Towards Generalisable Inverse Modelling for Frequency-Domain Diffuse Optical Tomography via a Hybrid CNN–Transformer

MSc Dissertation

Max Andrew Hart

School of Computer Science
College of Engineering and Physical Sciences
University of Birmingham
2024–25

Abstract

Frequency-domain diffuse optical tomography (FD-DOT) enables portable 3D imaging, yet learned reconstructions often degrade when the source–detector (SD) geometry or anatomy changes. This dissertation evaluates a concrete strategy for path-agnostic reconstruction: combine broad phantom/probe diversity with systematic geometry randomisation, and use a geometry-aware hybrid model. Stage 1 trains a 3D CNN autoencoder to learn a compact latent prior; Stage 2 uses a transformer to map tokens comprising log-amplitude, phase, and explicit SD coordinates into that latent while the Stage 1 decoder remains frozen.

On diverse ellipsoidal phantoms with widely varying SD layouts, Stage 1 attains low voxel error (≈ 0.064 mm^{-1}), high contrast (≈ 0.88), and Dice coefficient ≈ 0.71. Decoding Stage 2 predictions through the fixed decoder yields tightly clustered performance: total RMSE ≈ 0.134 mm^{-1}, contrast ≈ 0.613, and Dice ≈ 0.31. A single model operates across layouts and tissue shapes without per-case tuning; reconstructions are globally faithful with softened edges and occasional merging of nearby inclusions. The contribution is a concise, reproducible blueprint—data pipeline, training protocol, and geometry-aware architecture—for geometry-robust DL-DOT, with clear routes to sharper boundaries via targeted follow-ups.

Declarations

I certify that this project is my own work. Code development and debugging were assisted by Claude Sonnet (via GitHub Copilot in VS Code), and writing clarity, consistency, and coherence were improved with ChatGPT-5. I also used these tools to summarise papers and clarify background concepts during the literature review. The research questions, system design, implementation choices, experiments, and results are my own; any AI-generated suggestions were used as aids and were reviewed, adapted, and edited by me to ensure accuracy and originality.

Contents

CHAPTER 1

Introduction

1.1 Background and Motivation

Diffuse optical tomography (DOT) reconstructs tissue optical properties from near-infrared (NIR) measurements in the 650–900 nm window, enabling three-dimensional imaging of physiological parameters (e.g., blood volume, oxygenation). These biomarkers of vascularisation and haemodynamics make DOT attractive in oncology and neuroscience. Unlike ionising modalities (CT/PET), DOT is safe and portable, suiting longitudinal monitoring, point-of-care screening, and intraoperative guidance [1, 2].

Clinically, these advantages translate into clear use cases. In breast oncology—the most common cancer in women—early detection and treatment monitoring are critical for outcomes. DOT provides functional information such as tumour oxygenation and haemoglobin concentration that is not readily accessible with conventional imaging. In particular, DOT has been investigated for monitoring response to neoadjuvant chemotherapy, where frequent, non-invasive, low-cost imaging is required but impractical with MRI or mammography [3]. In neuroscience, DOT-based functional imaging has been used to study brain activation and cerebral oxygenation, offering a portable alternative to fMRI that is valuable at the bedside and in paediatrics [4].

Recent instrumentation expands DOT's potential: handheld and wearable systems support real-time scanning and adapt to varied patient geometries [5]. The bottleneck is now computation—reconstructions must be rapid and robust to probe variability, anatomy, and noise. Classical iterative solvers (diffusion-equation inversion with repeated Jacobians) are prohibitively slow, often minutes per volume even on high-performance hardware. Deep learning-based DOT (DL-DOT) offers sub-second inference while maintaining or improving physics-based fidelity [6].

1.2 Frequency-Domain Diffuse Optical Tomography

DOT uses near-infrared light in the 650–900 nm window, which maximises haemoglobin contrast while minimising water and lipid absorption. The principal parameters are absorption μ_a (mm^{-1}), reflecting chromophore concentration, and reduced scattering μ_s' (mm^{-1}), influenced by tissue microstructure; together they govern photon fluence and are the quantities to be reconstructed [1, 2].

Measurements are acquired using surface arrays of sources and detectors. Each source–detector

(SD) pair samples a diffuse photon path, with SD separation (SDS) controlling depth sensitivity. In frequency-domain DOT (FD-DOT), sinusoidal modulation produces a detected signal described by amplitude attenuation and phase shift relative to the input; working with $\log A$ (log-amplitude) and ϕ (phase) yields complementary sensitivity to μ_a and μ_s'. Short SDS ($< 15\,\text{mm}$) probe superficial layers, whereas larger separations (30–40 mm) reach deeper tissue at reduced SNR.

The forward model follows from the frequency-domain diffusion equation and, for realistic geometries, is solved numerically (e.g. via FEM) [7]. The operator $\mathcal{F}$ maps spatial fields of μ_a and μ_s' to boundary measurements of $\log A$ and ϕ. Recovering these parameters from sparse, surface-only data is underdetermined and therefore requires regularisation—via smoothness constraints, sparsity-promoting penalties, or learned data-driven priors [1]. Hereafter we use *FD-DOT* for frequency-domain DOT (amplitude and phase), reserve *DOT* for the modality in general, and *DL-DOT* for deep learning-based reconstruction. We next formalise the measurement representation and notation used throughout.

1.3 Problem Formulation and Notation

This dissertation focuses on frequency-domain diffuse optical tomography (FD-DOT), where the photon field is sinusoidally modulated at frequency f (Hz). For each source–detector (SD) pair, the measurement is expressed as a complex value:

$$M = Ae^{i\phi},$$

with A the detected amplitude and ϕ the phase shift relative to the source. Because amplitudes span several orders of magnitude, reconstructions use the logarithm of amplitude, $\log A$, together with ϕ. These two quantities form the core measurement features for each SD pair.

Each SD pair is described by the source and detector coordinates (x_s, y_s, z_s) and (x_d, y_d, z_d). The per-pair feature vector is

$$m_i = \{\log A_i,\ \phi_i,\ x_{s,i}, y_{s,i}, z_{s,i}, x_{d,i}, y_{d,i}, z_{d,i}\},$$

and a scan with N pairs forms

$$\mathbf{y} = \{m_1, \ldots, m_N\} \in \mathbb{R}^{N\times 8}.$$

The frequency-domain diffusion model solved by FEM defines

$$\mathbf{y} = \mathcal{F}(\mu_a, \mu_s') + \epsilon,$$

where $\mathcal{F}$ maps (μ_a, μ_s') to boundary $(\log A, \phi)$ and ϵ denotes additive Gaussian system noise.

The reconstruction task is to estimate voxelwise maps of μ_a and μ_s' on a $64 \times 64 \times 64$ grid (1 mm resolution), yielding $\sim 2.6 \times 10^5$ voxels per parameter, or about 5.2×10^5 unknowns in total. Denoting the reconstructions by $\hat{\mu}_a$ and $\hat{\mu}_s'$, the inverse mapping can be expressed as

$$\mathcal{G} : \mathbf{y} \mapsto \{\hat{\mu}_a, \hat{\mu}_s'\},$$

where $\mathcal{G}$ is implemented by a learned neural network.

This inverse problem is severely underdetermined. Even with $N = 1000$ SD pairs, the measurement tensor $\mathbf{y} \in \mathbb{R}^{1000\times 8}$ contains far fewer entries than the hundreds of thousands of voxels to be estimated. Sensitivity is also highly non-uniform, with superficial voxels contributing disproportionately more than deeper ones. Together, these properties make the problem intrinsically unstable without strong priors, motivating the design of models and training strategies that can embed spatial structure, accommodate probe variability, and generalise across diverse tissue anatomies.

1.4 Challenges in FD-DOT Reconstruction

FD-DOT faces challenges that arise from both the physics of light transport and the practical requirements of clinical deployment:

- **Ill-posedness.** Sparse, surface-only measurements must be mapped to dense three-dimensional volumes. Each detector integrates photons after multiple scattering, producing broad, overlapping sensitivity profiles; deep signals are especially weak, amplifying inversion instability. Without strong priors, reconstructions risk capturing only superficial structure while missing deeper inclusions.
- **Geometry shift.** In handheld or wearable systems, probe layouts vary with operator handling, patient anatomy, and motion. Classical FEM-based solvers can accommodate arbitrary geometries by recomputing Jacobians, but most deep learning models assume fixed layouts and degrade when geometry changes [8].
- **Noise robustness.** FD-DOT measurements are sensitive to electronic noise, coupling variability, and instrumental drift. Even small perturbations in amplitude or phase can destabilise reconstructions if not addressed during model design and training, underscoring the need for noise-aware approaches.
- **Simulation-to-real gap.** Synthetic datasets enable supervised training at scale but cannot fully capture heterogeneous anatomy, motion artefacts, or hardware imperfections. This mismatch introduces a domain gap between simulated and clinical data that must be narrowed for reliable deployment.
- **Latency.** Real-time use requires reconstructions in under 0.1 s per volume. Iterative solvers are far too slow, often demanding hundreds of iterations per scan. Learned inverse solvers are therefore essential to achieve clinically viable runtimes while preserving image quality.

Together, these challenges frame the central difficulty of FD-DOT: reconstructions must remain accurate despite sparse, noisy data, shifting probe geometries, and the simulation-to-clinical mismatch. Meeting these demands requires stronger priors, explicit adaptation to variability, and inference fast enough for real-time use. The next chapter reviews conventional physics-based solvers alongside recent learning-based approaches to establish current progress and the research gap that motivates this dissertation's methodology.

CHAPTER 2

Literature Review

2.1 Conventional Foundations in FD-DOT

This section outlines the physics and algorithms underpinning learning-based DOT: from radiative transport to the diffusion approximation, frequency-domain forward modelling with the finite element method (FEM), and classical inverse formulations with regularisation that motivate data-driven alternatives.

2.1.1 From radiative transport to diffusion

Fundamentally, photon propagation in tissue is described by the radiative transport equation (RTE), which evolves radiance over space and angle. The RTE is physically exact but high-dimensional and computationally prohibitive, making direct inversion infeasible for DOT [1]. In highly scattering tissue—typical of the NIR window (650–900 nm)—the angular distribution of photons rapidly becomes isotropic. Under this regime, the RTE reduces to the diffusion approximation, a parabolic PDE that describes photon fluence while discarding angular detail [1, 2, 9]. The diffusion model is sufficiently accurate for most DOT scenarios, though it breaks down near tissue boundaries or in regions of low scattering [10]. In the frequency-domain case, sinusoidal modulation yields a complex-valued diffusion equation whose boundary solutions predict the measured amplitude attenuation and phase shift—the observables in FD-DOT.

2.1.2 Frequency-domain forward modelling with FEM

Clinical use cases such as breast and brain imaging demand solvers that handle complex boundaries and heterogeneous optical properties. The finite element method (FEM) has become the de facto forward model in DOT for this reason [7]. The tissue domain is discretised into elements, optical parameters are defined locally, and the weak form of the diffusion equation is assembled into sparse linear systems. The resulting forward operator $\mathcal{F}$ maps spatial distributions of absorption μ_a and reduced scattering μ_s' to boundary measurements of log-amplitude ($\log A$) and phase (ϕ).

Alternative schemes—finite differences, boundary elements, Monte Carlo accelerations—have been explored, but FEM dominates as it flexibly accommodates anatomical priors (e.g., from MRI/CT) and frequency-domain extensions. Its drawback is computational: each forward solve requires factorising

or iteratively solving large systems, and repeated Jacobian evaluations for inversion compound the cost. These bottlenecks underpin the minutes-per-scan latency in conventional FD-DOT pipelines and motivate learned surrogates or end-to-end predictors.

2.1.3 Classical inverse formulations and regularisation

The inverse problem seeks maps of (μ_a, μ_s') from sparse, surface-only data. A common variational formulation is

$$\hat{\mu} \;=\; \arg\min_{\mu} \; \left\| \mathbf{y} - \mathcal{F}(\mu) \right\|_2^2 \;+\; \lambda\, R(\mu),$$

where $\mathbf{y}$ denotes measured $\log A$ and ϕ, $R(\mu)$ encodes prior structure, and $\lambda > 0$ balances fidelity against regularity. Standard solvers linearise $\mathcal{F}$ about the current estimate and update iteratively with Gauss–Newton or Levenberg–Marquardt schemes, relying on FEM-derived Jacobians for sensitivity [1]. These methods remain valuable but are slow (minutes per 3D volume) and degrade at low SNR.

Regularisation is essential. Quadratic (Tikhonov) penalties suppress noise but blur boundaries; sparsity and total variation (TV) priors sharpen inclusions but require non-smooth optimisation and careful parameter tuning [2]. Bayesian approaches, such as MAP estimation and approximation-error modelling, offer uncertainty quantification and hierarchical priors at the cost of further computation [11]. Anatomical priors from MRI/CT can reduce ambiguity and improve localisation, but they tie DOT to external imaging and do not eliminate the heavy Jacobian burden [1]. Across these strategies, three recurring issues are evident: (i) reliance on hand-crafted priors and manual tuning, (ii) vulnerability to model–data mismatch, and (iii) prohibitive latency from large-scale PDE solves.

As a result, the diffusion approximation and FEM provide a physically grounded, widely validated forward model for FD-DOT, and classical inverse solvers remain an important benchmark. Nonetheless, their computational cost and reliance on hand-crafted priors motivate learning-based methods that embed regularity from data and enable near real-time inference. The next section reviews such approaches in FD-DOT and related modalities, with emphasis on architectures that address geometry variability, noise, and domain shift.

2.2 Learning-Based Reconstruction for FD-DOT

Deep learning (DL) offers two capabilities that address the bottlenecks of FD-DOT: the ability to learn strong, data-driven priors, and the capacity to amortise computation for near real time inference once trained. Whereas conventional solvers require repeated FEM solves and hand-crafted regularisation, DL can embed prior knowledge and replace iterative inversion with a single forward pass. Insights from other imaging modalities motivate these strategies for FD-DOT.

2.2.1 Why learning helps FD-DOT

In CT, MRI, and photoacoustics, DL has improved reconstructions through two main strategies. The first is *unrolling*: a classical optimisation algorithm (e.g., gradient descent or iterative shrinkage) is rewritten as a neural network whose layers mimic iterations, combining physics-based operators with

learnable components. This preserves interpretability and enforces data-consistency while allowing priors to be adapted from training data [12, 13]. The second strategy is *direct inversion*, where a network learns a mapping from raw or minimally processed measurements to images, bypassing explicit iterations entirely. Examples such as the Learned Primal–Dual algorithm of Adler and Öktem show how unrolled approaches can handle non-linear forward operators while remaining efficient at test time [14]. Together, these precedents demonstrate that learning-based reconstructions can stabilise ill-posed problems under noise and sparsity—precisely the conditions that challenge FD-DOT.

2.2.2 DOT-specific DL: successes and limits

Within DOT, early studies applied convolutional encoder–decoders (e.g., U-Nets) to reconstruct absorption maps from simulated or phantom data, achieving sharper inclusions and large speed-ups over Tikhonov-regularised FEM solvers [15].Deep learning (DL) offers two capabilities that address the bottlenecks of FD-DOT: the ability to learn strong, data-driven priors, and the capacity to amortise computation for near real time inference once trained. More recent work has scaled these ideas to 3D and even demonstrated feasibility in vivo: Deng *et al.* introduced FDU-net, a three-module network trained on simulations and validated on human breast measurements, reporting improved anomaly localisation with sub-second inference [16]. Other frameworks (e.g., Periodic-net) show the generality of DL-based reconstructions across diffuse optical imaging tasks [17].

Despite these advances, limitations remain. Most published networks assume fixed probe geometries, modest datasets, and narrow anatomical variability. As a result, models degrade under geometry shift and often fail to transfer from simulation to experiment. These shortcomings motivate more flexible architectures that encode geometry explicitly, together with training regimes that expose models to richer phantom and probe distributions.

2.2.3 CNN encoders/decoders for volumetric DOT

CNNs remain attractive because they learn spatially local, translation-equivariant priors over volumetric images. In 3D FD-DOT, encoder–decoder topologies such as U-Nets and autoencoders provide a strong inductive bias for recovering compact inclusions against smooth backgrounds. Practical trade-offs arise: higher-resolution 3D convolutions improve spatial detail but increase latency and memory cost, while multi-head designs that estimate both μ_a and μ_s' add flexibility but also complexity.

A useful pattern—adopted in this dissertation—is to learn a *spatial latent* with a 3D CNN by autoencoding ground-truth volumes, and then to regress that latent from measurements. This separates (i) capturing anatomical or structural priors and (ii) fusing measurement information. Prior work in optical and medical imaging confirms that such learned spatial latents stabilise reconstructions and reduce noise sensitivity, making them a natural backbone for more advanced architectures [15, 16].

2.2.4 Transformers for measurement sequences

FD-DOT measurements naturally form sets of source–detector (SD) pairs, each described by $(\log A, \phi)$ together with the explicit 3D coordinates of the source and detector. Self-attention is well suited to

aggregate these tokens: it models long-range dependencies across SD pairs, is inherently permutation-invariant, and can incorporate geometry-aware embeddings to encode spatial layout [18].

Embedding SD coordinates allows the model to be *path-agnostic*—able to operate across probe configurations—while still retaining *geometry awareness*. Self-attention provides global context, which is important because deeper tissue structures may only weakly influence long-separation measurements. The main drawback of transformers is their data hunger, but this can be offset by strong spatial priors (CNN latents) and extensive geometry randomisation during training, both of which are emphasised in this dissertation.

2.2.5 Hybrid two-stage designs for FD-DOT

Hybrid CNN–Transformer frameworks combine the strengths of volumetric priors and measurement-sequence aggregation. Dale *et al.* demonstrated high-speed, multi-parameter FD-DOT using such a hybrid, reconstructing absorption and reduced scattering at sub-second rates [6]. In this two-stage paradigm (also adopted here), Stage 1 learns a volumetric latent from ground-truth (μ_a, μ_s'), while Stage 2 consumes SD tokens built from $(\log A, \phi)$ and coordinates, using attention to integrate global information and regress the latent (and hence the reconstructed volume).

Variants of this framework extend its capabilities: multi-query attention can stabilise aggregation, and sophisticated spatial embeddings improve geometry-awareness. These refinements improve robustness while retaining the latency advantage of amortised inference. Key benefits of the hybrid paradigm are twofold: (i) robustness, by grounding reconstructions in anatomy-like variability captured by CNN latents, and (ii) efficiency, by shifting computation into a single forward pass.

Typical limitations such as geometry shift or data shift can be mitigated by dataset design—through phantom/probe diversity and geometry randomisation—and by geometry-aware tokenisation. This dissertation extends that trajectory by placing particular emphasis on systematic geometry randomisation and improved spatial embeddings as strategies to enhance generalisation across probe layouts and anatomical variability. In summary, learning-based FD-DOT has evolved from CNN encoder–decoders to attention-based sequence models, with hybrid CNN–Transformer frameworks now offering a promising compromise between robustness and speed; these approaches, however, still face practical deployment challenges—including geometry variability, noise robustness, and the simulation-to-real gap—which form the focus of the next section.

2.3 Robustness, Baselines, and the Research Gap

This section turns from architectural advances to the practical challenges hindering deployment of learning-based FD-DOT. It consolidates limitations affecting robustness in real-world settings, then reviews a path-agnostic baseline before motivating the research gap addressed in this dissertation.

2.3.1 Deployment limitations

Geometry shift. In handheld or wearable systems, source–detector (SD) layouts vary with operator handling, patient anatomy, and motion. Models trained on a single geometry often degrade when

SD patterns change because the mapping from $(\log A, \phi)$ to volume depends on photon sampling paths. Classical FEM-based solvers accommodate arbitrary layouts by recomputing Jacobians [1, 7], but most DL-DOT studies assume fixed arrays or limited variability [15, 16]. This motivates *explicit* geometry handling in learned models and *systematic* geometry randomisation during training.

Noise robustness. FD-DOT measurements are sensitive to coupling variability, electronic noise, and instrumental drift. Small perturbations in amplitude or phase can destabilise reconstructions if models implicitly overfit to noise patterns. Conventional solvers manage this through regularisation and uncertainty modelling [1, 11], but learned approaches must incorporate noise-aware training and invariances to remain reliable across sessions and hardware.

Simulation-to-real gap. Large synthetic datasets enable supervised training, but they rarely capture the full heterogeneity of anatomy, motion artefacts, or hardware imperfections. As a result, models trained only on narrow synthetic distributions often transfer poorly to in vivo data [2, 10]. Bridging this gap requires richer phantom distributions, augmentation strategies that respect physics, and architectures designed to remain stable under moderate model–data mismatch.

2.3.2 Baseline: path-agnostic DL-DOT

Dale *et al.* proposed a hybrid CNN–Transformer framework that treats SD measurements as tokens augmented with explicit spatial information (source and detector coordinates). This allows the network to integrate arbitrary scanning pathways while reconstructing 3D absorption and reduced scattering at sub-second speeds [6]. The architecture combines a volumetric prior (learned by a CNN) with a transformer encoder that aggregates measurements through self-attention, and has been shown to generalise better across probe layouts than fixed-array models [6, 8, 19].

This baseline establishes two key principles for practical DL-DOT: (i) *geometry-aware tokenisation*, which mitigates layout dependence, and (ii) *amortised inference*, which meets clinical latency requirements. At the same time, its performance is still bounded by the diversity of training phantoms, the realism of noise models, and the extent of geometry variation represented during training. These remaining challenges define the space into which this dissertation contributes.

2.3.3 Problem statement and contributions

Building on the path-agnostic hybrid baseline of Dale [6], this dissertation addresses the challenge of *geometry- and anatomy-generalised* FD-DOT reconstruction that remains robust under realistic noise conditions. The central problem is to learn a mapping from frequency-domain measurement sequences to volumetric optical properties that remains stable across probe layouts, anatomical variability, and measurement perturbations.

The contributions of this work are threefold:

1. **Phantom and probe diversity.** A high-throughput pipeline generates ellipsoidal tissue volumes with diverse shapes, sizes, and orientations, together with randomised tumour inclusions and surface-aware source–detector placement. This yields a richer training distribution and broader anatomical/probe variability than prior slab-based datasets.

2. **Systematic geometry randomisation.** Each phantom produces a large set of source–detector measurements, which are dynamically subsampled into fixed-length sequences during training. This strategy enforces invariance to probe placement, augments the dataset, and mitigates the degradation typically seen under geometry shift.
3. **Hybrid CNN–Transformer framework.** A two-stage design is adopted that combines volumetric priors learned by a CNN with measurement-sequence aggregation by a transformer encoder. This hybrid approach leverages spatial structure alongside path-agnostic token processing, enabling efficient and robust reconstruction.

> *The hypothesis of this dissertation is that by combining phantom and probe diversity, systematic geometry randomisation, and a hybrid geometry-aware architecture, reconstruction models can achieve stronger generalisation across probe layouts and anatomical variability. The goal is not to surpass existing baselines outright, but to demonstrate that these strategies jointly improve robustness and stability under geometry shift, noise, and the synthetic-to-real gap, thereby taking a concrete step towards clinically viable DL-DOT.*

2.3.4 From research gap to methodology

The remainder of this dissertation translates these challenges and contributions into a concrete methodology. Chapter 3 introduces the physics-based data pipeline and geometry randomisation protocol, including phantom construction, optical property assignment, probe placement, and noise modelling. Chapter 4 presents the hybrid CNN–Transformer architecture; Chapter 5 details the training strategy that operationalises it. Chapter 6 reports experimental results and analysis. Chapter 7 presents Discussion, Limitations, and Conclusions. Together, these chapters translate the identified research gap into an implementable, evaluable framework.

CHAPTER 3

Physics-Based Synthetic Data Pipeline

3.1 Objectives and End-to-End Overview

As part of our *methodology*, this chapter documents the physics-based pipeline used to generate supervised training data for frequency-domain diffuse optical tomography (FD–DOT). It produces large, diverse, clinically plausible measurement–label pairs while ensuring reproducibility and a clean interface to the learning model. Diversity comes from anatomical variation and varying probe geometry; fidelity from solving the FD diffusion model on meshes derived from labelled anatomy; reproducibility from deterministic seeding and versioned configuration.

Phantoms are built on a $64\times64\times64\,\mathrm{mm}^3$ voxel grid (1 mm spacing) with healthy and tumour labels. Absorption μ_a [mm^{-1}] and reduced scattering μ_s' [mm^{-1}] maps are assigned and transferred to a tetrahedral mesh. Sources and detectors are placed on the accessible surface under instrument-plausible separations, and the FD diffusion model is solved to yield complex boundary data. Measurements are converted to log-amplitude and phase, noise is applied, and the *full* channel set (source/detector coordinates and ordering) is written to HDF5 with OP maps and provenance metadata. Normalisation and fixed-length subsampling for training are applied later at data loading. The next section (§3.2) specifies the phantom geometry and OP models that underpin this process.

3.2 Phantom Geometry and Optical Properties

This section specifies how volumetric phantoms are constructed and how optical properties (OPs) are attached. The workflow is: build a labelled voxel volume on a fixed grid; assign per-voxel absorption μ_a [mm^{-1}] and reduced scattering μ_s' [mm^{-1}] at a single wavelength (800 nm) according to tissue labels; then pass these OP maps forward for meshing and forward modelling in Section 3.3. This ordering ensures that the voxel OP maps saved as ground truth are consistent with those assigned element-wise on the mesh; the forward solver operates directly on these per-element fields.

3.2.1 Voxel Grid and Coordinates

Phantoms are defined on a cubic domain of $64 \times 64 \times 64\,\mathrm{mm}^3$ with 1 mm spacing in x, y, and z. Voxel indices $(i, j, k) \in \{0, \ldots, 63\}^3$ map to physical coordinates via $(x, y, z) = (i, j, k) \times 1\,\mathrm{mm}$. Three label

types are used: air (0), healthy tissue (1), and tumour inclusions (labels $2, \ldots, K{+}1$). All coordinates and distances are reported in millimetres.

3.2.2 Anatomical Phantoms: Healthy Tissue and Inclusions

The healthy-tissue background is a randomly oriented ellipsoid centred near the grid midpoint. With $R \in \mathrm{SO}(3)$ a random rotation (Euler-angle sampling) and (r_x, r_y, r_z) the semi-axes, a voxel at (x, y, z) is labelled healthy if

$$\left(\frac{x'}{r_x}\right)^2 + \left(\frac{y'}{r_y}\right)^2 + \left(\frac{z'}{r_z}\right)^2 \leq 1, \quad [x', y', z']^\top = R\,[x - c_x,\, y - c_y,\, z - c_z]^\top.$$

Semi-axes are drawn independently as $r_x, r_y, r_z \sim U(25, 30)\,\mathrm{mm}$. When $r_x{=}r_y{=}r_z$, the centred sphere of diameter 50 mm occupies $\approx 25\%$ of the 64 mm cube and the 60 mm sphere occupies $\approx 43\%$; general ellipsoids with $r_x, r_y, r_z \in [25, 30]\,\mathrm{mm}$ remain within these bounds.

Each phantom also contains $K \in \{0, \ldots, 5\}$ ellipsoidal tumour inclusions. Inclusion centres are sampled within the healthy region with a boundary safety margin; semi-axes are drawn as $r_x, r_y, r_z \sim U(5, 15)\,\mathrm{mm}$ with an independent random $R \in \mathrm{SO}(3)$ (Euler-angle sampling) per inclusion. At least 80% of each inclusion's volume must lie inside healthy tissue; voxels inside a valid inclusion *overwrite* the healthy label to tumour, and any out-of-tissue remainder is left as air.

Table 3.1: Core geometry settings used during phantom construction.

Item	**Setting**
Healthy-tissue semi-axes	$r_x, r_y, r_z \sim U(25, 30)\,\mathrm{mm}$ (independent draws)
Tumour semi-axes	$r_x, r_y, r_z \sim U(5, 15)\,\mathrm{mm}$ (independent draws)
Number of tumours	$K \in \{0, 1, 2, 3, 4, 5\}$
Orientation	Random rotation in SO(3) (Euler-angle sampling)

3.2.3 Optical Property Models and Notation

Let μ_a [mm^{-1}] denote absorption and μ_s' [mm^{-1}] reduced scattering at 800 nm. For each phantom, a healthy-tissue baseline $(\mu_a^{\mathrm{H}}, \mu_s'^{\mathrm{H}})$ is drawn from physiological intervals reported in Dale's thesis; tumour properties are then generated by multiplicative contrasts relative to the *same* baseline, ensuring coherent intra-phantom ratios:

$$\mu_a^{\mathrm{T}} = \alpha_a\, \mu_a^{\mathrm{H}}, \qquad \mu_s'^{\mathrm{T}} = \alpha_s\, \mu_s'^{\mathrm{H}}.$$

The resulting OP maps (μ_a, μ_s') are stored as two voxel volumes aligned to the labels and passed unchanged to meshing. Table 3.2 records the exact distributions adopted from Dale's thesis [19].

Table 3.2: Optical property distributions at 800 nm and tumour-to-healthy contrast model.

Tissue/Factor	μ_a [mm^{-1}]	μ_s' [mm^{-1}]
Healthy tissue	$U(0.003, 0.007)$	$U(0.78, 1.18)$
Tumour contrast	$\alpha_a \sim U(1.5, 3.5)$	$\alpha_s \sim U(1.5, 2.5)$

Labels are deterministically mapped to voxel-aligned optical properties using ranges and contrast factors consistent with Dale's thesis, preserving coherent tumour-to-healthy ratios and enabling direct comparison to prior FD-DOT datasets.

3.3 Meshing, Probe Layout, and FD Solve

This section uses voxel-aligned optical-property (OP) maps from Section 3.2 to produce complex frequency-domain (FD) measurements. First, the labelled volume is tetrahedralised and each element inherits (μ_a, μ_s') from the OP maps. Next, the *air–to–tissue* surface is extracted to place sources and detectors under instrument-plausible source–detector separation (SDS) constraints. Finally, the FD diffusion model is solved on the mesh to obtain log-amplitude and phase per measurement link. Noise and normalisation follow next.

3.3.1 Mesh Generation and Property Mapping

Starting from the binary label volume (air / healthy tissue / tumour) and its OP maps (μ_a, μ_s'), a surface-conforming tetrahedral finite-element mesh is generated with target cell size 1.65 mm. The tissue boundary is preserved during voxel-to-tetra conversion, and quality checks (e.g., element-volume distributions) are applied. Each tetrahedron is *assigned* (μ_a, μ_s') by label using the same draws defining the voxel OP maps—no smoothing or averaging—so per-element fields match the voxel maps exactly. Typical mesh scales are reported in Table 3.3: meshes usually contain ~ 65–90×10^3 tetrahedra with mean element volume close to 1 mm^3; counts scale with tissue coverage, and meshing dominates the runtime. The mesh and OP fields are then passed to the *NIRFASTer* forward model [7].

3.3.2 Surface Extraction and Source–Detector Layout

The accessible surface is defined as *air voxels 26-connected to tissue* (faces/edges/corners), matching the morphological dilation used in the simulator. A circular surface patch of radius 30 mm is centred on a valid surface location (orange points in Fig. 3.1). Within this patch, source positions (red) are sampled using a Poisson-disk rule with **5 mm** minimum source–source spacing, suppressing clustering and promoting near-uniform coverage. For each source, detectors (blue) are drawn *within the same patch* subject to source–detector separation (SDS) bounds of 10–40 mm (grey lines show example links). This yields 50 sources with 20 detectors each (1,000 links) while keeping forward solves low (one per unique source). As in Fig. 3.1, anatomical variability (three tumours on the left, one on the right) changes the feasible patch location and the resulting optode geometry; corresponding surface and patch sizes are quantified in Table 3.3.

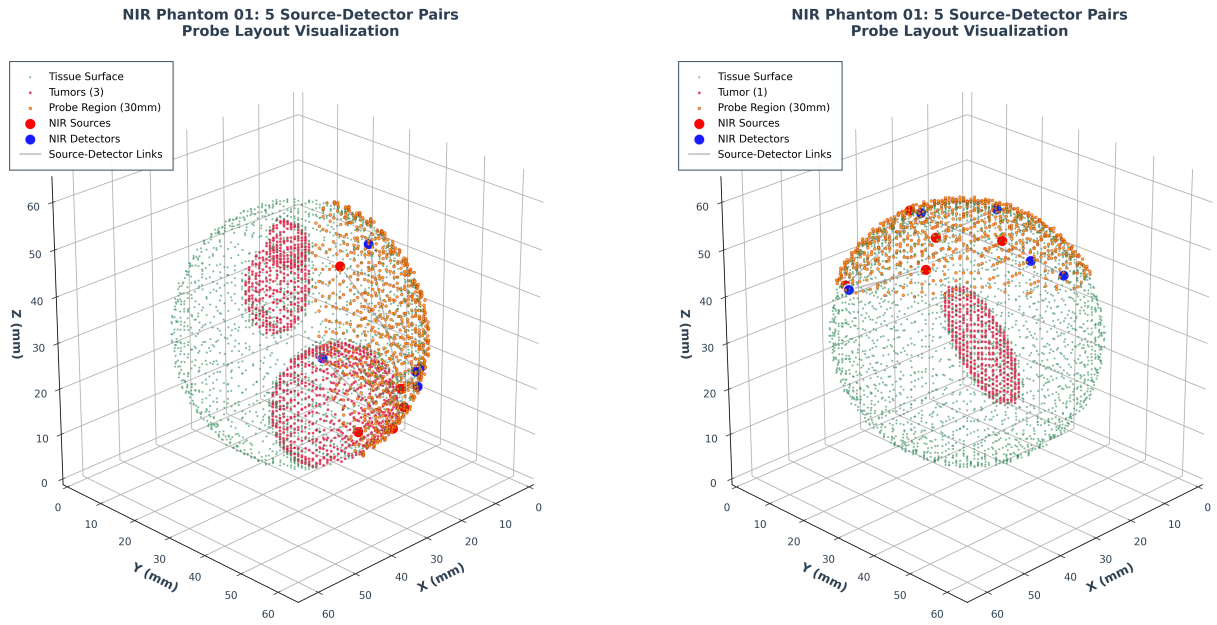

Figure 3.1: Representative probe layouts (left: **Phantom 4000** with three tumours; right: **Phantom 7800** with one tumour). Green points: tissue surface; magenta: tumours; orange: selected surface patch; red: sources; blue: detectors; grey: example source–detector links. For clarity, only a random subsample of *five* links is rendered in each panel; each scan uses 1,000 links in total.

Table 3.3: Mesh and surface statistics for the two phantoms shown in Fig. 3.1.

Metric	**Phantom 4000**	**Phantom 7800**
Tissue coverage (%)	26.9	31.4
Tetrahedra	68,314	78,421
Nodes	12,391	14,176
Mean element vol. ± sd (mm^3)	1.027 ± 0.209	1.045 ± 0.203
Meshed tissue volume (mm^3)	70,171	81,924
Accessible surface voxels	7,068	7,802
Patch voxels (R=30 mm)	2,378	2,266
Meshing time (s)	6.62	8.88
Total pipeline time (s)	9.17	11.92

Figure 3.1 shows the qualitative view (geometry, patch, optodes), and Table 3.3 quantifies the two cases. The right panel has greater tissue coverage (31.4% vs. 26.9%), driving a larger meshed volume; the patch-voxel difference is small and not visually apparent.

3.3.3 FD Diffusion Model and Measurements

We model light transport on the mesh with the frequency-domain diffusion approximation:

$$-\nabla\cdot\Big[D(\mathbf{r})\,\nabla\Phi(\mathbf{r},\omega)\Big] \;+\; \Big[\mu_a(\mathbf{r}) + \mathrm{i}\,\frac{\omega}{c/n}\Big]\Phi(\mathbf{r},\omega) \;=\; S(\mathbf{r},\omega), \tag{3.1}$$

where Φ is the complex fluence, $D = 1\Big/\big(3[\mu_a + \mu_s']\big)$ the diffusion coefficient, ω the modulation frequency, c the speed of light in vacuum, and n the refractive index. The forward model is solved with NIRFASTer-FF at 140 MHz; for each source, the complex field is computed and evaluated at detector nodes.

Before solving, `touch_optodes()` projects all optodes onto the mesh boundary, and these *mesh-projected* coordinates are saved for downstream token construction.

Link measurements are reported as log-amplitude and phase,

$$A_\ell = |\Phi_\ell|, \qquad \phi_\ell = \arg(\Phi_\ell) \text{ (degrees)}, \qquad \log A_\ell = \ln\Big(\max(A_\ell, 10^{-8})\Big).$$

At data loading, arrays are repacked into per-measurement tokens,

$$\mathbf{t}_\ell = \Big[\log A_\ell,\ \phi_\ell,\ x_s,\ y_s,\ z_s,\ x_d,\ y_d,\ z_d\Big]^\top,$$

where (x_s, y_s, z_s) and (x_d, y_d, z_d) are the mesh-projected source and detector coordinates in millimetres. Dataset-level noise and normalisation follow in Section 3.4.

3.4 Noise, Standardisation, and Data Packaging

This section converts the mesh–based FD solutions of §3.3 into learning-ready sequences and files, detailing the noise model, data-loader standardisation, the HDF5 layout, and dataset-level checks.

3.4.1 Measurement Noise

We add instrument-like noise to the raw complex measurements before features are formed. Amplitude is perturbed with zero-mean Gaussian noise whose standard deviation is 0.5% of that phantom's mean amplitude (computed across its 1,000 source–detector measurements), yielding a comparable relative perturbation across phantoms. Phase (in degrees) receives zero-mean Gaussian noise with a fixed 0.5° standard deviation to model a consistent phase jitter. After noise, we construct features as $\log A = \ln(\max(A, 10^{-8}))$ and ϕ (degrees). We intentionally use this simple Gaussian model to keep the focus on probe geometry and anatomical variation; more sophisticated, instrument-aware noise has been explored previously, e.g. by Dale et al. [19], and can be incorporated in future work.

3.4.2 Standardisation and Sequence Assembly

All preprocessing occurs in the data loader. Measurement channels $(\log A, \phi)$ are *independently* z-scored using training-set statistics computed over all source–detector measurements from the training phantoms. Coordinate channels use a simple affine normalisation: subtract the cube centre $(32, 32, 32)$ mm and divide by the side length (64 mm), applied per coordinate to the mesh-projected source and detector positions; values lie in $[-0.5, 0.5]$. Each phantom yields one scan comprising **1,000** source–detector measurements. For learning, we form a fixed-length sequence by subsampling **256** measurements *without replacement*. Training draws a *new random* subsample each epoch; validation and test use a deterministic subsample for exact repeatability.

3.4.3 Data packaging and quality checks

Each phantom is saved as a single HDF5 containing the full scan (1,000 measurements), voxel-aligned optical-property maps with labels, mesh-projected and original source/detector coordinates, and a measurement index (source–detector IDs); metadata are stored as attributes and datasets are compressed. Automated checks verify mesh quality, optode projection, solver completion (no NaNs/Infs), and plausible $\log A/\phi$; failures trigger a per-phantom retry. Figures give dataset-level sanity checks; unless noted, plots use a 500-phantom subsample, and the probe-layout coverage aggregates 10 phantoms.

Figure 3.2 shows tumour counts are evenly distributed between 0 and 5 per phantom, and that tumours exhibit higher absorption and scattering than healthy tissue.

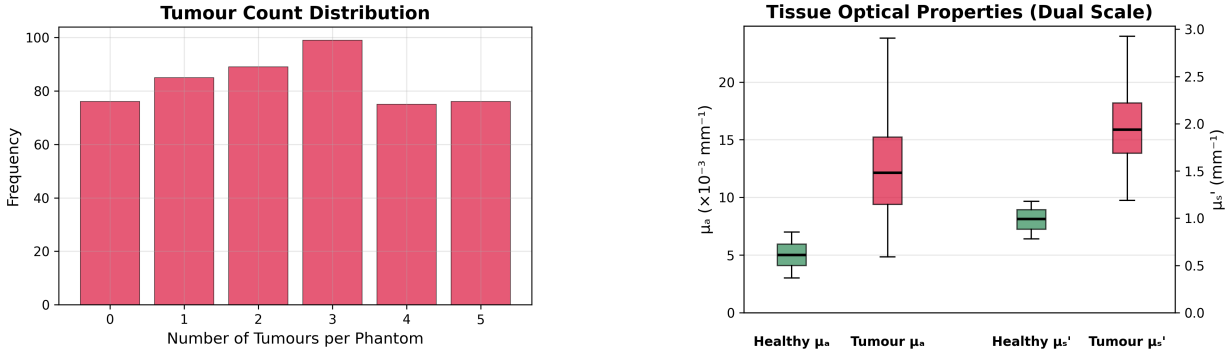

Figure 3.2: **Left:** Tumour count per phantom. **Right:** Optical properties (healthy vs. tumour).

Figure 3.3 focuses on measurement geometry. Source–detector separations fall within the 10–40 mm design bounds with a median near 25 mm, while probe coverage achieves near-uniform placement over the 30 mm patch. The Poisson-disk rule enforcing $\geq$5 mm source–source spacing applies per phantom, so the aggregated multi-phantom coverage appears denser.

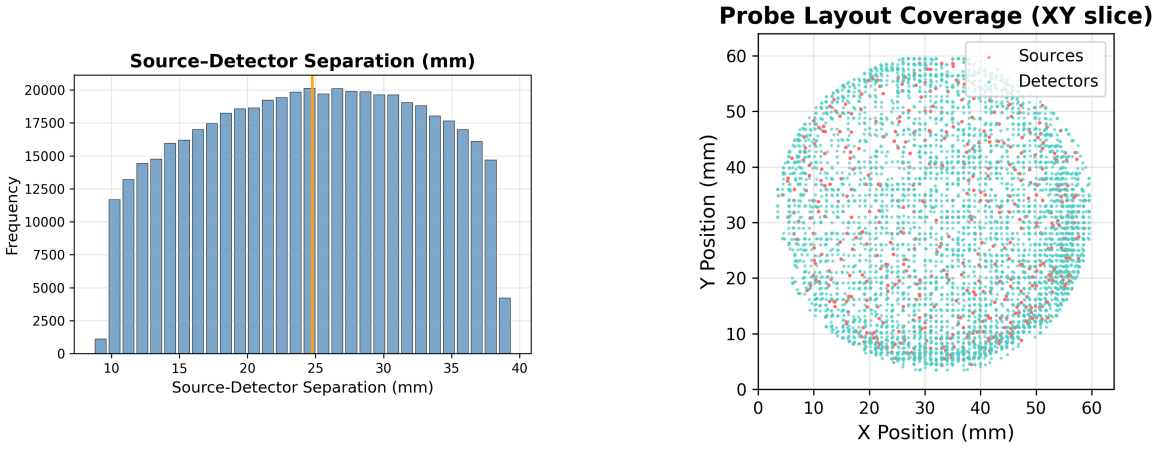

Figure 3.3: **Left:** Source–detector separation histogram. **Right:** Probe layout coverage.

The dataset comprises 10,000 phantoms, each saved as a single HDF5 with 1,000 measurements; at load time these are standardised into 256-measurement sequences. This provides a uniform, physics-faithful interface that Chapter 4 uses directly when introducing the hybrid CNN–Transformer.

CHAPTER 4

Proposed Hybrid CNN–Transformer Model

4.1 Architectural Overview

From a *methods* standpoint, this chapter develops a two-stage hybrid model mapping frequency-domain DOT *source–detector measurements* to volumetric optical properties. *Stage 1* is a compact 3D CNN autoencoder trained on ground-truth absorption and scattering volumes; its decoder is reused as a volumetric generator, ensuring reconstructions remain stable and anatomically plausible. By constraining outputs through this decoder, the model benefits from a spatial prior that captures structure and suppresses artefacts. *Stage 2* operates on measurements $[\log A, \phi]$ with explicit source/detector coordinates. From the full set of 1,000, we use a fixed subset of N=256 to present a consistent input to Stage 2. These tokens are processed by a transformer encoder that models long-range interactions and aggregates them into a 256-D latent $\hat{\mathbf{z}}$, which the frozen decoder maps to $\hat{\mathbf{V}} \in \mathbb{R}^{2\times64\times64\times64}$. The result is *amortised inference*: a direct mapping from measurements to volumes that avoids costly iterative optimisation while remaining geometry-aware through coordinates and attention.

The overarching idea is *inspired by and builds on* the CNN–Transformer paradigm for path-agnostic DOT established by Dale *et al.* [6, 8, 19]. Their work introduced the philosophy of combining convolutional spatial priors with transformer-based measurement encoding, which this project adopts explicitly. The specific implementation, however, was developed independently: the measurement embedding, transformer configuration (depth, heads, widths, and normalisation strategy), the aggregation mechanism, and the 3D CNN autoencoder architecture (encoder/decoder blocks and latent width) were all specified and tuned in this study. In this way, the model acknowledges Dale *et al.*'s foundation while contributing an independently designed implementation tailored to the datasets and objectives of this dissertation. A consolidated schematic appears in Fig. 4.1. We now develop the Stage 1 CNN autoencoder.

4.2 Stage 1: CNN Autoencoder

Stage 1 provides a compact learned *spatial prior* that constrains reconstructions to anatomically plausible volumes. A 3D residual autoencoder encodes (μ_a, μ_s') on a 64^3 grid into $\mathbf{z} \in \mathbb{R}^{256}$ and decodes to $\hat{\mathbf{V}} \in \mathbb{R}^{2\times64\times64\times64}$. The design (7 M parameters) balances stability and speed while retaining capacity for breast-like structure.

4.2.1 Encoder: Residual Blocks and Downsampling

The encoder uses an efficient residual backbone: an initial strided reduction, then four residual stages that progressively downsample to a small grid, followed by global average pooling to yield a fixed-length code. Batch normalisation and ReLU are used throughout; a light dropout precedes the final linear projection to 256-D. Residual connections ease optimisation [20], batch normalisation accelerates convergence [21], and global average pooling provides a parameter-efficient bottleneck [22]. The channel schedule was pruned from an earlier 27 M-parameter variant to 7 M to cut memory and latency without measurable loss downstream.

4.2.2 Latent Space Design

The latent $\mathbf{z} \in \mathbb{R}^{256}$ is the sole interface to Stage 2: the transformer learns to predict a compatible $\hat{\mathbf{z}}$ from measurements, and the decoder maps it to volumes. A compact code is desirable to (i) impose a strong prior and (ii) present a well-conditioned target for alignment. Classical autoencoder results support the use of low-dimensional codes for faithful reconstruction when trained appropriately [23]. In practice, a 128-D bottleneck was evaluated but produced over-smooth, under-detailed volumes; 256-D delivered sharper structure without destabilising Stage 2. Larger codes were not pursued to keep alignment tractable. *Design note (skips).* U-Net style encoder–decoder skip connections are common for 3D medical imaging [24], but ablations here showed that skips let information bypass the bottleneck, yielding good reconstructions yet a *weak* latent. When Stage 2 attempted teacher–student alignment, the latent proved insufficiently informative, and training degraded. Skips were therefore removed to ensure *all* information is routed through $\mathbf{z}$.

4.2.3 Decoder: Progressive Upsampling and Reconstruction

The decoder mirrors the encoder: a linear expansion reshapes the 256-D code to a small seed volume, followed by five upsampling stages that double spatial size while reducing channel width, ending with a 3^3 convolution to produce $(\hat{\mu}_a, \hat{\mu}'_s)$. No hard output activation is applied; physical scaling occurs outside the model. Standard anti-checkerboard practices are used to avoid artefacts [25]. Once pre-trained, the decoder is frozen in Stage 2 so the transformer learns only the geometry-aware mapping into the 256-D latent.

4.3 Stage 2: Geometry-Aware Transformer

Stage 2 converts a *set* of frequency-domain *source–detector measurements* into a single 256-D latent $\hat{\mathbf{z}}$ compatible with the frozen decoder. Each measurement $\mathbf{m}_i = [\mathbf{s}_i, \mathbf{p}_i] \in \mathbb{R}^8$ comprises *signals* $\mathbf{s}_i = [\log A, \phi]$ and *coordinates* $\mathbf{p}_i = [\mathbf{x}_s, \mathbf{x}_d]$, standardised/scaled as in Section 3.4.2. The pipeline: select a fixed subset, embed to 256-D tokens, process with a transformer encoder, and pool tokens to a scan-level latent for decoding. Set-attention architectures naturally handle unordered collections and learned pooling [26, 27], aligning with prior hybrid DOT designs [8, 19].

4.3.1 Fixed-length Subsampling

Each phantom yields a scan of 1,000 measurements. *Before* embedding, a fixed subset of N=256 is drawn: random per batch during training and deterministic for validation/testing to ensure exact repeatability. This achieves (i) constant transformer tensor shapes, (ii) controlled compute/memory, (iii) strong data augmentation via many distinct subsets across epochs, and (iv) robustness to probe geometry, since the effective source–detector layout varies from batch to batch. Denote the subset $\mathbf{M}_{\text{sub}} \in \mathbb{R}^{N\times 8}$. Fixing N=256 also obviates padding and attention masks, simplifying implementation and keeping compute predictable.

4.3.2 Spatially-Aware Embedding

For each $\mathbf{m}_i \in \mathbf{M}_{\text{sub}}$, we map $\mathbf{m}_i$ to a token $\mathbf{t}_i \in \mathbb{R}^{256}$ using two small MLP branches with GELU activations and LayerNorm, followed by fusion. The *signal* branch uses $2 \to 16 \to 32$; the *position* branch uses $6 \to 32 \to 64$. Their outputs (96-D) are concatenated and fused via $96 \to 128 \to 256$; stacking across i forms $\mathbf{T} \in \mathbb{R}^{N\times 256}$. Ablations guided these choices: a narrower position branch (ending at 32) underfit geometry and weakened downstream reconstructions, whereas the adopted $6 \to 32 \to 64$ path captured greater variance in source–detector layout. Likewise, replacing the two-step fusion with a direct $96 \to 256$ projection proved less stable and produced softer volumes; the intermediate 128-D layer consistently improved results, indicating it helps mix signal and coordinate subspaces before expansion. LayerNorm was retained to keep the combined representation well-conditioned when signals and coordinates arrive with different scales.

4.3.3 Transformer Encoder

The token matrix $\mathbf{T} \in \mathbb{R}^{N\times 256}$ is processed by an 8-layer, 8-head *pre-norm* transformer encoder [18]. "Pre-norm" places LayerNorm *before* each sublayer, improving optimisation stability in deeper stacks. Each layer applies multi-head self-attention with head dimension 256/8=32, followed by a position-wise MLP with GELU activations and hidden width 4×256=1024 (the "MLP ratio 4"). No additional positional encoding is injected: explicit coordinates in $\mathbf{t}_i$ already carry spatial information, letting attention learn geometry-aware interactions directly. The encoder output is $\mathbf{H} \in \mathbb{R}^{N\times 256}$, a token-wise refinement of $\mathbf{T}$ that captures non-local relations across measurements and feeds the pooling module below. We evaluated smaller settings (4 layers/4 heads and 6 layers/6 heads), which underfit and yielded weaker reconstructions through the frozen decoder; the 8-layer/8-head configuration provided a strong balance between expressivity and the $O(N^2)$ attention cost at N=256.

4.3.4 Multi-Query Attention Pooling

Prior work used global mean pooling in related hybrid DOT models [6, 8, 19], implicitly weighting all measurements equally. Because informativeness varies with separation and location, we instead use *multi-query* attention pooling.

Let the transformer output be $\mathbf{H} \in \mathbb{R}^{N\times d}$ with d=256, and learn Q=4 queries $\mathbf{Q} \in \mathbb{R}^{Q\times d}$. With

linear projections

$$\mathbf{K} = \mathbf{H}\mathbf{W}_K, \qquad \mathbf{V} = \mathbf{H}\mathbf{W}_V \quad (\mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{d\times d}),$$

each query $\mathbf{Q}_q$ attends to tokens via

$$\boldsymbol{\alpha}_q = \mathrm{softmax}\Big(\tfrac{\mathbf{Q}_q\mathbf{K}^\top}{\sqrt{d}}\Big), \qquad \mathbf{h}_q = \boldsymbol{\alpha}_q\,\mathbf{V} \in \mathbb{R}^{1\times d}.$$

Concatenating the Q summaries and projecting yields the scan latent

$$\hat{\mathbf{z}} = \mathrm{Proj}\Big([\mathbf{h}_1; \ldots; \mathbf{h}_Q]\Big) \in \mathbb{R}^{256}.$$

This follows Set Transformer's PMA and Perceiver's latent-query idea for order-agnostic set summarisation [26, 27]. Empirically, Q=4 outperformed mean pooling and single-query variants, while larger Q gave diminishing returns. The resulting $\hat{\mathbf{z}}$ is passed to the frozen decoder to produce $\hat{\mathbf{V}} \in \mathbb{R}^{2\times 64\times 64\times 64}$.

4.3.5 Tensor Shapes at a Glance

Table 4.1 consolidates the principal tensors and their shapes for Stage 2 and is intended to aid implementation checks and cross-references in later sections.

Table 4.1: Stage 2 tensors and shapes per batch (B), with N=256, d=256.

Quantity	Symbol	Shape
Full measurements	$\mathbf{M}$	$[B, 1000, 8]$
Subsampled measurements	$\mathbf{M}_{\text{sub}}$	$[B, N, 8]$
Embedded tokens	$\mathbf{T}$	$[B, N, d]$
Transformer outputs	$\mathbf{H}$	$[B, N, d]$
Pooled scan latent	$\hat{\mathbf{z}}$	$[B, d]$
Decoder output volume	$\hat{\mathbf{V}}$	$[B, 2, 64, 64, 64]$

4.4 Hybrid Integration, Alignment, and Inference

Stage 2 and the Stage 1 decoder form a single mapping from unordered measurement sets to volumes. The transformer produces a scan-level latent $\hat{\mathbf{z}} \in \mathbb{R}^{256}$; the pre-trained components from Stage 1—*encoder* (to generate target codes) and *decoder* (to render volumes)—are frozen by default, so reconstructions remain constrained by the learned spatial prior rather than ad-hoc postprocessing.

4.4.1 Latent Alignment and Decoder Reuse

During training, ground-truth volumes are passed through the *frozen* Stage 1 encoder to obtain target latents $\mathbf{z} \in \mathbb{R}^{256}$. In parallel, the Stage 2 stack maps the subsampled measurements to $\hat{\mathbf{z}}$; the loss supervises *only* this 256-D interface, encouraging the measurement encoder to produce codes the

decoder already "understands." This keeps supervision bandwidth low (256 numbers versus 64^3 voxels) and stabilises learning—an idea consistent with using a compact latent as the contract between encoder and decoder [28], and with the representation-matching spirit of distillation [29]. The choice to align latents, rather than supervise voxel outputs directly, follows the hybrid philosophy introduced for DOT by Dale *et al.* [8, 19]. Supervision therefore acts only on the 256-D latent interface. Figure 4.1 summarises this training alignment and the corresponding inference path.

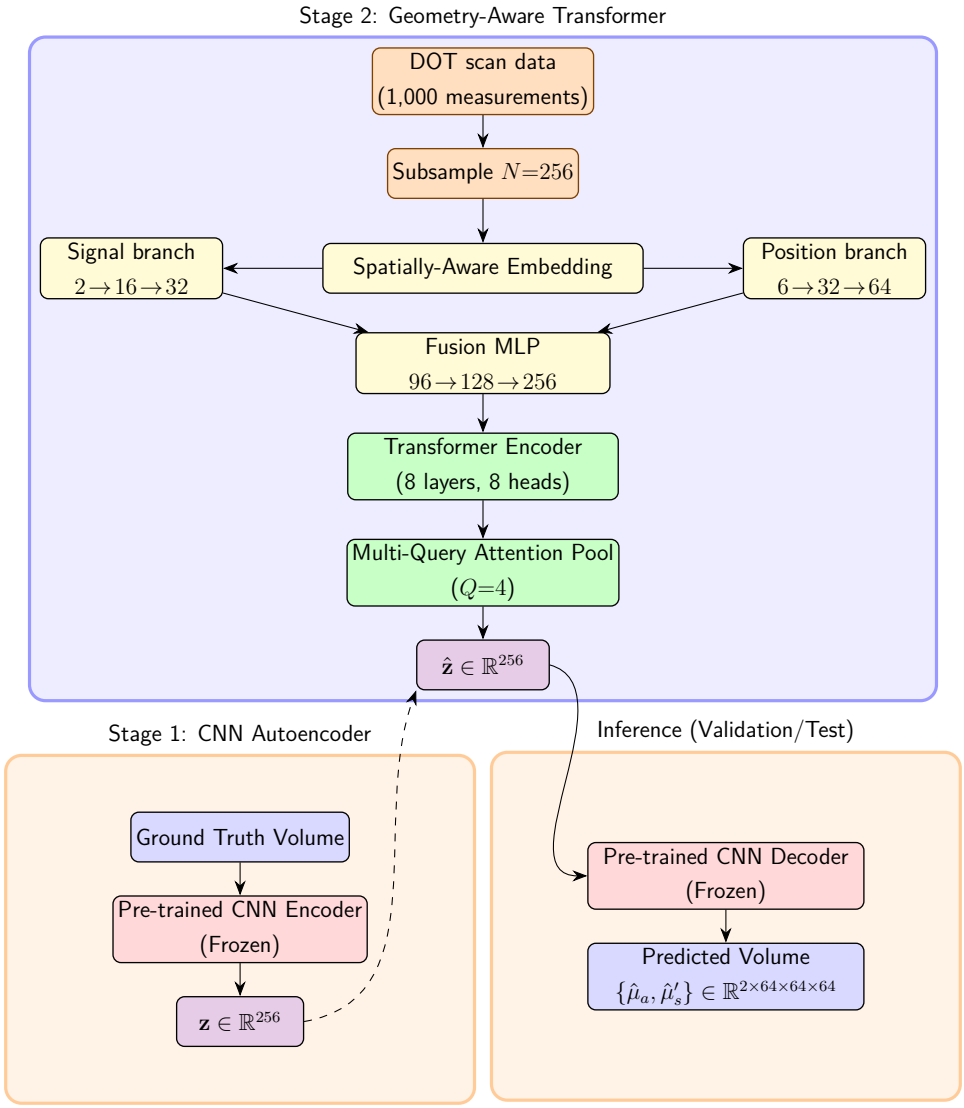

Figure 4.1: End-to-end hybrid pipeline used in §4.4. *Centre:* Stage 2 maps a fixed subset of measurements to a scan-level latent $\hat{\mathbf{z}}$. *Left:* during training, $\hat{\mathbf{z}}$ is aligned to teacher codes $\mathbf{z}$ from the frozen Stage 1 encoder. *Right:* at test time, the frozen decoder reconstructs $(\hat{\mu}_a, \hat{\mu}'_s)$ from $\hat{\mathbf{z}}$.

4.4.2 Inference Pathway

At test time, the pipeline runs in a single forward pass (right panel of Fig. 4.1): (i) standardise signals and scale coordinates; (ii) select N=256 measurements; (iii) embed as tokens; (iv) process with an 8-layer/8-head transformer; (v) pool via multi-query attention to obtain $\hat{\mathbf{z}}$; and (vi) decode to $(\hat{\mu}_a, \hat{\mu}'_s)$ on a 64^3 grid. This yields a direct, single-pass inversion with fixed tensor shapes and no iterative optimisation. Runtime and throughput results are given in Chapter 5.

CHAPTER 5

Training Strategies and Optimisation

5.1 Overview and Objectives

As part of the *methodology*, this chapter turns the design into a working learner. It details *how* the hybrid model is trained—what supervision is used, which objectives drive optimisation, and which practical choices keep training stable and repeatable. The pipeline is two-stage: *Stage 1* pre-trains a compact 3D CNN autoencoder on standardised (μ_a, μ_s') volumes to form a spatial latent and decoder. *Stage 2* trains a transformer to map tokenised measurements into that latent, with the decoder frozen so validation yields interpretable full-volume reconstructions. Training operates in the standardised space; selection and reporting use raw physical units for clinical interpretability (§5.4).

Our aims are pragmatic. First, *reliable learning*: objectives and schedules avoid divergence and prioritise physically meaningful validation signals. Second, *efficient learning*: stage-matched schedulers and lightweight components maintain throughput without sacrificing fidelity. Third, *rigorous protocol*: a fixed data split, three-seed repeats, and deterministic evaluation make outcomes comparable (details in §5.4). We now detail Stage 1 pre-training.

5.2 Autoencoder Pre-Training for the Spatial Prior

The first stage establishes a spatial prior by training a 3D convolutional autoencoder on ground-truth optical-property volumes. The objective is to compress anatomical structure into a well-conditioned latent $z \in \mathbb{R}^{256}$—chosen to interface cleanly with the transformer in §5.3—and a decoder that maps such codes back to (μ_a, μ_s'). We first set out the supervision signal and the standardisation that balances the loss, then describe the schedule used to fit a stable teacher for Stage 2.

5.2.1 Supervision and Targets

The pre-training task is voxelwise reconstruction of absorption μ_a and reduced scattering μ_s' (mm^{-1}). Let $V \in \mathbb{R}^{2\times 64\times 64\times 64}$ denote the target with channels (μ_a, μ_s') and $\hat{V}$ the reconstruction. The loss is the root mean squared error (RMSE)

$$\mathcal{L}_{\text{vox}} = \sqrt{\frac{1}{2\,|\Omega|} \sum_{c\in\{\mu_a,\mu_s'\}} \sum_{v\in\Omega} \left(\hat{V}_v^{(c)} - V_v^{(c)}\right)^2}, \tag{5.1}$$

where Ω indexes voxels ($|\Omega| = 64^3$). Measurement data are *not* used in Stage 1; all spatial information must pass through the encoder's bottleneck z, encouraging the latent to capture global structure rather than local artefacts.

5.2.2 Standardisation

To stabilise optimisation and make the channels commensurate, each is z-scored using statistics computed over the training split (8k volumes):

$$V^{(\mu_a)} \leftarrow \frac{V^{(\mu_a)} - \mu^{\text{train}}_{\mu_a}}{\sigma^{\text{train}}_{\mu_a}}, \quad V^{(\mu'_s)} \leftarrow \frac{V^{(\mu'_s)} - \mu^{\text{train}}_{\mu'_s}}{\sigma^{\text{train}}_{\mu'_s}},$$

with $\mu^{\text{train}}_{\mu_a}, \sigma^{\text{train}}_{\mu_a}$ and $\mu^{\text{train}}_{\mu'_s}, \sigma^{\text{train}}_{\mu'_s}$ the channel-wise mean and standard deviation over *training* data only. Using train-only statistics avoids leakage from validation/test, while equalising channel scale prevents the higher-variance channel from dominating the RMSE. The same affine transform is applied to validation and test volumes; for Chapter 6, predictions are inverse-transformed to physical units.

5.2.3 Optimiser and Schedule

Parameters are updated with AdamW [30], which decouples weight decay from the adaptive update and yields more predictable regularisation than ℓ_2 with moments. A OneCycle learning-rate policy [31] runs for 200 epochs: the rate rises smoothly from a small start to a peak, then anneals to a low final value. This encourages rapid early progress and stable late convergence. Mild weight decay (10^{-3}) regularises convolutional weights (biases and normalisation parameters excluded), global gradient-norm clipping (max 1.0) guards against spikes, and a small CNN dropout (0.05) curbs overfitting without harming spatial fidelity. Together with the standardisation above, these choices promote smooth loss curvature and a well-conditioned latent space, so Stage 2 begins from a stable teacher whose reconstructions are already anatomically faithful.

Figure 5.1 shows the schedule used throughout Stage 1. The warm-up, peak, and anneal phases align with observed loss smoothness and stable gradient magnitudes, supporting consistent convergence across seeds.

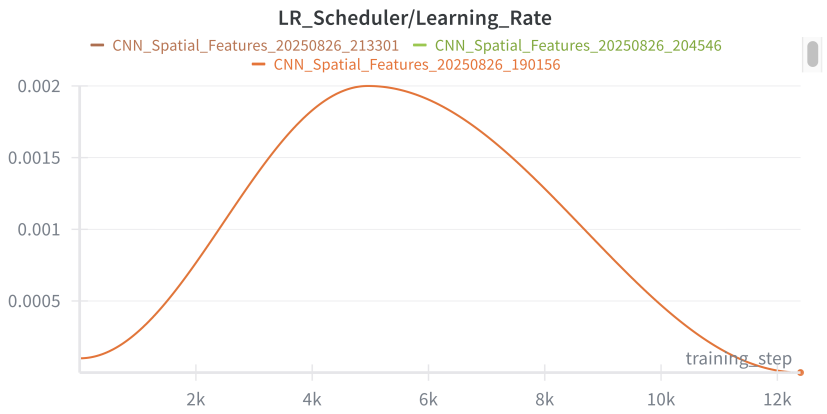

Figure 5.1: Learning-rate schedule for autoencoder pre-training (OneCycle): gradual warm-up to a peak rate followed by monotonic anneal to a small final rate.

5.3 Transformer Training for Latent Mapping

This stage learns to map frequency-domain measurements into the spatial latent established in §5.2. The decoder remains frozen so learning focuses on the *measurement encoder*—the Stage 2 module comprising the measurement/coordinate embedding, transformer encoder blocks, and the multi-query pooling head that outputs the scan-level code $\hat{z}$. Input standardisation and the fixed 256-token sequence follow the data pipeline in §3.4.2.

5.3.1 Objective and Teacher–Student Setup

Let $z \in \mathbb{R}^{256}$ be the teacher latent from the frozen Stage 1 encoder on the target volume, and $\hat{z} \in \mathbb{R}^{256}$ the student latent from measurements. Training minimises the latent RMSE

$$\mathcal{L}_{\text{latent}} = \sqrt{\frac{1}{256}\sum_{j=1}^{256}\left(\hat{z}_j - z_j\right)^2}. \tag{5.2}$$

This teacher–student design recasts Stage 2 as 256-dimensional regression rather than full-volume reconstruction (2×64^3 voxels), reducing variance, improving sample efficiency, and avoiding the instability of decoding during learning. With the decoder fixed, capacity concentrates on the measurement encoder (embedding → transformer blocks → multi-query pooling), and the learned code stays anchored to the Stage 1 spatial prior. End-to-end reconstructions are computed only for validation and model selection; selection/early stopping is centralised in §5.4.

5.3.2 Optimisation Schedule and Regularisation

Parameters are optimised with AdamW [30]. The learning rate uses a linear warm-up followed by cosine decay, improving early stability and then annealing smoothly as attention layers mature [18, 32]. Dropout within attention and feed-forward blocks provides mild regularisation; weight decay is applied to non-normalisation weights; and global gradient-norm clipping prevents occasional spikes. An exponential moving average (EMA) of trainable weights is *enabled by default* (unless stated otherwise in ablations), maintaining a temporally smoothed copy of the model and yielding steadier validation estimates; Chapter 6 quantifies the effect of EMA alongside other toggles. The resulting warm-up ramp and cosine decay used throughout Stage 2 are shown in Fig. 5.2.

Figure 5.2: Learning-rate schedule for latent mapping: linear warm-up followed by cosine decay.

With these Stage 2 choices in place, the shared protocol for batching, precision, tracking, and model selection is consolidated in §5.4.

5.4 Experimental Protocol and Infrastructure

This section fixes the shared controls that keep results comparable across stages and interpretable in Chapter 6: a single data protocol, consistent experiment tracking, and a common precision/hardware setup.

5.4.1 Protocol and Tracking

We adopt one fixed split of the 10k-phantom corpus (8k/1k/1k train/validation/test) and keep it unchanged across both stages, so the transformer is evaluated on the same cases as the autoencoder. Each experiment is repeated under three independent global seeds—1337, 28, and 1994—and Chapter 6 reports the mean and standard deviation across repeats. Evaluation is deterministic: the same seed initialises all random number generators; samples are processed in a fixed order; and, for measurement sequences, the fixed 256-subset at validation/test follows §3.4.2. A single dataloader is reused for all runs (training with shuffling/augmentation; validation/test without), ensuring identical preprocessing and a fair basis for early-stopping comparisons.

Weights & Biases serves as the primary tracker. Metrics are logged at the appropriate cadence—per *batch* for fast-changing quantities such as learning rate and latent RMSE, and per *epoch* for validation losses—and key artefacts are exported for analysis and for figures in Chapters 5–6. Model selection is based on the *raw* voxel-space RMSE computed end-to-end on the validation set for both stages. Early stopping monitors the same raw RMSE with a patience of 50 epochs; the retained checkpoint is the first epoch achieving the minimum validation RMSE. These choices ensure that optimisation (in a standardised space) is ultimately judged against clinically interpretable units.

5.4.2 Precision and Hardware

Both stages use automatic mixed precision (AMP) to reduce memory traffic and increase throughput: most operations run in FP16, key accumulations remain in FP32, and automatic loss scaling handles rare overflows [33]. Batch size is fixed at 128 across seeds and stages. For Stage 1, 3D convolutions use channels-last memory format where beneficial; Stage 2 transformers use the default layout. All experiments run on a single NVIDIA H200 NVL (140 GB VRAM) hosted on an Intel Xeon® 6747P machine with 258 GB RAM, using PyTorch with CUDA 12.6.

With the protocol and infrastructure fixed, we now turn to empirical results obtained under these shared controls. Chapter 6 presents Stage 1 autoencoder pre-training and Stage 2 transformer performance, moving from latent alignment to end-to-end voxel-space metrics and representative reconstructions.

CHAPTER 6

Experimental Evaluation and Results

6.1 Evaluation Protocol and Metrics

Before reporting numbers, this opening section fixes how results are produced and read so subsequent figures and tables are immediately interpretable. Full protocol details appear in §5.4; here we specify what is evaluated, how numbers are aggregated, and which signals drive model selection.

6.1.1 Experiments and Aggregation

Across this chapter we evaluate *Stage 1* (autoencoder) and *Stage 2* (transformer with a frozen decoder) under three independent global seeds. For each seed, the checkpoint used for reporting is the one attaining the minimum *raw* voxel-space RMSE on the validation set. All other metrics are computed at that same epoch, and results are then aggregated as *mean* $\pm$ *sd* across seeds. In Stage 2 only, validation uses an exponential moving average (EMA) of the transformer's trainable weights; Stage 1 omits EMA because its curves are already smooth and low-variance. Logging matches signal timescales: fast training signals (e.g., learning rate; Stage 2 latent RMSE/cosine) are logged *per batch*, while validation metrics are logged *per epoch*; plots therefore show dense training traces with clearly marked validation points.

6.1.2 Metrics

We report voxel-space fidelity in *raw physical units* and, for Stage 2, complementary latent-alignment diagnostics. Detailed formulas and the fixed mask/ROI construction are in Appendix A.1.

- **RMSE (primary).** Overall voxel error across both channels ("total RMSE"; Eq. (A.1)) is our selection signal (mm^{-1}; lower is better). We also report *per-channel* RMSE (Eq. (A.2))—RMSE_{μ_a} and $\mathrm{RMSE}_{\mu_s'}$—to localise where residual error concentrates; magnitudes are not directly comparable across channels because their physical ranges differ.
- **Sørensen–Dice (overlap).** Dice quantifies recovery of high-absorption/scattering *regions of interest*. Masks are formed by *batch-wide, per-channel* min–max normalisation (over the current evaluation batch) and a fixed 0.5 threshold (Eq. (A.3)); we report the unweighted mean across channels (Eq. (A.4)). Values lie in $[0, 1]$; higher is better.

- **Contrast ratio (separation).** Contrast assesses target/background separation as the ratio of inside/outside means, using ROIs defined from the ground truth via *batch-wide, per-channel* min–max normalisation and a fixed 0.5 threshold (Eq. (A.5)); we report the predicted-to-true contrast ratio averaged across channels (Eq. (A.6)). Values near 1 indicate preserved separation; < 1 under-contrast; > 1 over-contrast.
- **Latent diagnostics (Stage 2 only).** Teacher–student *latent RMSE* (Eq. (A.7); lower is better) and *cosine similarity* (Eq. (A.8); higher is better) characterise alignment between the predicted latent and the Stage 1 teacher. These do not drive checkpoint selection.

With the protocol and metrics fixed, we begin by assessing Stage 1 to confirm a stable spatial prior, then turn to Stage 2's measurement-to-latent mapping under the frozen encoder–decoder.

6.2 Stage 1: Autoencoder Pre-Training Results

Stage 1 pre-trains the convolutional autoencoder that later serves as a fixed backbone in Stage 2. Our aims are to show that it (i) learns stably with low seed-to-seed variance, (ii) achieves strong quantitative fidelity on validation data, and (iii) reconstructs representative phantoms cleanly across both channels. We first discuss learning dynamics and validation signals, then summarise seed-wise metrics, and finally show qualitative slices.

6.2.1 Learning Dynamics

Learning progresses in two phases. During warm-up the optimiser rapidly reduces the standardised training error; thereafter improvements are steady, with near-overlapping traces across seeds indicating low optimisation variance. The validation RMSE (Eq. A.1) mirrors this trajectory without late-epoch drift (Fig. 6.1); the curve shape matches the OneCycle warm-up/anneal schedule in Fig. 5.1.

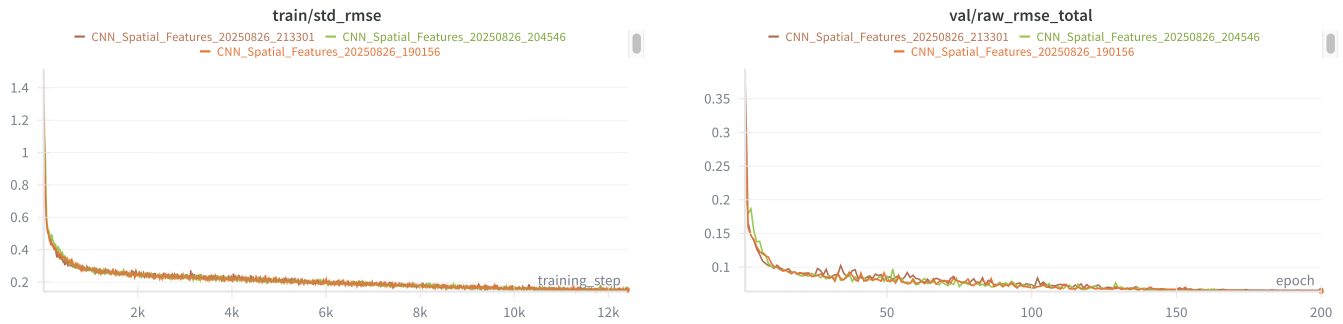

Figure 6.1: Stage 1 learning curves (three-seed overlays). **Left:** training *standardised* RMSE (per batch). **Right:** *raw* validation RMSE (per epoch; selection metric). The early drop and smooth late plateau match the warm-up/anneal phases in Fig. 5.1.

Validation overlap and separation metrics tell the same story: Dice rises from near-chance to ≈0.7 and stabilises, while the contrast ratio increases monotonically to ≈0.88 with only small oscillations. See Fig. 6.2. Together with Fig. 6.1, these traces indicate a reliable spatial prior and smooth convergence.

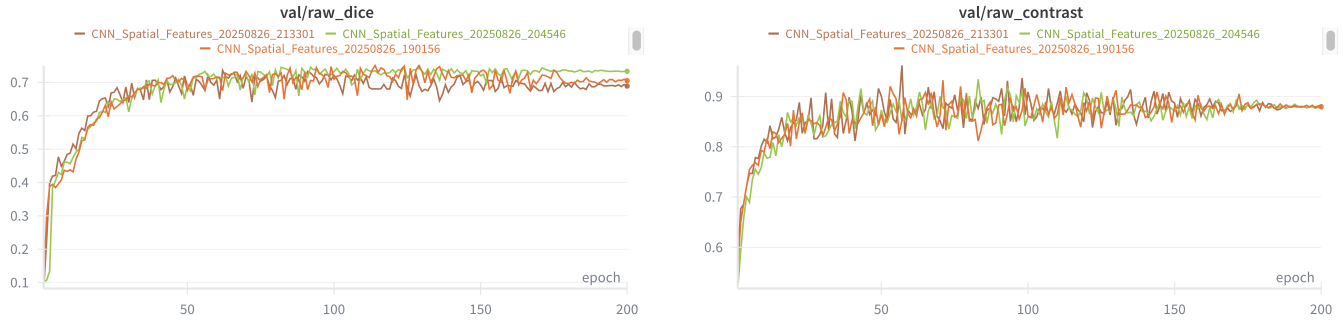

Figure 6.2: Stage 1 validation signals (three-seed overlays). **Left:** overall Sørensen–Dice coefficient. **Right:** overall contrast ratio. Both rise quickly and then stabilise, mirroring the RMSE behaviour.

6.2.2 Quantitative Summary

Table 6.1 reports validation metrics at the seed-selected checkpoints. Total voxel error is low ($\approx$0.064 mm^{-1}) with a tight spread. Channel-wise RMSEs differ because the physical ranges differ: μ_a is small (order 10^{-3}–10^{-2} mm^{-1}), whereas μ_s' is larger (order 10^{-1} mm^{-1}). Consequently, μ_s' dominates absolute RMSE while μ_a errors remain tiny. Dice around 0.71 and high contrast ($\approx$0.88) confirm clean target delineation and strong separation.

Table 6.1: Validation metrics at the seed-selected checkpoints; mean ± sd aggregates three seeds. RMSE in mm^{-1}.

Metric	Seed A (1337)	Seed B (28)	Seed C (1994)	Mean ± sd
$\text{RMSE}_{\text{total}}$	0.06380	0.06393	0.06542	**0.06438 ± 0.00090**
RMSE_{μ_a}	0.00067	0.00067	0.00068	$6.73\times10^{-4} \pm 5.77\times10^{-6}$
$\text{RMSE}_{\mu_s'}$	0.09022	0.09040	0.09252	0.09105 ± 0.00128
$\text{Dice}_{\text{total}}$	0.70528	0.73358	0.68906	0.7093 ± 0.0225
$\text{Contrast}_{\text{total}}$	0.87990	0.87950	0.87985	0.8798 ± 0.0002

6.2.3 Qualitative Reconstructions

Figure 6.3 shows XY slices from the *held-out test set* (9001, 9002) for both channels, rendered by the trained Stage 1 autoencoder in inference mode (the test split is never used for training or selection). For fair visual comparison, each channel is displayed with a fixed, channel-specific intensity window (mm^{-1}) derived from the validation-set range; the same window is used for GT and AE, without per-image rescaling. Reconstructions preserve inclusion topology and relative intensities, backgrounds are smooth, and edges are only mildly softened, consistent with the moderate Dice. There is no systematic bias between channels or phantoms; i.e., neither μ_a vs. μ_s' nor 9001 vs. 9002 consistently appears easier or harder.

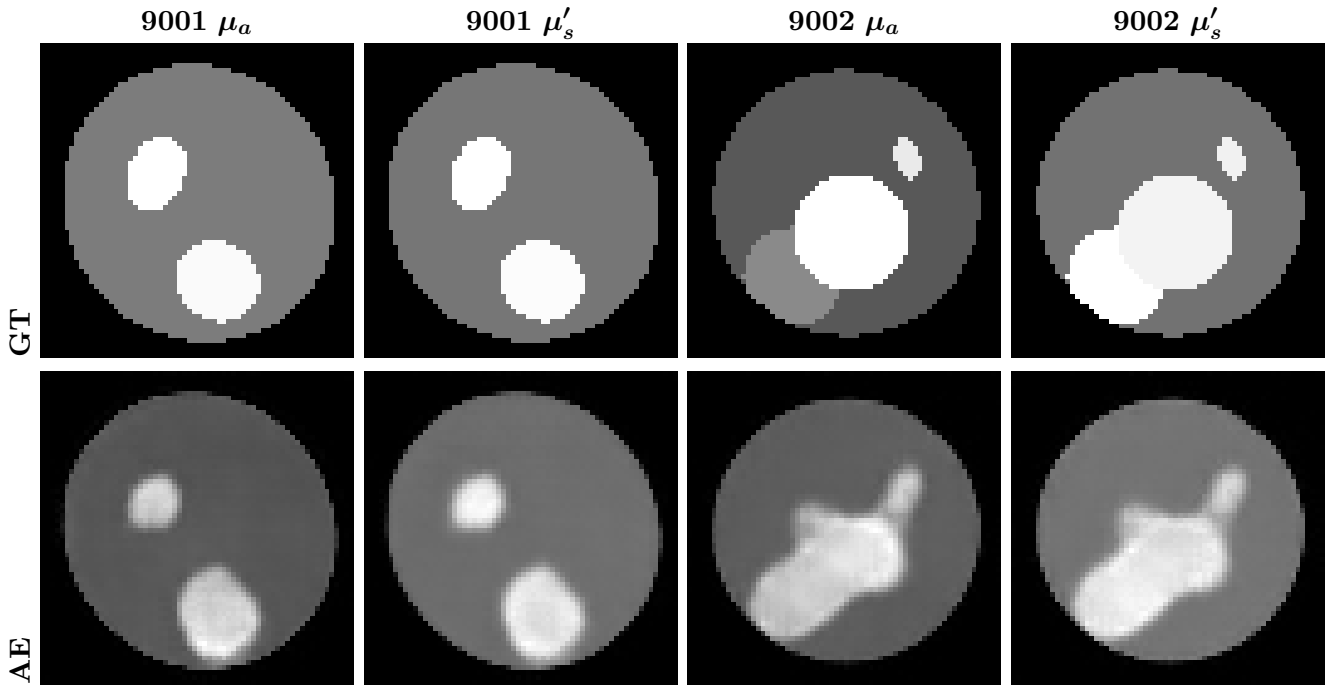

Figure 6.3: Qualitative Stage 1 reconstructions (*held-out test set*; XY slices). **Top:** ground truth (GT). **Bottom:** autoencoder (AE). Columns show phantoms 9001 and 9002 for both channels (absorption μ_a and reduced scattering μ'_s). Channel-specific intensity windows (mm^{-1}) are fixed across GT/AE for comparability.

In summary, Stage 1 delivers a stable, accurate autoencoder with low error, strong overlap/contrast, and visually clean reconstructions.

6.3 Stage 2: Transformer Results

Stage 2 trains a transformer to predict the *Stage 1 latent* directly from measurements while the Stage 1 encoder (teacher latent) and decoder remain frozen. We first examine latent alignment with the teacher, then report end-to-end validation behaviour (raw voxel-space metrics; EMA vs. non-EMA), and finally show qualitative reconstructions on the held-out test set.

6.3.1 Latent Alignment Dynamics

As training proceeds, latent alignment strengthens steadily across seeds. The epoch-averaged latent RMSE drops quickly during the first few thousand steps and then continues to improve at a slower but persistent rate; in parallel, the epoch cosine similarity rises monotonically from near chance to ~ 0.85 by the end of training (Fig. 6.4). Together these traces indicate consistent progress in mapping measurements into the teacher space; early stopping is triggered by validation criteria rather than saturation of the latent signals.

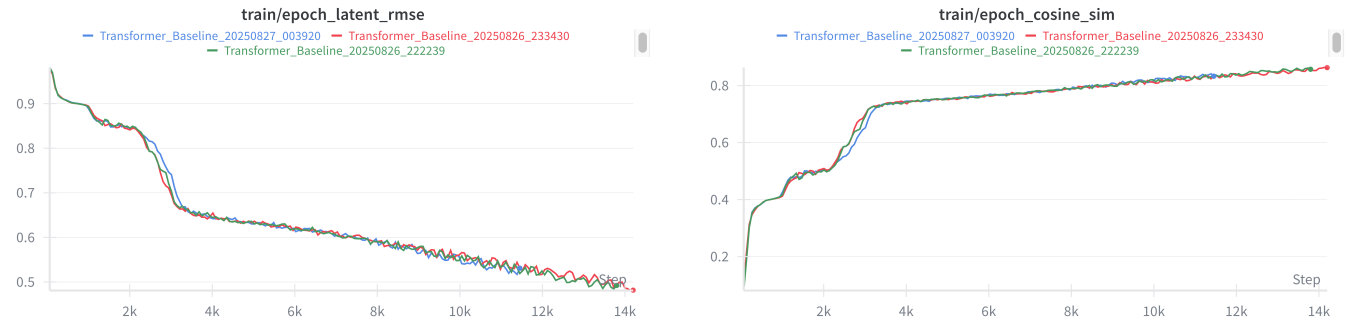

Figure 6.4: Stage 2 latent alignment (epoch traces; three seeds). **Left:** latent RMSE decreases throughout training. **Right:** latent cosine similarity increases steadily toward ≈ 0.85.

Two auxiliary signals help interpret these trends. *Student latent magnitude* (the ℓ_2 norm of the predicted latent) rises over training and approaches, but remains below, the teacher's approximately constant magnitude (≈ 15.7 across runs). This indicates a mild *scale bias*: the student progressively matches the teacher's *direction* (high cosine) while still under-estimating its *length*; continued training would likely reduce the gap further, but selection halts when validation no longer improves. *Attention entropy* (Shannon entropy of the *transformer's self-attention* weight distributions, averaged over heads/layers on validation) decreases smoothly, indicating attention that is less diffuse and more concentrated on a small set of informative tokens/positions—an expected correlate of improving latent alignment.

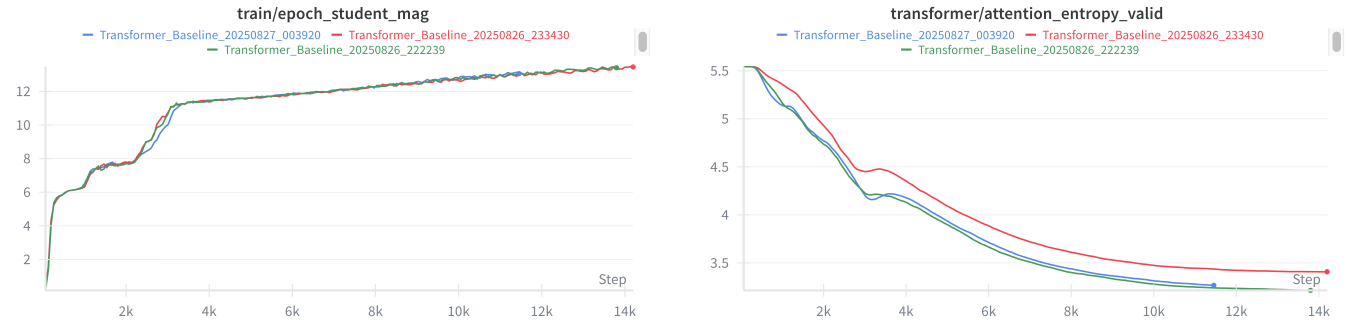

Figure 6.5: Supporting signals (epoch). **Left:** student latent magnitude increases throughout training, narrowing the gap to the teacher. **Right:** validation attention entropy decreases, indicating progressively sharper focus.

6.3.2 Validation Through the Frozen Decoder

We now decode the predicted latent with the frozen Stage 1 decoder and track raw voxel-space metrics on validation. The total raw RMSE shows a rapid early drop, then slows and flattens, entering a clear plateau by ∼10k steps; the per-channel curves follow the same shape and reflect the differing physical ranges, with μ_s' dominating absolute error and μ_a remaining in the $10^{-3}\,\mathrm{mm}^{-1}$ band. The plateau marks the start of the early-stopping *patience* window; training continues for roughly 50 further epochs without improvement before early stopping actually fires. Comparing EMA vs. non-EMA indicates a small but consistent benefit from EMA: across seeds, EMA total raw RMSE averages 0.13418 ± 0.00034 mm^{-1} whereas the non-EMA counterpart averages 0.13821 ± 0.00399 mm^{-1}

(mean±sd), i.e., a ~ 0.004 absolute gain with markedly lower variance.

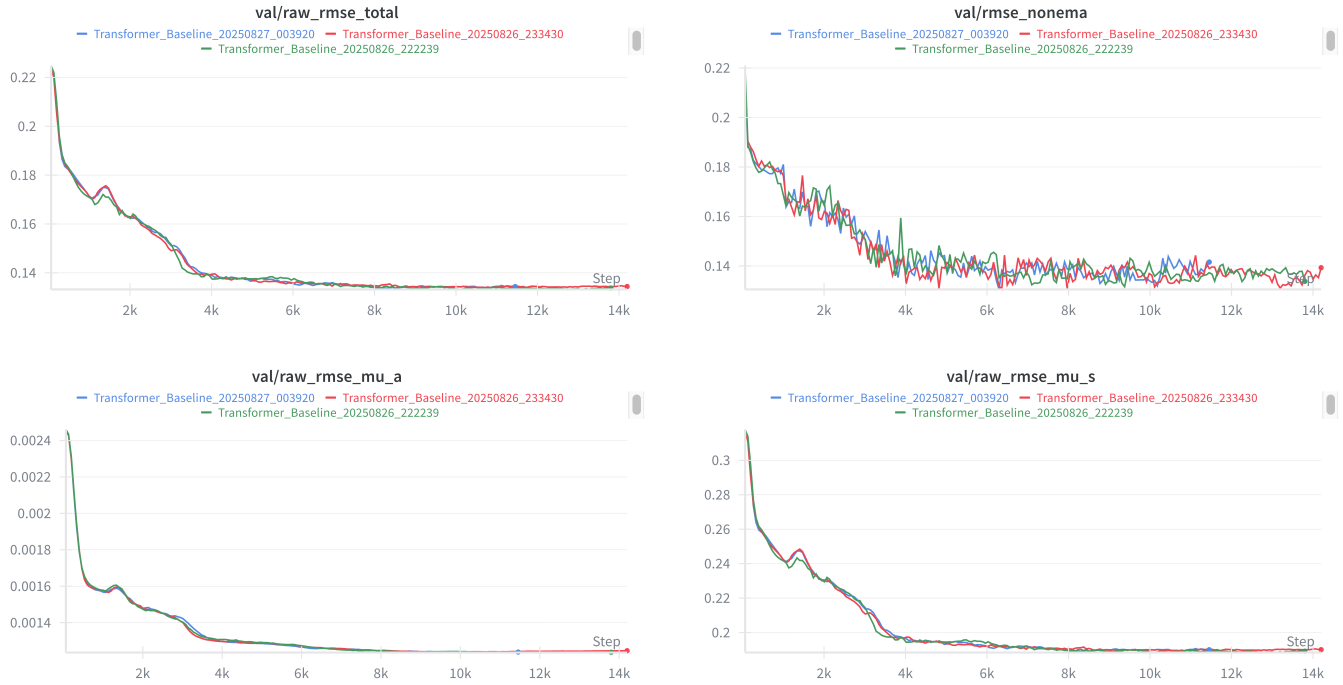

Figure 6.6: Validation curves (epoch; three seeds). **Top left:** total raw RMSE with EMA (selection signal). **Top right:** total raw RMSE without EMA. **Bottom:** per-channel raw RMSE with EMA (μ_a left; μ_s' right).

6.3.3 Quantitative Summary

Table 6.2 reports validation metrics for each seed together with mean±sd aggregates. Total raw RMSE clusters tightly around 0.134 mm^{-1}; overall Dice is modest (~ 0.31) while overall contrast is high (~ 0.613), i.e., targets are well separated from background but spatial overlap is limited.

Table 6.2: Stage 2 validation metrics. RMSE units: mm^{-1}.

Metric	Seed A (1337)	Seed B (28)	Seed C (1994)	Mean ± sd
$\text{RMSE}_{\text{total}}$	0.13378	0.13439	0.13436	**0.13418 ± 0.00034**
RMSE_{μ_a}	0.00124	0.00125	0.00124	0.001243 ± 0.000006
$\text{RMSE}_{\mu_s'}$	0.18918	0.19005	0.19001	0.18975 ± 0.00049
$\text{Dice}_{\text{total}}$	0.31094	0.31141	0.31147	**0.31127 ± 0.00029**
$\text{Contrast}_{\text{total}}$	0.61320	0.61277	0.61226	**0.61274 ± 0.00047**
$\text{RMSE}_{\text{non-EMA}}$	0.13379	0.13928	0.14156	0.13821 ± 0.00399

6.3.4 Qualitative Reconstructions

Figure 6.7 shows XY slices from the *held-out test set* (never used for training or model selection) for two representative phantoms (9001, 9002) and both channels. Visually, Stage 2 preserves global contrast and approximate target locations but tends to collapse multiple inclusions into a single

broad hotspot with softened boundaries. For **9001** (both μ_a and μ_s'), the two GT inclusions are represented by one dominant, blurred region with only a hint of secondary structure; background is smooth but edges are diffuse. For **9002**, the prediction similarly merges the GT inclusions into an elongated bright area, again with strong intensity separation but limited shape fidelity. The behaviour is consistent across channels: μ_s' exhibits slightly higher apparent contrast yet the same smoothing of boundaries, matching the quantitative pattern of high contrast but modest Dice. Tiles use fixed, channel-specific intensity windows in raw units (mm^{-1}), applied identically to GT and predictions for fair visual comparison.

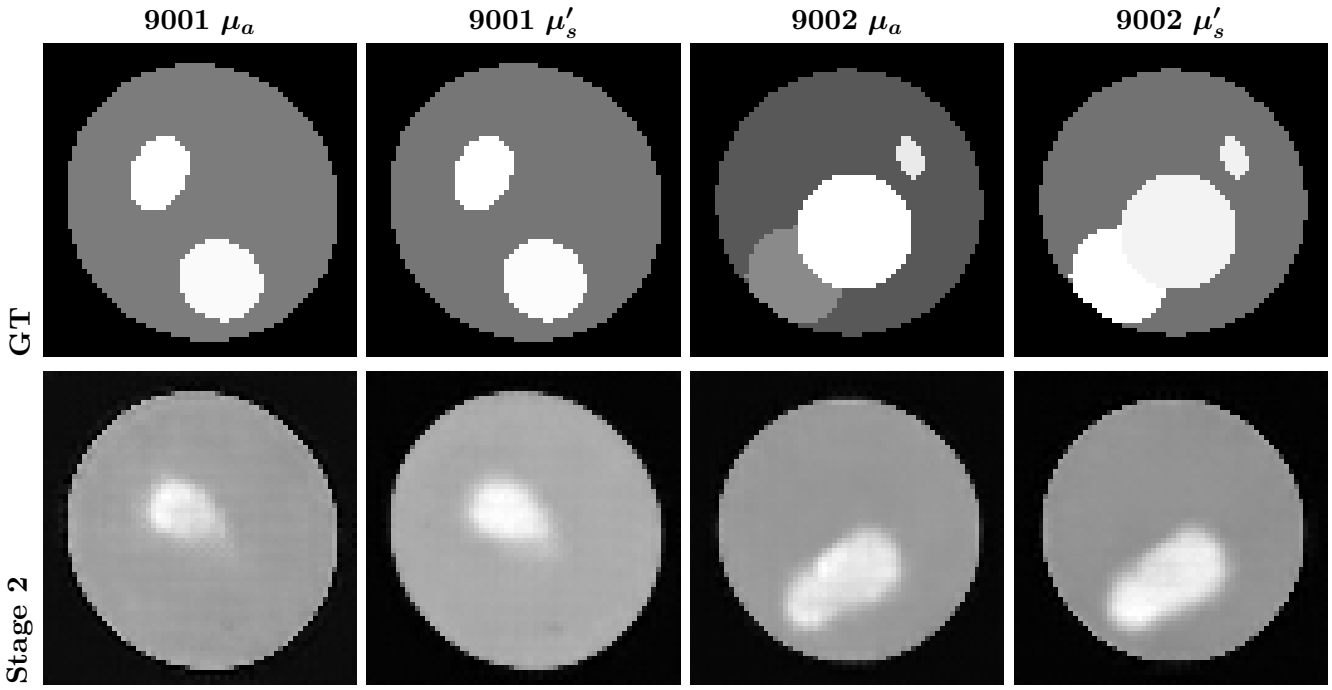

Figure 6.7: Qualitative Stage 2 reconstructions (*held-out test set*; XY slices). Columns are phantom/channel pairs; rows show ground truth and Stage 2 predictions decoded by the frozen decoder.

6.3.5 Summary and Transition

Across three seeds, Stage 1 yields a stable, high-fidelity spatial prior. Stage 2 maps measurements into this latent space with steadily improving alignment; validation curves plateau cleanly, EMA provides a small consistent gain, and decoding through the frozen decoder gives low total raw RMSE and high contrast, albeit with softened boundaries and modest Dice. The next chapter examines these findings in depth—diagnosing the smoothing, considering geometry variability, and outlining practical remedies and extensions.

CHAPTER 7

Discussion, Limitations, and Conclusions

7.1 Revisiting the Research Gap and What We Showed

This work tested a concrete hypothesis: combining (i) broad phantom/probe diversity, (ii) *systematic* geometry randomisation, and (iii) a hybrid, geometry-aware CNN–Transformer improves robustness under probe-layout variation, anatomical variability, measurement noise, and the synthetic-to-real gap. We aimed for models that *generalise under geometry shift*—when the source–detector (SD) layout departs from any canonical probe—and under anatomical variation. The goal was not to beat prior voxel-space baselines outright, but to show these ingredients *together* yield a model that maintains performance across SD layouts and tissue shapes without per-layout tuning or case-specific re-tuning.

Chapters 1–2 motivated this gap and specified the ingredients; Chapters 3–6 instantiated them in a two-stage protocol. We operationalise "generalise reliably under geometry shift" via three criteria: (a) no per-layout fine-tuning or calibration; (b) tight seed-wise variance under a fixed selection protocol; and (c) preservation of voxel-space fidelity and target/background separation when decoded by the frozen Stage 1 decoder. These criteria are quantified in Chapter 6 and interpreted in §7.2.

Relative to Dale *et al.* [8, 19], our evaluation domain is deliberately harder. Like that work, we encode geometry to support path-agnostic operation, but Dale's probe layout was fixed and the phantom family less varied (targets on a cuboidal domain), whereas our phantoms occupy an ellipsoidal tissue region within an air box with more complex boundaries and wider SD layouts. Consequently, absolute scores are not directly comparable: we accept that broader geometry/phantom variation can depress headline numbers while providing a stronger test of layout-agnostic behaviour. We therefore do *not* claim superiority over fixed-geometry training or over Dale's reported metrics (the protocols differ, and no controlled fixed-geometry baseline was run here). What we do claim—and substantiate—is a robust, layout-agnostic pipeline whose performance remains stable as SD layouts vary within a strictly controlled, reproducible protocol. The sections that follow interpret these outcomes, acknowledge limitations, and outline practical implications and extensions.

7.2 Discussion of Results

Stage 1 establishes a trustworthy spatial prior. Training and validation traces are smooth across seeds (Fig. 6.1), and the overlap/separation signals rise and stabilise (Fig. 6.2). At the seed-selected

checkpoints (Table 6.1) and on held-out test slices (Fig. 6.3), reconstructions are clean and stable for both channels. Practically, the decoder–latent "contract" is reliable: Stage 2 can focus on predicting the latent rather than compensating for decoder drift.

Stage 2 behaves as expected under variable probe layouts and tissue shapes. Latent alignment improves steadily (latent RMSE falls; cosine rises; Fig. 6.4); the student latent's norm moves toward the teacher's (teacher magnitude ≈ 15.7), and attention becomes more focused (Fig. 6.5). When decoded by the frozen Stage 1 decoder, validation curves plateau (Fig. 6.6), so model selection is well defined and repeatable across seeds (Table 6.2). EMA gives a small, consistent gain with lower variance, and we keep multi-query attention pooling as the default since it yielded stronger validation behaviour and cleaner reconstructions than global mean pooling in this protocol.

The end-to-end fidelity is good but not yet excellent. Stage 2's total *raw* RMSE clusters near 0.134 mm^{-1} (Table 6.2), higher than Stage 1's autoencoding error (≈ 0.064 mm^{-1}; Table 6.1). A gap is expected—Stage 2 infers the latent from measurements under layout variability—but it indicates residual loss in the measurement $\rightarrow$ latent step. Per-channel behaviour follows physical scales: μ_s' dominates absolute error due to its larger dynamic range, while μ_a remains small in raw units. Complementary metrics align: high contrast (~ 0.613 overall) with modest Dice (~ 0.31) means targets are clearly separated but edges are softened.

The qualitative test slices (Fig. 6.7) match these numbers: nearby inclusions can merge into a hotspot with diffuse boundaries. Three factors plausibly contribute. *First*, latent-only supervision through a *frozen* decoder favours latents that decode to globally plausible volumes, which helps coarse structure and separation but under-constrains fine detail. *Second*, the student latent's norm stays slightly below the teacher's (Fig. 6.5); this can under-estimate peak intensities, so thresholded ROIs become smaller and Dice falls even when contrast is strong. *Third*, the protocol is intentionally hard—diverse tissue shapes, randomised SD layouts, and $\sim$256 measurements per phantom—so we trade a little best-case accuracy for *consistency across layouts*; with fewer, geometry-spread paths, the model sees less redundancy to tease apart inclusions.

Two final points. (i) Across all three seeds we observe the same pattern—rapid early improvement (loss $\downarrow$, cosine $\uparrow$) followed by a stable plateau—indicating these behaviours reflect the design rather than noise. (ii) While multi-query pooling helps stability, the main ceilings are inferring a precise latent under layout variability and decoding with a fixed prior that emphasises global plausibility over sharp boundaries. In short, the system is robust to geometry and tissue-shape variability and produces globally faithful reconstructions with strong target–background separation, at the cost of softened edges and occasional merging of inclusions. The next steps focus on sharpening boundaries and lowering voxel-space error without sacrificing layout robustness (§7.4).

7.3 Limitations and Practical Implications

The system is built for path-agnostic use: Stage 2 turns geometry-aware tokens (SD coordinates *and* measured signals) into a latent, and the frozen Stage 1 decoder renders a volume. Below we summarise how to run and read the model, and the main limits of Chapter 6.

1. **Path-agnostic operation with stable selection.** Because tokens include SD geometry and

signals, a *single* trained model handles varied layouts without per-layout tuning (Sec. 6.1). Training converges cleanly and early stopping is clear; three seeds land within a tight band at selection (Table 6.2), so checkpointing is reproducible and behaviour consistent across runs.

2. **How to read the outputs.** We see low total raw error but modest Dice (Table 6.2); visually, targets localise well but edges are soft with occasional merging (Fig. 6.7). These volumes suit fast localisation/triage and ROI proposal. If crisp boundaries are required, add a light refinement inside the ROI. Remember: RMSE is in raw units and is dominated by μ_s'; Dice and Contrast use the fixed thresholds/ROI rules in Appendix A.1. Thus "high Contrast with modest Dice" means good separation but softer edges—not mislocalisation. Results reflect a time-bounded prototype; boundary fidelity can plausibly be improved without changing the path-agnostic protocol.
3. **Inference recipe and simple checks.** Use ∼256 geometry-diverse SD paths per scan (Sec. 6.1); if possible, average two or three 256-path subsets to stabilise borderline cases without retraining. Before deployment, ensure SD coordinates are normalised as in training and signals (log-amplitude, phase) are preprocessed the same way. During *validation only*, latent diagnostics—cosine and student-norm trends in Fig. 6.4 and Fig. 6.5—help catch configuration errors; teacher latents are unavailable at runtime. Keep thresholds as in Appendix A.1 for comparability.
4. **External validity (simulation→real).** All results are simulated. Real tissue variability, hardware drift, calibration error, and noise may not be captured. Any clinical claims require prospective testing on real acquisitions; this study should be read as a strong proof of concept.
5. **Internal/practical limits (frozen decoder and coverage).** Freezing the Stage 1 decoder stabilises evaluation and isolates the measurement-to-latent step, but can cap boundary sharpness by under-constraining fine detail (Sec. 7.2). We did not test joint fine-tuning or partial unfreezing. Results also assume ∼256 paths per sample; with sparse or uneven coverage, nearby inclusions may merge (Fig. 6.7). Without retraining, mitigations include acquiring more geometry-diverse paths, averaging a few independent 256-path subsets at test time, or applying a high-resolution/physics-guided refinement inside the proposed ROI.

These bounds keep claims scoped to the *combined* protocol (Sec. 7.1). The next section outlines targeted extensions (e.g., tissue patches, latent-scale calibration, smarter token embedding, ROI refinement) to improve boundary fidelity without sacrificing robustness.

7.4 Future Directions

The current system is a time-bounded prototype that proved path-agnostic reconstruction can work in practice. Below are six focused extensions that stay true to this design while pushing boundary fidelity and overall accuracy (§7.3).

1. **Add local context via tissue patches.** For each measured path, attach two small tissue patches—centred on the source and on the detector—to the token that already carries SD coordinates and signals. The transformer then sees both the path and local tissue context at source and detector, which should help separate close inclusions and sharpen edges. We planned this but could not run it within the time budget; it remains the most direct next step.

2. **Smarter token embedding for signals + geometry.** Keep the two-branch MLPs (signals, positions), but add relative geometry cues (source–detector direction, distance) and a small set of sinusoidal position codes. Replace plain concat→MLP with light gated fusion (e.g., FiLM/SE) so geometry can modulate signals (and vice versa). Token size stays 256; the aim is to expose signal–geometry interactions pre-transformer and lift edge fidelity without touching the decoder.
3. **Match the latent scale to the teacher.** In Stage 2 the predicted latent vector tends to have a slightly smaller norm than the teacher's (Fig. 6.5). Simple fixes—e.g., a small "gain" parameter, a norm-matching penalty, or projecting the latent to a target norm—could remove this bias. Closing the scale gap should lift peak intensities and improve Dice without hurting contrast.
4. **Tune path count and how paths are chosen.** We used ∼256 paths per sample (per Dale's setup [8, 19]). Our domain is harder (varied layouts and tissue shapes), so test nearby counts (128/256/384/512) to map the speed–accuracy trade-off. Pick paths well spread in distance and angle (a stratified sampler suffices). At test time, averaging two independent 256-path subsets from the same scan is a no-retrain baseline that stabilises borderline cases.
5. **Lightweight ROI sharpening after decoding.** Keep the decoder frozen for stability, but add a tiny sharpening step only inside the proposed region of interest (ROI)—for example, a shallow high-resolution CNN or a fast physics-guided correction confined to the ROI. This preserves the decoder's global plausibility while cleaning edges where it matters.
6. **Narrow the sim→real gap.** Briefly fine-tune on a small real dataset (tens of scans) while keeping the protocol fixed. Also broaden the noise model (e.g., correlated sensor noise, calibration drift) so training better matches hardware. Optionally, at test time average predictions from two independent 256-path subsets to stabilise results.

These steps target the observed weaknesses (soft edges, occasional merging) without giving up the strengths we demonstrated—robust, layout-agnostic inference and reproducible selection.

7.5 Concluding Remarks

This dissertation tested whether *diverse data* (phantoms and probes), *systematic* geometry randomisation, and a *geometry-aware* hybrid architecture can deliver robust, path-agnostic DL-DOT. The evidence supports it: a single model handles varied SD layouts and tissue shapes without per-layout or per-case tuning, with stable selection across seeds and consistently high target–background separation. The main caveat is boundary fidelity—edges are softened and nearby inclusions can merge—seen as modest Dice despite strong contrast. This is a time-bounded prototype, not an endpoint. Next steps are concrete: add local context via tissue patches, match the latent scale, improve the signal + geometry embedding, tune the number and geometric spread of measurement paths, apply light ROI sharpening, and narrow the sim→real gap with usable confidence. Taken together, these extensions are practical and incremental; they target known weaknesses without sacrificing demonstrated strengths. We conclude with a clear blueprint—data pipeline, training protocol, and architecture—ready to be stress-tested on real acquisitions and iterated into clinically useful DL-DOT.

Bibliography

[1] Simon R. Arridge. Optical tomography in medical imaging. *Inverse Problems*, 15(2):R41–R93, 1999.

[2] Adam P. Gibson, Jeremy C. Hebden, and Simon R. Arridge. Recent advances in diffuse optical imaging. *Phys. Med. Biol.*, 50(4):R1–R43, 2005.

[3] Bruce J. Tromberg, Zheng Zhang, Anaïs Leproux, Thomas D. O'Sullivan, Albert E. Cerussi, Philip M. Carpenter, Rita S. Mehta, Darren Roblyer, Wei Yang, Keith D. Paulsen, et al. Predicting responses to neoadjuvant chemotherapy in breast cancer: ACRIN 6691 trial of diffuse optical spectroscopic imaging. *Cancer Res.*, 76(20):5933–5944, 2016.

[4] Adam T. Eggebrecht, Benjamin R. White, Silvina L. Ferradal, Cheng Chen, You Zhan, Abraham Z. Snyder, Hamid Dehghani, and Joseph P. Culver. Mapping distributed brain function and networks with diffuse optical tomography. *Nat. Photonics*, 8(6):448–454, 2014.

[5] Roy A. Stillwell and Thomas D. O'Sullivan. A real-time fully handheld frequency-domain near-infrared spectroscopy imaging system. In *Proc. SPIE Multiscale Imaging and Spectroscopy III*, volume 11944 of *Proc. SPIE*, page 119440D. SPIE, 2022.

[6] Robin Dale, Biao Zheng, Felipe Orihuela-Espina, Nicholas Ross, Thomas D. O'Sullivan, Scott Howard, and Hamid Dehghani. Deep learning-enabled high-speed, multi-parameter diffuse optical tomography. *J. Biomed. Opt.*, 29(7):076004, 2024.

[7] Hamid Dehghani, Matthew E. Eames, Phaneendra K. Yalavarthy, Sean C. Davis, Subhadra Srinivasan, Colin M. Carpenter, Brian W. Pogue, and Keith D. Paulsen. Near-infrared optical tomography using NIRFAST: algorithm for numerical model and image reconstruction. *Commun. Numer. Methods Eng.*, 25(6):711–732, 2009.

[8] Robin Dale, Nicholas Ross, Scott Howard, Thomas D. O'Sullivan, and Hamid Dehghani. Transformer-encoder for real-time DOT scanning. In *ECBO (European Conference on Biomedical Optics)*, 2025.

[9] David A. Boas, Daniel H. Brooks, Eric L. Miller, Charles A. DiMarzio, Misha Kilmer, Robert J. Gaudette, and Quan Zhang. Imaging the body with diffuse optical tomography. *IEEE Signal Process. Mag.*, 18(6):57–75, 2001.

[10] Simon R. Arridge and John C. Schotland. Optical tomography in medical imaging: theory, models, and applications. *Inverse Problems*, 25(12):123010, 2009.

[11] Tanja Tarvainen, Ville Kolehmainen, Jari P. Kaipio, and Simon R. Arridge. Corrections to linear methods for diffuse optical tomography using approximation error modelling. *Biomed. Opt. Express*, 1(1):209–222, 2010.

[12] Vishal Monga, Yuelong Li, and Yonina C. Eldar. Algorithm unrolling: interpretable, efficient deep learning for signal and image processing. *IEEE Signal Process. Mag.*, 38(2):18–44, 2021.

[13] Simon R. Arridge, Peter Maass, Ozan Öktem, and Carola-Bibiane Schönlieb. Solving inverse problems using data-driven models. *Acta Numer.*, 28:1–174, 2019.

[14] Jonas Adler and Ozan Öktem. Learned primal–dual reconstruction. *IEEE Trans. Med. Imaging*, 37(6):1322–1332, 2018.

[15] Xu Feng, Wei Chen, Long Wei, and Fei Gao. Deep learning-based image reconstruction for diffuse optical tomography. *Biomed. Opt. Express*, 11(11):6366–6381, 2020.

[16] Bin Deng, Hanxue Gu, Hongmin Zhu, Ken Chang, Katharina V. Hoebel, Jay B. Patel, Jayashree Kalpathy-Cramer, and Stefan A. Carp. FDUNet: Deep learning-based three-dimensional diffuse optical image reconstruction. *IEEE Trans. Med. Imaging*, 42(8):2439–2450, 2023.

[17] Nazish Murad, Min-Chun Pan, and Ya-Fen Hsu. Periodicnet: an end-to-end data-driven framework for diffuse optical imaging of breast cancer from noisy boundary data. *J. Biomed. Opt.*, 28(2):026001, 2023.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[19] Robin Dale. *Deep Learning for Flexible and Real-Time DOT (Diffuse Optical Tomography)*. PhD thesis, University of Birmingham, 2025.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[21] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[22] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *arXiv*, 2014.

[23] Geoffrey E. Hinton and Ruslan R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[24] Özgün Çiçek, Ahmed Abdulkadir, Sören S. Lienkamp, Thomas Brox, and Olaf Ronneberger. 3D UNet: Learning dense volumetric segmentation from sparse annotation. In *MICCAI*, 2016.

[25] Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts. *Distill*, 2016.

[26] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam R. Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *ICML*, 2019.

[27] Andrew Jaegle, Felipe Gimeno, Andrew Brock, Oriol Vinyals, Andrew Zisserman, et al. Perceiver: General perception with iterative attention. In *ICML*, 2021.

[28] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *arXiv*, 2014.

[29] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv*, 2015.

[30] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

[31] Leslie N. Smith. A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay. *arXiv*, 2018.

[32] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *ICLR*, 2017.

[33] Paulius Micikevicius, Sharan Narang, Jonah Alben, Greg Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and Hao Wu. Mixed precision training. In *ICLR*, 2018.

APPENDIX A

Supplementary Methods and Results

A.1 Loss Functions and Metrics

This section compiles the formulae used in Chapter 6. All voxel-space metrics are computed in *raw physical units* from inverse-standardised outputs (see Chapter 5). Throughout, let

- $c \in \{\mu_a, \mu_s'\}$ denote the channel (absorption, reduced scattering),
- $V^{(c)} \in \mathbb{R}^{64\times64\times64}$ the ground-truth volume,
- $\hat{V}^{(c)}$ the corresponding reconstruction,
- Ω the tissue-voxel set (mask) with size $|\Omega|$.

Unless stated, metrics are averaged over the evaluation set. For mask-based metrics, *min–max normalisation is computed per batch and per channel*, matching the evaluation code. We use small stabilisers $\varepsilon_{\mathrm{D}} = 10^{-6}$ (Dice) and $\varepsilon = 10^{-8}$ (ratios).

1. Total RMSE

Overall voxel error across both channels (units: mm^{-1}):

$$\mathrm{RMSE}_{\mathrm{total}} = \sqrt{\frac{1}{2|\Omega|}\sum_{c\in\{\mu_a,\mu_s'\}}\sum_{v\in\Omega}\left(\hat{V}_v^{(c)} - V_v^{(c)}\right)^2}. \tag{A.1}$$

2. Per-channel RMSE

Reconstruction error for a single channel c (mm^{-1}):

$$\mathrm{RMSE}_c = \sqrt{\frac{1}{|\Omega|}\sum_{v\in\Omega}\left(\hat{V}_v^{(c)} - V_v^{(c)}\right)^2}. \tag{A.2}$$

3. Sørensen–Dice

Measures *spatial overlap* between predicted and ground-truth *high-value regions*. In line with the implementation, for each channel we compute batch-wise min–max normalisation for GT and

prediction separately and then threshold both at 0.5:

$$\begin{aligned} m_c &= \min_{b,v} V^{(c)}_{b,v}, \quad M_c = \max_{b,v} V^{(c)}_{b,v}, \\ \hat{m}_c &= \min_{b,v} \hat{V}^{(c)}_{b,v}, \quad \hat{M}_c = \max_{b,v} \hat{V}^{(c)}_{b,v}, \\ \tilde{V}^{(c)} &= \frac{V^{(c)} - m_c}{M_c - m_c + \varepsilon}, \qquad \tilde{\hat{V}}^{(c)} = \frac{\hat{V}^{(c)} - \hat{m}_c}{\hat{M}_c - \hat{m}_c + \varepsilon}, \\ M^{(c)} &= \mathbf{1}\{\tilde{V}^{(c)} > 0.5\}, \qquad \hat{M}^{(c)} = \mathbf{1}\{\tilde{\hat{V}}^{(c)} > 0.5\}. \end{aligned} \tag{A.3}$$

Dice per channel is

$$\mathrm{Dice}_c = \frac{2 \sum_{v\in\Omega} M^{(c)}(v)\, \hat{M}^{(c)}(v) + \varepsilon_{\mathrm{D}}}{\sum_{v\in\Omega} M^{(c)}(v) + \sum_{v\in\Omega} \hat{M}^{(c)}(v) + \varepsilon_{\mathrm{D}}}.$$

The reported figure is the unweighted mean across channels:

$$\mathrm{Dice} = \tfrac{1}{2}\Big(\mathrm{Dice}_{\mu_a} + \mathrm{Dice}_{\mu'_s}\Big) \in [0, 1]. \tag{A.4}$$

Interpretation. Higher is better. Because GT and prediction are each normalised before thresholding, Dice reflects agreement of *where* high-intensity structure appears, not its absolute magnitude.

4. Contrast ratio

Assesses *target/background separation.* For each channel, define the ROI from the *ground truth only* using batch-wise min–max normalisation and a 0.5 threshold; let $M^{(c)}$ be the ROI and $\bar{M}^{(c)} = 1 - M^{(c)}$ its complement. For any $X^{(c)} \in \{\hat{V}^{(c)}, V^{(c)}\}$:

$$\mu^{(c)}_{\mathrm{in}}(X) = \frac{\sum_{v\in\Omega} M^{(c)}(v)\, X^{(c)}_v}{\sum_{v\in\Omega} M^{(c)}(v) + \varepsilon}, \qquad \mu^{(c)}_{\mathrm{out}}(X) = \frac{\sum_{v\in\Omega} \bar{M}^{(c)}(v)\, X^{(c)}_v}{\sum_{v\in\Omega} \bar{M}^{(c)}(v) + \varepsilon}. \tag{A.5}$$

The channel-wise contrast is the *ratio of means* inside vs. outside; the reported metric is the ratio of predicted-to-true contrast, averaged across channels:

$$\mathrm{Contrast} = \frac{1}{2} \sum_{c\in\{\mu_a,\mu'_s\}} \frac{\dfrac{\mu^{(c)}_{\mathrm{in}}(\hat{V})}{\mu^{(c)}_{\mathrm{out}}(\hat{V}) + \varepsilon}}{\dfrac{\mu^{(c)}_{\mathrm{in}}(V)}{\mu^{(c)}_{\mathrm{out}}(V) + \varepsilon} + \varepsilon}. \tag{A.6}$$

Interpretation. Contrast $= 1$ means the model preserves the target/background amplitude ratio; < 1 indicates under-contrast, > 1 over-contrast. With positive-valued volumes this metric lives on $\mathbb{R}^+$ (i.e., it is not bounded by 1).

5. Latent RMSE

Discrepancy between teacher and student latents ($z, \hat{z} \in \mathbb{R}^{256}$):

$$\mathrm{LatentRMSE} = \sqrt{\frac{1}{256} \sum_{j=1}^{256} \Big(\hat{z}_j - z_j\Big)^2} \;\geq 0. \tag{A.7}$$

6. Cosine similarity

Directional alignment between student and teacher latents:

$$\text{CosSim} = \frac{\langle \hat{z},\, z \rangle}{\|\hat{z}\|_2 \, \|z\|_2} \in [-1, 1]. \tag{A.8}$$

Interpretation. Values near 1 indicate strong alignment (similar direction) even if magnitudes differ; values near $0/-1$ indicate orthogonal/opposed codes.

APPENDIX B

Codebase and Reproducibility Guide

B.1 Project and Repository Overview

This section gives the folder layout, main entry points, and repository locations so a reader can run the project from the repository root.

Root: `mah422/` **Code footprint:** 26 Python files, 14,199 LOC (in `code/` only).

```
mah422/
|-- code/                       # Core implementation (~14k LOC, 26 files)
|   |-- data_processing/        # Phantom generation & data loading
|   |-- models/                 # CNN, Transformer, hybrid architectures
|   |-- training/               # Stage 1/2 trainers & config
|   `-- utils/                  # Metrics, standardisation, logging
|-- setup/                      # Environment setup & deployment scripts
|-- nirfaster-FF/               # External forward-modelling library
|-- checkpoints/                # Saved model states
|-- logs/                       # Training/data logs
`-- data/                       # Generated datasets (HDF5)
```

Entry points:

- Data generation: `python -m code.data_processing.data_simulator`
- Training (Stage 1/2): `python -m code.training.train_hybrid_model`
  Select the stage via `CURRENT_TRAINING_STAGE` in `code/training/training_config.py`.
- Dataset analysis: `python -m code.data_processing.phantom_dataset_analysis`

Repository reference:

- GitLab: `https://git.cs.bham.ac.uk/projects-2024-25/mah422`
- GitHub: `https://github.com/maxhartml/mah422`
- Main branch: `main` Total commits: **215**

B.2 Environment and Platform Setup

Create a lightweight virtual environment and install dependencies listed in `setup/requirements.txt`.

```
python3 -m venv nir-dot-env
source nir-dot-env/bin/activate          # Windows: nir-dot-env\Scripts\activate
pip install -r setup/requirements.txt
```

Training was performed on a single NVIDIA H200 via Vast.ai; any recent CUDA-capable GPU works. For platform-specific setup of the forward solver, follow the NIRFASTer-FF instructions: `https://github.com/milabuob/nirfaster-FF`. When using a remote host, keep jobs resilient with a session manager (e.g., `tmux new -s training`) and sync checkpoints from `checkpoints/` as needed.

B.3 Data Generation, Training, and Outputs

Data generation. Generate synthetic datasets for training and validation using the simulator.

```
python -m code.data_processing.data_simulator
```

Key parameters can be tuned: dataset size, volume shape (64^3), number of measurements, and the master seed.

Outputs.

```
data/
‘-- phantom_001/
    |-- phantom_001_scan.h5      # measurements + ground truth
    ‘-- probe_001.png            # probe layout visualisation
```

A master seed in the simulator yields deterministic per-phantom seeds for repeatability.

Training. Launch Stage 1 (autoencoder pretraining) and Stage 2 (transformer→latent→frozen decoder) from the same entry point.

```
# Stage 1
#   set CURRENT_TRAINING_STAGE="stage1" in code/training/training_config.py
python -m code.training.train_hybrid_model


# Stage 2
#   set CURRENT_TRAINING_STAGE="stage2"
python -m code.training.train_hybrid_model
```

Typical controls: epochs, batch size, OneCycleLR (base/peak LR, warm-up), weight decay, early stopping. Latent alignment in Stage 2 uses RMSE in a 256-D code space; EMA is optional.

Artifacts and logs. Outputs are written during data generation and training to:

- **Checkpoints**
  `checkpoints/checkpoint_stage1_.pt`
  `checkpoints/checkpoint_stage2_.pt`
- **Logs**
  `logs/training/`
  `logs/data_processing/`
- **Experiment tracking (optional)**
  Weights & Biases project `nir-dot-reconstruction`

B.4 Quickstart Recipes

Copy–paste recipes for common workflows.

Local CPU demo.

```
python3 -m venv nir-dot-env && source nir-dot-env/bin/activate
pip install -r setup/requirements.txt
python -m code.data_processing.data_simulator                  # small dataset
# In code/training/training_config.py:
#   set EPOCHS_STAGE1=5 and CURRENT_TRAINING_STAGE="stage1"
python -m code.training.train_hybrid_model
```

Single-GPU training.

```
python -m code.data_processing.data_simulator
# Stage 1
#   CURRENT_TRAINING_STAGE="stage1"
python -m code.training.train_hybrid_model
# Stage 2
#   CURRENT_TRAINING_STAGE="stage2"
python -m code.training.train_hybrid_model
```

Remote host (e.g., Vast.ai).

```
ssh -p <PORT> <USER>@<IP>
git clone https://github.com/maxhartml/mah422.git && cd mah422
python3 -m venv venv && source venv/bin/activate
pip install -r setup/requirements.txt
tmux new -s training
python -m code.training.train_hybrid_model
```