.

A measurement width of 64 px is selected as NCC reaches its lowest RMSE and SP+LG its best balance between RMSE and fix rate at this width, with no further improvement at 128 px. It is therefore used to compare NCC, SP+LG, and the combined approach in Section 4.3.

4.2.2. Measurement Noise Covariance $\boldsymbol{R}$

To determine the measurement noise covariance for the selected 64 px measurement width, positioning accuracy is evaluated as a function of search radius on 500 random measurements from test area A, with results shown in Figure 4.3 and Table D.3.

Figure 4.3 shows that a measurement width of 64 px improves positioning accuracy for both methods compared to 16 px, with NCC RMSE decreasing substantially while the median remains at 0 m, and SP+LG improving most at small search radii where ambiguity is low. NCC achieves a median of 0 m at all radii while RMSE grows with radius due to occasional false positives. SP+LG is accurate at small search radii but invalid matches rise sharply beyond 200 m as larger search regions increase match ambiguity. The resulting measurement noise covariances $\boldsymbol{R} = \sigma_r^2 \boldsymbol{I}$, with $\sigma_r^2 = 1846.16\text{m}^2$ for NCC and $\sigma_r^2 = 58942.01\text{m}^2$ for SP+LG at $r_{\max} = 500$ m, are used in Section 4.2.3 to determine the optimal update interval Δt.

4.2.3. Update Interval Δt

The derived values for σ_r^2 for NCC and SP+LG are used to determine the optimal update interval for each positioning method.

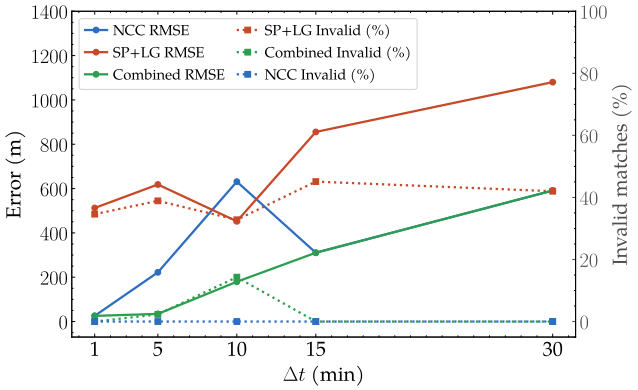


Figure 4.4: RMSE and invalid match rate as a function of update interval Δt for NCC, SP+LG, and Combined, evaluated on 100 routes through test area A at a measurement width of 64 px.

The results are shown in Figure 4.4 and Table D.4. NCC performs best at $\Delta t = 1$ min, achieving an RMSE of 25.5 m and standard deviation of 0.7, with near-zero invalid matches across all update intervals. Performance degrades sharply beyond $\Delta t = 1$ min as drift exceeds the 500 m search region. At $\Delta t = 15$ the mean RMSE appears lower than at $\Delta t = 10$ min (631.2 m), but this is misleading because $\Delta t = 10$ min partial convergence in some runs inflates the mean, whereas $\Delta t = 15$ min all runs fail uniformly.

SP+LG performs considerably worse across all intervals, with RMSE exceeding 500 m even at $\Delta t = 1$ min and invalid match rates at 33-45%. The combined method mirrors NCC at $\Delta t = 1$ min, confirming that NCC dominates at short intervals. At $\Delta t = 10$ min, its RMSE increases to 179.5 m, outperforming NCC alone (631.2 m), though the invalid match rises to 14.3%, reflecting cases where the NCC position fix is rejected and the SP+LG fallback also fails.

Due to the consistently low RMSE and near-zero invalid match rate, $\Delta t = 1$ min is selected for all navigation experiments in Section 4.3.

4.3. Navigation Experiments

With the parameters established in Section 4.2, the full simulation configuration is summarized in Table 2.

Symbol	Value	Unit
Δt	1	min
v	10	knots
N	5000	–
σ_p	$0.01 \cdot v \cdot \Delta t$	m
σ_r^2	NCC: 1848.16, SP+LG: 58942.01	m^2

Table 2: Navigation experiments configuration.

This section evaluates SAND-E using the parameter configuration in Table 2. The experiments assess statistical performance over 500 Monte Carlo runs, robustness to an outdated reference map, and generalization across geographically distinct test areas.

4.3.1. Monte Carlo Analysis

The Monte Carlo analysis evaluates each positioning method over 500 independently generated five-waypoint routes of maximum 6.5 km. At each time step, a measurement is generated from the true vessel position and heading and passed to the positioning method. The 500 runs provide a distribution of outcomes across varied routes rather than relying on a single representative trajectory.

From Table 3, NCC achieves the strongest standalone performance, with an RMSE of 92.1 m, a final error of 115.3 m, and a 100% fix rate. As shown in Figure 4.5, the interquartile range for NCC remains below 100 m, although the long upper tail indicates occasional large errors. SP+LG performs considerably worse, with an RMSE of 308.2 m, a final error of 387.9 m, and a fix rate of only 47.5%, with a median RMSE above 200 m and high variance. The combined method closely matches NCC, with the same RMSE of 92.1 m and an overlapping error distribution, confirming that NCC provides reliable fixes throughout the runs and leaves the SP+LG fallback largely unused.

Method	RMSE (m)	Final error (m)	< 500 m (%)	Fix (%)
NCC	92.1	115.3	92.6	100.0
SP+LG	308.2	387.9	67.8	47.5
Combined	92.1	115.8	92.4	100.0

Table 3: Performance summary for NCC, SP+LG, and Combined over 500 Monte Carlo runs through test area A.

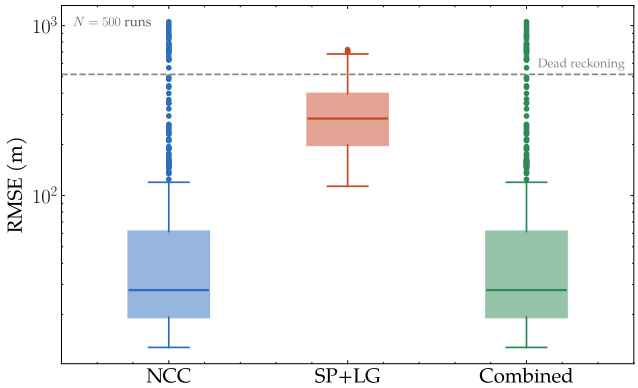
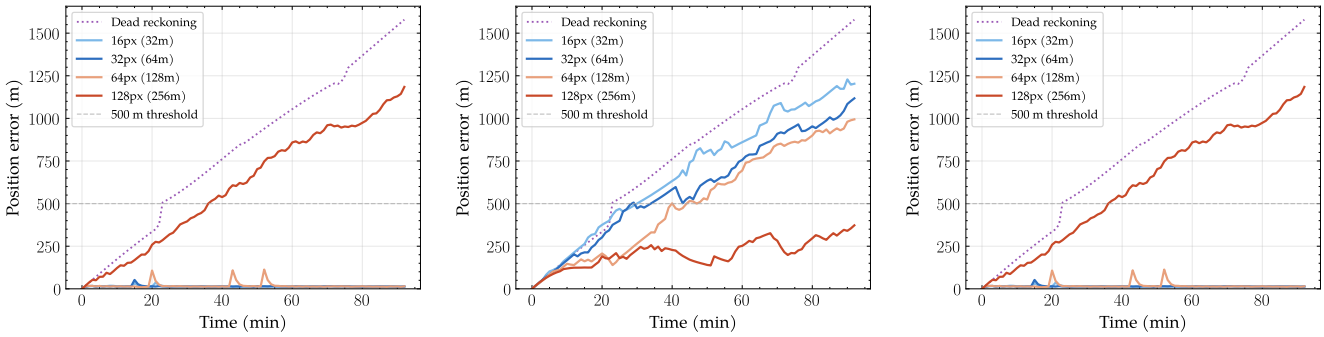


Figure 4.5: Distribution of RMSE across 500 Monte Carlo runs for NCC, SP+LG, and Combined. The dashed line indicates the DR baseline.

Table 4 reports the runtime per position fix for each method. NCC runs at a median of 0.5 ms per fix, while SP+LG is roughly 180 times slower at 91.0 ms. The combined method achieves 0.4 ms, slightly faster than NCC alone.



(a) NCC (b) SP+LG (c) Combined

Figure 4.7: Position error over time for test area A: NCC (left), SP+LG (middle), and Combined (right) across measurement widths of 16, 32, 64, and 128 pixels.

Method	Median (ms)	Std (ms)
NCC	0.5	0.1
SP+LG	91.0	22.6
Combined	0.4	0.1

Table 4: Runtime per position fix for each positioning method.

4.3.2. Representative Run

To visually illustrate the results from Section 4.3.1, a single representative run from the test area A is shown in Figure 4.6. NCC successfully bounds drift throughout the route, keeping position error below 100 m with occasional spikes (due to false positives) that are quickly corrected. SP+LG fails as a standalone method, with error growing nearly linearly, crossing 500 m after approximately 10 minutes and reaching 1000 m by the end. The combined method is indistinguishable from NCC, with SP+LG never invoked as a fallback. DR reaches a final error of 1600 m, demonstrating that NCC reduces final position error by over a factor of 100 compared to unaided navigation.

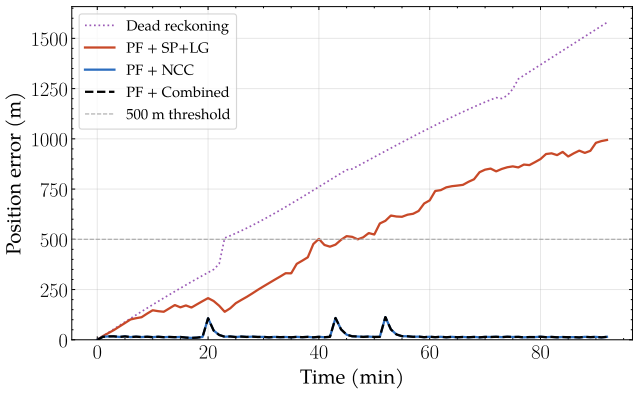


Figure 4.6: Position error over time for NCC, SP+LG, and Combined on test area A.

Figure 4.7 shows the effect of measurement width on positioning error for this specific run, consistent with the findings in Section 4.2.1.

4.3.3. Outdated Map Robustness

To evaluate robustness to map age, SAND-E is tested on test area A using the 2022 reference map while the true vessel trajectory is simulated using 2025 bathymetry, introducing a three-year temporal mismatch between the reference map and near real-time measurements. Figure 4.8 shows the bathymetric depth change between 2022 and 2025 for test area A.

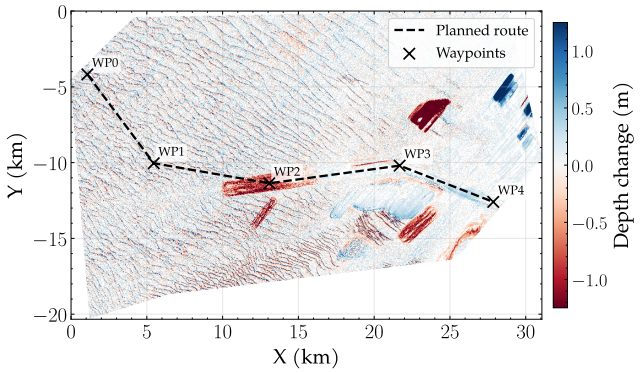


Figure 4.8: Bathymetric depth change between 2022 and 2025 for test area A, with the planned route overlaid. Blue areas indicate decreasing water depth (sediment accretion) and red areas indicate increasing depth (erosion or dredging).

The mean depth change across the survey area is 0.02 m, indicating overall seabed stability,though zones of significant change are visible along the planned route in the dredged regions. In these areas, depth values in the reference map no longer match near real-time measurements, making matching more difficult.

SAND-E is tested on this scenario and the resulting position error over time is shown in Figure 4.9.

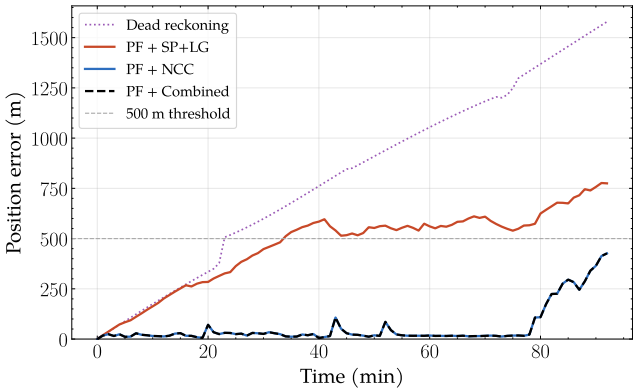


Figure 4.9: Position error over time for NCC, SP+LG, and Combined on test area A with a three-year-old reference map.

With a three-year-old reference map, NCC remains viable for most of the route, with errors below 100 m until minute 80, where degradation coincides with the zone of significant seabed change between 2022 and 2025 identified in Figure 4.8. SP+LG degrades far more severely, crossing the 500 m threshold at minute 20, confirming greater sensitivity to seabed change than NCC. The combined method tracks NCC. Both NCC and combined remain far below the DR baseline, demonstrating that seabed-aided maritime navigation provides significant value even with a three-year-old outdated reference map.

4.3.4. Generalization

SAND-E is further evaluated on two additional test areas shown in Section 4.1 to assess generalization beyond area A. Area B is the second North Sea location with more uniform seabed morphology, while area C represents a structurally distinct seabed in the Atlantic Ocean. Each method is evaluated on a single simulated route through each area using the parameters established in Section 4.2. Results for area B are a shown in Figure 4.10.

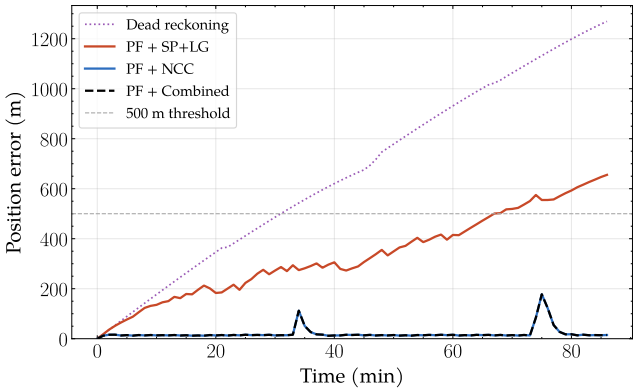


Figure 4.10: Position error over time for NCC, SP+LG, and Combined on test area B, North Sea.

Area B shows NCC performing well, with position error below 200 m throughout and only two brief correctable spikes. SP+LG again fails, with an error growing near-linearly, crossing 500 m around minute 40 and reaching 650 m by the end. The combined method tracks NCC exactly, with SP+LG never triggered as fallback. DR diverges to approximately 1250 m.

Results for area C are shown in Figure 4.11.

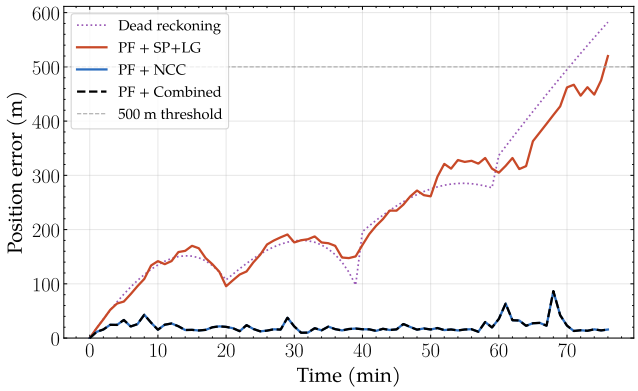


Figure 4.11: Position error over time for NCC, SP+LG, and Combined on test area C, Atlantic Ocean.

NCC remains robust, with error below 100 m and only small correctable spikes, demonstrating generalization beyond North Sea bathymetry. SP+LG fails again, with error growing steadily and crossing 500 m near the end of the route. The combined method once more tracks NCC exactly, with SP+LG unused throughout.

5. Discussion

5.1. NCC Performance and Limitations

NCC demonstrates robust positioning across all experiments, with a consistently zero median error. When terrain is distinctive, NCC finds the exact match. When it is not, false positives are produced that are indistinguishable from correct matches as confidence scores are equally high for both, making detection impossible without external validation. This structural limitation cannot be resolved by increasing measurement width or applying confidence thresholds, consistent with correlation-based TAN literature [1, 3] and image matching literature more broadly [31].

Occasional error spikes visible in the NCC and combined results are consistent with this behavior, occurring when the vessel traverses repetitive terrain where multiple candidate positions produce equally high correlation scores, causing a temporary false positive.

The particle filter mitigates this effectively. Occasional false positives temporarily shift particle weights, but subsequent correct fixes restore the distribution. This resilience to outlier fixes is a key advantage over deterministic methods where a single false positive permanently displaces the estimate.

NCC's robustness to a three-year outdated map follows from its matching on overall depth patterns, which change slowly. Degradation coincides specifically with dredged regions where depth changes of up to 1 m were observed, far exceeding the 0.02 m mean change. Since bathymetric surveys are expensive and infrequent, this map-age robustness is operationally significant.

5.2. SP+LG Performance and Domain Gap

Based on literature, SP+LG appeared promising due to its fast computation and robustness under challenging conditions such as low texture [31]. However, SP+LG fails consistently as a standalone method across all experiments, test areas, update intervals, and measurement widths. The root cause appears to be a domain gap. SuperPoint is trained on images containing corners, edges, and textured regions, none of which are abundant in bathymetric terrain. Gradual depth variation and repetitive patterns produce insufficient and low-quality keypoints, causing SP+LG to abstain when fewer than 3 correspondences are found and to return inaccurate position fixes when they are. Together, these factors make SP+LG unsuitable as a standalone method.

SP+LG is capable of accurate matching when spatial ambiguity is constrained, as demonstrated by its accuracy at small search radii. As search regions grow, ambiguity increases and performance degrades.The update interval has almost no effect, confirming the failure is terrain-related rather than operational.

SP+LG runs at 91 ms per position fix, approximately twice the 44 ms reported for SuperPoint together with LightGlue on standard image pairs [32]. This is caused by the 8x upscaling and CLAHE contrast enhancement applied to both arrays before feature extraction, increasing input size and processing time.

5.3. Combined Method Performance

The combined method matches NCC exactly at $\Delta t = 1$ min, with SP+LG never invoked. The gate adds value only at longer update intervals, when larger search regions cause more NCC outliers that the gate correctly rejects, allowing SP+LG to contribute when needed. The combined approach is therefore preferable in deployment as this ensures robustness for harder operating conditions. Gate parameters $\gamma = 1$ and $\tau_{\min} = 500$ m were set empirically. Optimization could allow SP+LG to contribute more meaningfully as a fallback even at shorter update intervals.

5.4. Limitations and Operational Considerations

All experiments use generated data and trajectories, meaning a perfect match always exists. Results therefore reflect best-case performance and do not capture the full complexity of real operational conditions. In practice, imperfect motion correction, additional measurement noise, and partial MBES coverage would degrade matching quality. Speed is assumed constant and current uniform, while real vessels face varying speeds and tidal currents. No INS is assumed for position prediction, representing the most constrained scenario. Generalization is assessed on three shallow coastal areas at a reference map resolution of 2 m. Coarser resolution maps, which are more commonly available in practice, would reduce recoverable terrain detail and likely degrade positioning accuracy. Deep ocean or polar regions remain untested. Routes follow fixed waypoints, while real operations involve complex maneuvers or sensor failures, which are not considered here. Finally, map availability remains a bottleneck as surveys are costly and often classified, though reference maps only need to be surveyed when GNSS is available after which real-time MBES measurements are matched in relative terms. Autonomous Underwater Vehicle (AUV)-based mapping could help [33], and broader open bathymetric data would enable civilian application, beyond the military domain.

6. Conclusions

SAND-E is designed around a particle filter that continuously estimates vessel position by fusing near real-time seabed-derived position fixes with a motion model. At each time step, the weighted mean of the particle distribution defines the center of the search region, and the weighted standard deviation defines its size, ensuring the search region covers the area where the vessel is most likely located while remaining computationally efficient. The framework requires only a pre-existing bathymetric reference map and an onboard MBES, with no dependence on external signals or infrastructure, making it suitable for military surface vessels operating in GNSS-denied environments. Although SAND-E is developed and evaluated for military surface vessels, the proposed seabed-fingerprinting approach is also applicable to underwater vehicles, provided that onboard bathymetric measurements, heading information, and a suitable reference map are available.

Within the particle filter framework, seabed fingerprinting is formulated as an image matching problem in which preprocessed MBES measurements are

matched against a search region extracted from the reference map. After orientation-alignment, matching reduces to a pure translation search. Three positioning models are evaluated: (1) NCC slides the measurement over the search region and selects the highest correlation peak, (2) SP+LG detects and matches keypoints across both arrays and estimates the translation from the median correspondence displacement, and (3) a combined approach gates NCC fixes using the particle filter prior, falling back to SP+LG when fixes are rejected. Each position fix is converted to world coordinates and fused into the particle filter to correct accumulated drift

All navigation experiments are conducted at $\Delta t = 1$ min. Contrary to expectations, the classical NCC baseline outperforms the learned SP+LG method on every metric across all experiments.

SAND-E with NCC achieves accurate navigation with a 100% fix rate, and 92.6% of 500 Monte Carlo routes ending within the 500 m threshold. The combined method matches NCC closely at 92.4%. SP+LG achieves an RMSE of 308.2 m with a fix rate of only 47.5% and 67.8% of routes within the threshold, making it unsuitable as a standalone positioning method. SP+LG performs well on standard image matching benchmarks but fails on bathymetric data, showing that strong general performance does not guarantee good performance on domain-specific data. For 500 randomly generated measurements, NCC achieves an RMSE of 60.67 m, a median error of 0 m, and a 100% valid-fix rate. At the trajectory level, NCC and the combined method successfully correct accumulated drift, whereas SP+LG does not. SP+LG achieves an RMSE of 343.34 m and a median error of 339.14 m.

In terms of robustness, NCC and the combined method remain viable with a three-year outdated reference map, with position error below 100 m until degradation coincides with the zone of significant seabed change. SP+LG degrades far more severely, confirming greater sensitivity to map age. SAND-E generalizes across three geographically distinct test areas, with NCC maintaining position error below 200 m throughout, while SP+LG grows near-linearly and is never invoked as a fallback.

In terms of computational efficiency, NCC runs at a median of 0.5 ms per fix, making it suitable for real-time navigation. The combined method runs at 0.4 ms. SP+LG is slower at 91.0 ms per position fix and a high standard deviation.

Future work.
Several directions remain for future work. A real-time continuous update strategy, producing a fix from every available measurement rather than at fixed intervals, would further reduce drift beyond the $\Delta t = 1$ min results. An elliptical search region aligned with the particle distribution could improve computational efficiency over the current square region [3]. NCC could be further improved using Fast Fourier Transform (FFT) correlation, which would be beneficial for larger search regions.

SP+LG's failure is terrain-related rather than operational, suggesting that fine-tuning SuperPoint or a detector-free method such as LoFTR on bathymetric data could close the performance gap [2, 16, 31].

INS integration could complement the particle filter and significantly improve performance between fixes. In addition, initialization without a known position would require distributing particles across the full reference map, which is computationally infeasible at the current particle count and is left for future work.

Finally, the current framework assumes the measurement and reference map share the same spatial resolution. Extending this to support different resolutions would allow the framework to work with reference maps from different sources, broadening its applicability.

References

[1] T. Ma, S. Ding, Y. Li, and J. Fan, "A review of terrain aided navigation for underwater vehicles," vol. 281. Elsevier Ltd, 2023. doi: 10.1016/j.oceaneng.2023.114779.

[2] J. Long, J. Zhao, X. Zhao, and C. Jin, "Underwater terrain matching method based on multibeam bathymetric point cloud descriptors," *International Journal of Digital Earth*, vol. 17, no. 1, 2024, doi: 10.1080/17538947.2024.2434655.

[3] G. Fan *et al.*, "Survey of terrain-aided navigation methods for underwater vehicles," vol. 11. Institute of Electrical, Electronics Engineers Inc., pp. 47510–47526, 2023. doi: 10.1109/ACCESS.2023.3275078.

[4] D. Wang, L. Liu, Y. Ben, P. Dai, and J. Wang, "Seabed terrain-aided navigation algorithm based on combining artificial bee colony and particle swarm optimization," *Applied Sciences*, vol. 13, no. 2, p. 1166, 2023, doi: 10.3390/app13021166.

[5] T. Haydon, A. Huang, and T. E. Humphreys, "Analysis of the trusted inertial terrain-aided navigation measurement function," *Navigation, Journal of the Institute of Navigation*, vol. 72, no. 3, 2025, doi: 10.33012/navi.707.

[6] P. Ding and X. Cheng, "A new contour-based combined matching algorithm for underwater terrain-aided strapdown inertial navigation system," *Measurement: Journal of the International Measurement Confederation*, vol. 202, 2022, doi: 10.1016/j.measurement.2022.111870.

[7] J. Melo and A. Matos, "Survey on advances on terrain based navigation for autonomous underwater vehicles," vol. 139. Elsevier Ltd, pp. 250–264, 2017. doi: 10.1016/j.oceaneng.2017.04.047.

[8] S. R. Dhayalkar, "Particle filter made simple: a step-by-step beginner-friendly guide," 2025, [Online]. Available: http://arxiv.org/abs/2511.01281

[9] E. Raggi, "Localization of a drifting underwater vehicle using a terrain-based particle filter," technical report. [Online]. Available: https://digitalcommons.uri.edu/theses

[10] K. Hong, S. Kim, J. Park, and H. Bang, "Particle filter approach to vision-based navigation with aerial image segmentation," *Journal of Aerospace Information Systems*, vol. 18, no. 12, pp. 964–972, 2021, doi: 10.2514/1.I010957.

[11] W. Zhang, D. Ma, J. Gao, T. Ma, and Y. Ben, "A robust seabed terrain contour aid navigation method facing the smooth terrain," in *2022 12th International Conference on CYBER Technology in Automation, Control, and Intelligent Systems, CYBER 2022*, Institute of Electrical, Electronics Engineers Inc., 2022, pp. 1166–1170. doi: 10.1109/CYBER55403.2022.9907123.

[12] K. B. Ånonsen, O. Hallingstad, O. K. Hagen, and M. Mandt, "Terrain aided AUV navigation-a comparison of the point mass filter and terrain contour matching algorithms," technical report, 2005. [Online]. Available: https://api.semanticscholar.org/CorpusID:37780706

[13] P. Li *et al.*, "Underwater terrain-aided navigation system based on combination matching algorithm," *ISA Transactions*, vol. 78, pp. 80–87, 2018, doi: 10.1016/j.isatra.2017.12.018.

[14] I. Nygren and M. Jansson, "Terrain navigation for underwater vehicles using the correlator method," *IEEE Journal of Oceanic Engineering*, vol. 29, no. 3, pp. 906–915, 2004, doi: 10.1109/JOE.2004.833222.

[15] D. Wang, L. Liu, Y. Ben, P. Dai, and J. Wang, "Underwater terrain-matching algorithm based on improved iterative closest contour point algorithm," *Measurement and Control (United Kingdom)*, vol. 57, no. 7, pp. 893–902, 2024, doi: 10.1177/00202940231224569.

[16] X. Peng, Y. Zhang, Z. Xu, Z. Zhang, L. Chen, and C. Li, "PL-Net: towards deep learning-based localization for underwater terrain," *Neural Computing and Applications*, vol. 37, no. 21, pp. 15979–15994, 2023, doi: 10.1007/s00521-023-08931-0.

[17] P. Diaz-Garcia, F. Escalona, and M. Cazorla, "UKDM: underwater keypoint detection and matching using underwater image enhancement techniques," 2025, [Online]. Available: http://arxiv.org/abs/2504.11063

[18] J. Ma, X. Jiang, A. Fan, J. Jiang, and J. Yan, "Image matching from handcrafted to deep features: a survey," *International Journal of Computer Vision*, vol. 129, no. 1, pp. 23–79, 2021, doi: 10.1007/s11263-020-01359-2.

[19] Y. Zhao, T. Chen, J. Dai, X. Gao, and X. Chen, "A survey of deep-learning-based image matching algorithms," in *International Conference on Remote Sensing, Mapping, and Image Processing (RSMIP 2025)*, F. Tosti and R. Alvarez, Eds., SPIE, 2025, p. 115. doi: 10.1117/12.3067778.

[20] S. Xu, S. Chen, R. Xu, C. Wang, P. Lu, and L. Guo, "Local feature matching using deep learning: a survey," *Information Fusion*, vol. 107, p. 102344, 2025, doi: 10.1016/j.inffus.2024.102344.

[21] D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperPoint: self-supervised interest point detection and description," 2018, [Online]. Available: http://arxiv.org/abs/1712.07629

[22] P. Lindenberger, P.-E. Sarlin, and M. Pollefeys, "LightGlue: local feature matching at light speed," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, IEEE, 2023, pp. 17581–17592. doi: 10.1109/ICCV51070.2023.01616.

[23] H. Bian, Q. Chen, H. Zhang, L. Qin, H. Cui, and X. Wang, "Local feature-based image matching: a comprehensive review and robustness evaluation," *The Visual Computer*, vol. 42, no. 1, p. 47, 2026, doi: 10.1007/s00371-025-04278-w.

[24] H. Wang, J. Yu, S. Jiang, D. Zhang, and Q. Li, "Aerial–terrestrial image feature matching: an evaluation of recent deep learning methods," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 18, pp. 15688–15706, 2025, doi: 10.1109/JSTARS.2025.3564326.

[25] S. Song, L. Morelli, X. Wu, R. Qin, H. Albanwan, and F. Remondino, "Deep learning meets satellite images – an evaluation on handcrafted and learning-based features for multi-date satellite stereo images," 2024, [Online]. Available: http://arxiv.org/abs/2409.02825

[26] V. V. Prutyanov, M. A. Ternov, and D. S. Kostrov, "Analysis of deep feature matching algorithms in UAV visual localization," in *2024 International Russian Automation Conference (RusAutoCon)*, IEEE, 2024, pp. 565–570. doi: 10.1109/RusAutoCon61949.2024.10694214.

[27] A. Meijster, ,. J. B. T. M. Roerdink, and W. H. Hesselink, "A general algorithm for computing distance transforms in linear time," technical report. [Online]. Available: https://pure.rug.nl/ws/files/3059926/2002CompImagVisMeijster.pdf/

[28] OpenCV, "Adding borders to your images." [Online]. Available: https://docs.opencv.org/4.x/dc/da3/tutorial_copyMakeBorder.html

[29] K. J. Zuiderveld, "Contrast limited adaptive histogram equalization," *Graphics gems IV*. AP Professional, Boston, pp. 474–485, 1994.

[30] Rijkswaterstaat, "Current direction and speed," [Online]. Available: https://rwsos.rws.nl/viewer/embed/map/noordzee/stroming?&theme=light&hidecontrols=alerts%2Csnelpeil

[31] M. Yang *et al.*, "Image matching: foundations, state of the art, and future directions," *Journal of Imaging*, vol. 11, no. 10, p. 329, 2025, doi: 10.3390/jimaging11100329.

[32] ETH Zurich CVG, "SuperPoint and LightGlue." [Online]. Available: https://huggingface.co/ETH-CVG/lightglue_superpoint

[33] R. Wang, J. Wang, Y. Li, T. Ma, and X. Zhang, "Research advances and prospects of underwater terrain-aided navigation," vol. 16, no. 14. Multidisciplinary Digital Publishing Institute (MDPI), 2024. doi: 10.3390/rs16142560.

[34] Royal Netherlands Navy, "Hydrographic Service." Accessed: June 05, 2026. [Online]. Available: https://english.defensie.nl/organisation/navy/navy-units/hydrographic-service

Appendices

A. Assumptions and Justifications

Assumptions

We agreed upon the following set of assumptions for SAND-E with CGI and the Royal Netherlands Navy:

1. Only MBES bathymetric survey data is used as the sensing modality for both the reference map and real-time measurements.
2. A bathymetric reference map is available.
3. A bathymetric reference map accurately represents the seabed topography.
4. The bathymetric reference map is expressed in a north-up projected coordinate system.
5. The bathymetric reference map covers all the planned vessel trajectories.
6. A bathymetric measurement is available at each update step during navigation.
7. A bathymetric measurement accurately represents the seabed topography.
8. The bathymetric reference map and bathymetric measurements have a fixed and constant resolution.
9. The bathymetric reference map and bathymetric measurements measure the same physical quantity: depth.
10. The bathymetric reference map and bathymetric measurements use the same horizontal coordinate reference system.
11. The bathymetric reference map and bathymetric measurements have the same vertical datum.
12. The bathymetric reference map and bathymetric measurements are consistently georeferenced.
13. The bathymetric measurements are acquired in the vessel-aligned frame.
14. Tide, heave, roll, pitch, and water-level effects are already corrected by the vessel's own software.
15. No interior NoData values remain in the bathymetric reference map after preprocessing, and no NoData values remain in the measurements.
16. A predicted vessel position is available at each update step during navigation, either by INS or particle filter.
17. The predicted vessel position lies within the boundaries of the reference map.
18. For evaluation purposes, the true vessel position is available and treated as ground truth.
19. A true vessel position lies within the boundaries of the reference map.
20. An estimated vessel position lies within the boundaries of the reference map.
21. A vessel heading is known at the point of GNSS loss.
22. A vessel heading is available from a compass or INS at each update step during navigation.
23. A vessel heading is measured clockwise from North, following the standard maritime convention.
24. A vessel heading is measured with an accuracy of 0.01 degrees.
25. Speed over ground v is known and constant.
26. Time is accurately known.
27. A constant and uniform current acts on the vessel.
28. No INS is available for position predictions. These are solely provided by the particle filter, although an INS could serve as an alternative source if available.
29. Position fix error follows $\nu_k \sim \mathcal{N}(\mathbf{0}, \boldsymbol{R})$, i.e. the error is Gaussian and isotropic.
30. Process noise on position and heading is Gaussian.
31. After north-alignment, the spatial relationship between the measurement patch and the search region reduces to a pure translation, with no remaining scale or rotation differences.

Justifications

Both the reference map and MBES measurements are assumed to accurately represent the seabed topography, which holds when standard MBES acquisition and processing protocols are followed. Residual errors in either directly translate to positioning errors and are assumed small enough to be negligible. Tide, heave, roll, pitch, and water-level corrections are assumed to have been applied by the vessel's motion reference unit prior to input, which is standard practice in bathymetric navigation. Heading is assumed accurate to 0.01 degrees (indicated by the Royal Netherlands Navy), consistent with the typical specification of commercial INS and compass systems used in marine survey vessels. A constant and uniform current is assumed to act on the vessel as a simplification to keep the motion model tractable. The effect of the current is absorbed by the position process noise. No INS is assumed available for position prediction, representing the most constrained operational scenario. An INS could replace

or complement the particle filter if available. Position fix errors are assumed Gaussian and isotropic, enabling the Gaussian likelihood model in the particle weight update. This is expected to hold under nominal operating conditions.

B. Data and Dataset Generation

Bathymetric 2.5D data covering a region of the North Sea is provided by the Hydrographic Service [34]. The data is used both as the reference map for matching and as the source from which simulated MBES measurements are extracted during parameter selection and navigation experiments.

B.1. Measurement Generation

Standalone random measurements are generated from valid locations in the bathymetric reference map using a fixed random seed of 42. In total, 500 measurements are generated. Each measurement is a fixed-size bathymetric patch extracted with a randomly sampled heading between 0 and 360 degrees and a 40 px margin to avoid rotation artefacts. Candidate samples are rejected if the extraction region contains too much missing data, if the final measurement contains NoData cells, or if the patch is nearly constant. For each valid measurement, the ground-truth position, heading, template size, coordinate reference system, and vertical reference are stored as metadata.

B.2. Monte Carlo Route Generation

Monte Carlo routes are generated within the valid bathymetric reference map using a fixed random seed for reproducibility. Routing is performed in UTM coordinates so that distances are defined directly in metres. Each route consists of five waypoints connected by four fixed-length 6.5-km segments.

The first waypoint is sampled from valid raster locations. Subsequent waypoints are generated by sampling random heading changes, with a maximum turn angle of 45 degrees between consecutive segments and a maximum cumulative heading drift of 60 degrees. Candidate waypoints are rejected if they fall outside the valid raster area or if the full measurement extraction window contains NoData cells.

The resulting routes are used as input for the particle filter so that all positioning methods are evaluated on identical trajectories. At each time step, a measurement is extracted from the true vessel position and heading and passed to the positioning method.

C. System Specifications

Category	Component	Specification/Version
Hardware	Processor (CPU)	Intel(R) Core(TM) Ultra 7 265HX (2.60 GHz)
	Memory (RAM)	32,0 GB
	GPU	NVIDIA RTX PRO 4000 Blackwell Generation
Software	Operating System	Ubuntu 24.04.3 LTS
	Python	Version 3.12.3
	Primary IDE	VS Code

Table C.1: System specifications.

D. Experiments

Parameter	Symbol	Value	Unit
Max turn rate	$\dot{\psi}_{\max}$	55	deg/min
True heading noise	-	0.01	deg/step
Particle heading noise	-	3.0	deg/step
Waypoint switching radius	-	500	m

Table D.1: Additional navigation experiments configuration.

Method	Metric	Radius (m)			
		100	200	400	500
NCC	RMSE (m)	15.30	29.02	70.38	97.56
	Mean (m)	3.25	5.83	15.95	22.94
	Median (m)	0.00	0.00	0.00	0.00
	σ_r^2 (m^2)	117.00	421.15	2476.44	4758.64
	Invalid matches (%)	0.0	0.0	0.0	0.0
SP+LG	RMSE (m)	76.46	146.92	291.91	391.60
	Mean (m)	71.22	135.55	270.45	363.00
	Median (m)	71.74	138.17	272.69	373.24
	σ_r^2 (m^2)	2923.30	10793.21	42606.07	76673.41
	Invalid matches (%)	54.2	67.0	67.6	64.2

Table D.2: Positioning accuracy vs. search radius. Measurement width of 16 px.

Method	Metric	Radius (m)			
		100	200	400	500
NCC	RMSE (m)	4.41	14.65	51.58	60.76
	Mean (m)	0.86	2.35	8.66	10.13
	Median (m)	0.00	0.00	0.00	0.00
	σ_r^2 (m^2)	9.74	107.30	1330.18	1846.18
	Invalid matches (%)	0.0	0.0	0.0	0.0
SP+LG	RMSE (m)	20.38	100.77	273.63	343.34
	Mean (m)	6.93	77.14	245.41	314.02
	Median (m)	0.28	71.68	250.45	339.14
	σ_r^2 (m^2)	207.60	5077.70	37437.27	58942.01
	Invalid matches (%)	2.2	36.8	52.0	43.4

Table D.3: Positioning accuracy vs. search radius. Measurement width of 64 px.

Method	Metric	Δt (min)				
		1	5	10	15	30
NCC	RMSE (m)	25.5	222.0	631.2	310.5	591.1
	Std (m)	0.7	5.1	181.6	3.4	8.3
	Invalid matches (%)	0.0	0.0	0.0	0.0	0.0
SP+LG	RMSE (m)	512.4	618.3	452.3	855.5	1080.2
	Std (m)	54.1	157.4	121.6	150.7	232.6
	Invalid matches (%)	34.6	38.9	32.8	45.1	42.0
Combined	RMSE (m)	25.5	**34.5**	**179.5**	310.5	591.1
	Std (m)	0.7	*7.4*	**96.5**	3.4	8.3
	Invalid matches (%)	0.0	*2.3*	*14.3*	0.0	0.0

Table D.4: Effect of update interval Δt on positioning performance. *Italic*: worse than NCC, and **Bold**: better than NCC. Measurement width of 64 px.

4. Closing Discussion

SAND-E connects to several topics and skills within the MSc Geomatics programme. The framework builds directly on spatial data processing, as bathymetric raster data, NoData handling, coordinate reference systems, and vertical datums are fundamental throughout the data preparation and matching pipeline. Image matching, a core technique in photogrammetry, is central to the framework and is evaluated using both the classical NCC method and the learned SuperPoint–LightGlue approach, directly linking the thesis to machine learning. As a georeferenced sensor used for positioning and navigation, MBES connects the thesis to sensing technologies. The use of 2.5D bathymetric raster data also relates to digital terrain modelling, in which such data representations are widely used.

Beyond Geomatics, the thesis draws on robotics through the particle filter framework and on statistics and probability theory through the Bayesian state estimation that underlies it. The particle filter maintains a probability distribution over the vessel position that is updated at each time step, which is a form of probabilistic spatial reasoning. SAND-E therefore demonstrates how Geomatics methods and data can enable autonomous maritime navigation without external infrastructure, a contribution relevant across Geomatics, robotics, and spatial information science.

Based on the results and limitations of this thesis, the following recommendations are made for the Royal Netherlands Navy, CGI, and future research. The combined method is recommended for operational deployment over NCC alone, as it provides greater robustness under more challenging operating conditions. Real-time continuous updating is recommended over fixed intervals as this would further reduce drift. Route planning over terrain with distinctive features would further improve positioning accuracy, consistent with path planning recommendations in TAN literature [19]. Most military surface vessels are currently equipped with a Single-Beam Echosounder (SBES). However, because SBES provides only a narrow depth profile rather than a two-dimensional bathymetric patch, it is unlikely to contain sufficient spatial information for robust seabed fingerprinting. Therefore, equipping more vessels with MBES systems would be important for operational deployment, as MBES provides the richer bathymetric measurements needed for reliable matching. Regular updates of bathymetric reference maps, particularly in operationally relevant areas, would further help ensure that the reference data remains representative of the current seabed.

References

[1] M. Yang *et al.*, "Image matching: foundations, state of the art, and future directions," *Journal of Imaging*, vol. 11, no. 10, p. 329, Sep. 2025, doi: 10.3390/jimaging11100329.

[2] D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperPoint: self-supervised interest point detection and description," Apr. 2018, [Online]. Available: http://arxiv.org/abs/1712.07629

[3] P. Lindenberger, P.-E. Sarlin, and M. Pollefeys, "LightGlue: local feature matching at light speed," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, IEEE, Oct. 2023, pp. 17581–17592. doi: 10.1109/ICCV51070.2023.01616.

[4] G. Kim, H. Kim, E. Zio, and G. Heo, "Application of particle filtering for prognostics with measurement uncertainty in nuclear power plants," *Nuclear Engineering and Technology*, vol. 50, no. 8, pp. 1314–1323, Dec. 2018, doi: 10.1016/j.net.2018.08.002.

[5] EGNSS Centre of Excellence, "Eindrapport IKUS-II: inventarisatie kwetsbaarheden uitval satellietnavigatie." 2022.

[6] J. Zidan, E. I. Adegoke, E. Kampert, S. A. Birrell, C. R. Ford, and M. D. Higgins, "GNSS vulnerabilities and existing solutions: a review of the literature," vol. 9. Institute of Electrical, Electronics Engineers Inc., pp. 153960–153976, 2021. doi: 10.1109/ACCESS.2020.2973759.

[7] J. Trýb and J. Hospodka, "GNSS interference and security: impacts on critical infrastructure and mitigation strategies," in *Procedia Computer Science*, Elsevier B.V., 2025, pp. 2635–2644. doi: 10.1016/j.procs.2025.01.323.

[8] AIVD, MIVD, and NCTV, "Threat assessment state-sponsored actors 2," technical report, Nov. 2022. [Online]. Available: https://english.nctv.nl/documents/2022/11/28/threat-assessment-state-sponsored-actors%E2%80%942

[9] P. Jansen, "The impact of jamming and spoofing on GNSS signals," technical report, 2022. [Online]. Available: https://cas.tudelft.nl/

[10] R. T. Loannides, T. Pany, and G. Gibbons, "Known vulnerabilities of global navigation satellite systems, status, and potential mitigation techniques," *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1174–1194, 2016, doi: 10.1109/JPROC.2016.2535898.

[11] AIVD, MIVD, and NCTV, "Threat assessment of state actors 2025," technical report, 2025. [Online]. Available: https://english.nctv.nl/site/binaries/site-content/collections/documents/2025/09/25/threat-assessment-of-state-actors-2025/DBSA+2025+ENG_losbladig+digitaal.pdf

[12] D. Medina, C. Lass, E. P. Marcos, R. Ziebold, P. Closas, and J. García, "On GNSS jamming threat from the maritime navigation perspective," in *International Conference on Information Fusion (FUSION)*, Ottowa, 2019, pp. 1–7. doi: 10.23919/FUSION43075.2019.9011348..

[13] A. Androjna and M. Perkovič, "Impact of spoofing of navigation systems on maritime situational awareness," *Transactions on Maritime Science*, vol. 10, no. 2, pp. 361–373, Oct. 2021, doi: 10.7225/toms.v10.n02.w08.

[14] N. Charitou, "Understanding and mitigating GPS spoofing attacks in shipping," Jan. 2025, [Online]. Available: https://dione.lib.unipi.gr/xmlui/handle/unipi/17419?locale-attribute=en

[15] C. Jeffrey and R. Munro, *An introduction to GNSS: a primer in using global navigation satellite systems for positioning and autonomy*, 3rd ed. Hexagon, 2023. [Online]. Available: https://novatel.com/an-introduction-to-gnss

[16] H. Bian, A. Li, H. Ma, and R. Wang, "Essentials of navigation: a guide for marine navigation," technical report, 2024. doi: 10.1007/978-981-99-5636-4.

[17] P. D. Groves, *Principles of GNSS, inertial, and multisensor integrated navigation systems*. Artech House, 2013, p. 800. [Online]. Available: https://www.researchgate.net/publication/224969497_Principles_of_GNSS_Inertial_and_Multisensor_Integrated_Navigation_Systems_Second_Edition

[18] T. Ma, S. Ding, Y. Li, and J. Fan, "A review of terrain aided navigation for underwater vehicles," vol. 281. Elsevier Ltd, Aug. 2023. doi: 10.1016/j.oceaneng.2023.114779.

[19] G. Fan *et al.*, "Survey of terrain-aided navigation methods for underwater vehicles," vol. 11. Institute of Electrical, Electronics Engineers Inc., pp. 47510–47526, 2023. doi: 10.1109/ACCESS.2023.3275078.

[20] Oxford University Press, "Fingerprinting." [Online]. Available: https://www.oxfordlearnersdictionaries.com/definition/english/fingerprinting?q=fingerprinting

[21] J. Yang, Y. Cai, K. Zhao, H. Xie, and X. Chen, "Concepts and applications of chemical fingerprint for hit and lead screening," vol. 27, no. 11. Elsevier Ltd, Nov. 2022. doi: 10.1016/j.drudis.2022.103356.

[22] S. Singh, K. Bhat, X. Riley, B. Resnick, J. Thickstun, and W. De Brouwer, "Robust neural audio fingerprinting using music foundation models," Nov. 2025, [Online]. Available: http://arxiv.org/abs/2511.05399

[23] A. L.-C. Wang, "An industrial-strength audio search algorithm," Jan. 2003. [Online]. Available: https://www.ee.columbia.edu/~dpwe/papers/Wang03-shazam.pdf

[24] M. Irfan, J. Hernandez Fernandez, A. Omri, S. Sciancalepore, and G. Oligeri, "Device fingerprinting in power line communications," *Ad Hoc Networks*, vol. 178, Nov. 2025, doi: 10.1016/j.adhoc.2025.103955.

[25] S. He and S. H. Chan, "Wi-Fi fingerprint-based indoor positioning: recent advances and comparisons," vol. 18, no. 1. Institute of Electrical, Electronics Engineers Inc., pp. 466–490, Jan. 2016. doi: 10.1109/COMST.2015.2464084.

[26] J. Lasky, "Bathymetry." [Online]. Available: https://www.ebsco.com/research-starters/science/bathymetry

[27] NOAA Ocean Exploration, "Multibeam Sonar." [Online]. Available: https://oceanexplorer.noaa.gov/technology/sonar-multibeam/

[28] Royal Netherlands Navy, "Hydrographic Service." Accessed: Jun. 05, 2026. [Online]. Available: https://english.defensie.nl/organisation/navy/navy-units/hydrographic-service

[29] H. Bian, Q. Chen, H. Zhang, L. Qin, H. Cui, and X. Wang, "Local feature-based image matching: a comprehensive review and robustness evaluation," *The Visual Computer*, vol. 42, no. 1, p. 47, Jan. 2026, doi: 10.1007/s00371-025-04278-w.

[30] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperGlue: learning feature matching with graph neural networks," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2020, pp. 4937–4946. doi: 10.1109/CVPR42600.2020.00499.

[31] S. R. Dhayalkar, "Particle filter made simple: a step-by-step beginner-friendly guide," Nov. 2025, [Online]. Available: http://arxiv.org/abs/2511.01281

A. Declaration of AI/LLM Usage

Statement:
"Some parts of this thesis were developed with assistance from AI tools, specifically ChatGPT (OpenAI) and Claude (Anthropic). These tools were used for coding support and debugging, Typst document formatting, mathematical notation consistency checks, and grammar refinement. All research content, mathematical derivations, experimental results, analyses, and conclusions are my own. I verified that all content is original and meets academic integrity standards."

B. Reproducibility Self-Assessment

"All data and analysis code related to this thesis are publicly available at https://github.com/juliapille/SAND-E as of June 2027, following a one-year embargo period. Complete instructions and a sample workflow are provided in the project README. I rate the reproducibility of this thesis as **high** according to the provided scale. The bathymetric data used for the North Sea experiments is provided by the Royal Netherlands Navy and cannot be shared publicly due to its sensitive nature. However, the data used for the generalization evaluation on the Atlantic Ocean test area is publicly available via the UKHO Seabed Mapping portal at https://seabed.admiralty.co.uk/?x=-19567.88&y=6780270.16&z=5.00 under the name *2024 2025-144353 Northern Ireland West of Laconia Bank*. The repository includes a README.md in the data folder with instructions for downloading the sample data from the northern Atlantic Ocean near the coast of Ireland, which is the same data used for the generalization evaluation in Section 3. The data can be downloaded from https://drive.google.com/drive/folders/1pwbhBlExmcKFolqErTxbQcTMtLkkQhYh?usp=drive_link."


TUDelft
CGI

This document was typeset using Typst, and uses the template freely available at https://github.com/tudelft3d/msc_geomatics_thesis_typst.