SIMEIT: A SCALABLE SIMULATION FRAMEWORK FOR GENERATING LARGE-SCALE ELECTRICAL IMPEDANCE TOMOGRAPHY DATASETS

Ayman A. Ameen[⋆] *Franziska Mathis-Ullrich*[⋆] *Bernhard Kainz*[⋆†]

[⋆] Friedrich-Alexander University Erlangen-Nürnberg
[†] Imperial College London

ABSTRACT

Electrical Impedance Tomography (EIT) offers advantages over conventional imaging methods such as X-ray and MRI but suffers from an ill-posed inverse problem. Deep learning can alleviate this challenge, yet progress is limited by the lack of large, diverse, and reproducible datasets. We present **SimEIT**, a scalable framework for deterministic simulation and generation of synthetic EIT data. SimEIT enables high-throughput creation of diverse geometries and conductivity maps using parallelized finite element simulations, reproducible seeding, and automated validation. The framework provides multi-resolution, AI-ready HDF5 outputs with PyTorch integration. Demonstrated on two datasets exceeding 100,000 samples, SimEIT bridges the gap between physical simulation and AI training, supporting reliable benchmarking and development of advanced reconstruction algorithms.

***Index Terms*—** EIT, Dataset Generation

1. INTRODUCTION

Electrical impedance tomography (EIT) utilizes electrode arrays to inject electrical currents and measure boundary voltages through diverse stimulation patterns across a region of interest (ROI), reconstructing spatial conductivity distributions to characterize internal material arrangements. EIT offers advantages over radiation-based techniques (*e.g.*, X-ray, MRI) through non-invasiveness, low cost-effectiveness, minimal power requirements, and rapid response, enabling medical applications such as breast cancer detection, stroke diagnosis, lung function monitoring, alongside industrial process monitoring.

Solving EIT's nonlinear, severely ill-posed inverse problem traditionally employs Jacobian-based regularization techniques, including Newton-Raphson and Tikhonov regularization. However, these model-dependent approaches suffer from reconstruction artifacts under minor modeling errors, low spatial resolution, and sensitivity to measurement noise. Deep learning methods address these limitations by embedding prior knowledge and enhancing noise robustness through architectures such as auto-encoders, convolutional neural networks, graph neural networks, and diffusion models [1].

The development of AI-driven EIT reconstruction is hindered by a scarcity of large-scale, open, and reproducible training datasets. Many existing datasets are proprietary,*e.g.*, for lung disease imaging, limiting their use for broader research. Even when datasets are public, such as the CDEIT [2] and Edinburgh mfEIT datasets [3], their generation code is unavailable.

Despite progress, EIT research is hampered by a lack of scalable, adaptable, and open-source datasets. This scarcity of diverse training data obstructs the development of robust AI-driven reconstruction algorithms needed to solve EIT's ill-posed inverse problem. To address this critical gap, we introduce **SimEIT**: an open-source, parallelized framework for generating large-scale, physically consistent EIT datasets. Built on the validated EIDORS engine, SimEIT integrates flexibility, reproducibility, and scalability through several key innovations:

- **Modular Framework & Parallel Processing**: Flexible architecture with interchangeable components enables parallel execution in geometry generation and simulation stages, overcoming scalability bottlenecks.
- **Geometry-Boundary Flexibility**: Parametric customization of inclusion shapes (*e.g.*, circles, ellipses, triangles), conductivity distributions, electrode placements, and domain boundaries (*e.g.*, spherical substrates).
- **Reproducible High-Throughput Synthesis**: Deterministic seed control ensures batch-wise traceability for large-scale, physically accurate data generation.
- **AI-Ready Data Optimization**: Multi-resolution ground-truth maps (*e.g.*, 256×256 to 32×32), differential outputs, and metadata-linked HDF5 storage with PyTorch integration.
- **EIDORS-Based Physical Fidelity**: Maintains physical consistency while supporting MATLAB and open-source Octave environments.
- **Open Ecosystem & Visualization**: Public codebase, Hugging Face demos, configurable noise models, and visualization tools enable community-driven expansion and validation.

By democratizing large-scale EIT data synthesis, SimEIT accelerates inverse solver development, enables systematic study of ill-posedness origins, and establishes a foundation

for reproducible AI advancements in the field, launching the realization of the EIT promise across medical, industrial, and scientific domains.

2. METHODOLOGY

2.1. Framework Architecture Overview

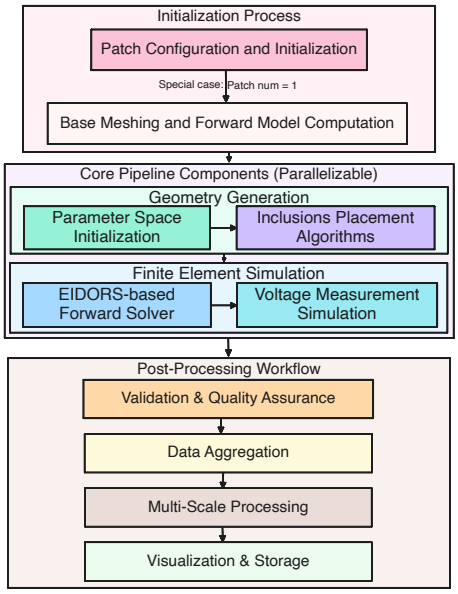

Fig. 1. Overview of the dataset generation architecture.

The proposed framework employs a modular architecture comprising three coordinated phases: **initialization, core framework execution, and post-processing**, as shown in Figure 2.1. The initialization phase establishes deterministic reproducibility through batch-specific seed initialization, base mesh configuration, and measurement pattern (e.g., adjacent/opposite current injection), while resolving baseline boundary conditions via cached the base forward solution. Subsequent framework stages integrate parallelized geometry generation and finite element simulation components, leveraging adaptations of the established EIDORS forward solver. Post-processing implements multi-scale data transformation and validation protocols, ensuring physical consistency while optimizing outputs for machine learning integration.

2.2. Framework Initialization

The framework initialization establishes the computational foundation for large-scale EIT dataset generation. The process begins by integrating custom function libraries into the working directory and initializing the EIDORS framework, ensuring access to validated forward solvers and reconstruction algorithms. User-defined parameters are then specified, including electrode count (default: 16), FEA mesh resolution (default: 256×256), and geometric domain dimensions. A batch processing system is implemented, where each batch operates with a unique identifier to enable parallel execution across distributed computing resources. Deterministic reproducibility is achieved through a random seed initialization function based on the batch number.

The initialization process sets up key parameters for data generation. Users can define the number of samples per batch, which defaults to 5,000, and control the diversity of simulated objects. The initialization also includes setting the maximum number of inclusions per sample (up to four by default), selecting the type of the inclusions from a library of shapes (default: circles, ellipses, rectangles, and triangles), and defining limits for their geometric properties like aspect ratio and size. To optimize performance, the framework first generates a homogeneous reference image and caches its forward solution. This cached result is then used for differential imaging across all subsequent batches, eliminating redundant computations.

2.3. Geometry and Phantom Generation

This component procedurally generates diverse and reproducible phantoms by parameterizing inclusion shapes, conductivities, and positions. Reproducibility is ensured through a batch-specific seeding mechanism, enabling deterministic generation of complex geometries for large-scale simulations.

The generation process involves two main steps: parameter definition and iterative placement.

Parameter Definition. Users can specify:

- **Inclusion Properties:** The number and type of shapes (*e.g.*, circles, polygons), along with their conductivity values sampled from user-defined distributions (*e.g.*, linear or logarithmic).
- **Geometric Constraints:** Rules governing object placement, such as minimum inclusion size, aspect ratio limits, and exclusion zones (*e.g.*, preventing placement near electrodes).

Iterative Placement. An algorithm generates valid phantom configurations by:

1. **Generating Candidates:** Stochastically creating inclusion geometries based on the defined parameters.
2. **Validating Placement:** Checking each candidate against spatial constraints, including collision detection to manage or prevent overlaps.
3. **Finalizing Configuration:** Iterating until a valid configuration that satisfies all rules is found. Invalid candidates are discarded and regenerated.

The framework stores the generated dataset shapes in metadata in CSVs for traceability and algorithm validation.

2.4. Finite Element Implementation

The EIT forward problem is modeled with the Complete Electrode Model (CEM) on a bounded domain $\Omega \subset \mathbb{R}^2$ with

boundary $\partial\Omega$ and electrodes $\{\Gamma_l\}_{l=1}^N$. For conductivity $\sigma(\mathbf{r})$, potential $u(\mathbf{r})$, contact impedances $\{\rho_l\}$, applied currents $\{I_l\}$, and electrode potentials $\{V_l\}$, the governing system is $\nabla \cdot \big(\sigma(\mathbf{r})\nabla u(\mathbf{r})\big) = 0, \mathbf{r} \in \Omega$; $u(\mathbf{r}) + \rho_l\, \sigma(\mathbf{r}) \frac{\partial u(\mathbf{r})}{\partial \mathbf{n}} = V_l, \mathbf{r} \in \Gamma_l$; $\int_{\Gamma_l} \sigma(\mathbf{r}) \frac{\partial u(\mathbf{r})}{\partial \mathbf{n}}\, dS = I_l, l = 1, \ldots, N$; $\sigma(\mathbf{r}) \frac{\partial u(\mathbf{r})}{\partial \mathbf{n}} = 0, \mathbf{r} \in \partial\Omega \setminus \bigcup_{l=1}^N \Gamma_l$; $\sum_{l=1}^N I_l = 0, \sum_{l=1}^N V_l = 0$.

FEM discretization Let $\{\phi_i\}_{i=1}^M$ be first order nodal basis functions on a fixed triangular mesh of Ω. With a piecewise constant conductivity vector $\boldsymbol{\sigma} = [\sigma_1, \ldots, \sigma_M]^\top$, the Galerkin discretization with CEM boundary conditions yields the sparse linear system

$$\mathbf{K}(\boldsymbol{\sigma})\, \mathbf{U} = \mathbf{F}, \tag{1}$$

where $\mathbf{U}$ collects nodal and electrode unknowns, and the stiffness matrix is

$$K_{ij} = \int_\Omega \sigma(\mathbf{r})\, \nabla\phi_i(\mathbf{r}) \cdot \nabla\phi_j(\mathbf{r})\, d\mathbf{r} + \sum_{l=1}^N \int_{\Gamma_l} \frac{1}{\rho_l}\, \phi_i(\mathbf{r})\phi_j(\mathbf{r})\, dS. \tag{2}$$

The right hand side $\mathbf{F}$ encodes the applied current pattern and the CEM constraints. Electrode voltages are obtained by the measurement operator $\mathbf{M}$ as $\mathbf{V} = \mathbf{M}\mathbf{U}$.

Differential measurement model To reduce modeling bias and drift, SimEIT computes differential voltages with respect to a homogeneous reference conductivity σ_0. Let $\mathbf{U}(\boldsymbol{\sigma})$ and $\mathbf{U}(\sigma_0)$ be the solutions of (1) for $\boldsymbol{\sigma}$ and σ_0, respectively. The synthetic measurement vector is

$$\Delta\mathbf{V} = \mathbf{M}\big(\mathbf{U}(\boldsymbol{\sigma}) - \mathbf{U}(\sigma_0)\big), \tag{3}$$

which improves conditioning and robustness for ML.

Cached mesh and high throughput SimEIT employs a fixed, precomputed base mesh and caches reference quantities to avoid remeshing. The matrix $\mathbf{K}(\sigma_0)$ for the homogeneous model and the corresponding solution $\mathbf{U}(\sigma_0)$ are computed once and reused. For each sample, only the elementwise conductivities in $\boldsymbol{\sigma}$ are updated on the fixed topology, followed by sparse reassembly and solution of (1). Forward solves and differential projections (3) are performed using the validated EIDORS implementation [4].

Batch parallelization For a batch $\mathcal{S}_b = \{s_1, \ldots, s_B\}$ with conductivities $\{\boldsymbol{\sigma}_s\}$, SimEIT evaluates, independently for each $s \in \mathcal{S}_b$, $\mathbf{U}_s = \mathbf{K}^{-1}\big(\boldsymbol{\sigma}_s\big)\, \mathbf{F}, \Delta\mathbf{V}_s = \mathbf{M}\big(\mathbf{U}_s - \mathbf{U}_0\big)$, where $\mathbf{U}_0 = \mathbf{U}(\sigma_0)$. The outputs $\{\boldsymbol{\sigma}_s, \Delta\mathbf{V}_s\}$ are written to HDF5 with metadata for deterministic reproducibility and downstream machine learning.

2.5. Post-Processing Workflow

Data Integrity Validation and Quality Assurance: Our framework integrates a validation stage to ensure data integrity. An automated process systematically scans for anomalies, such as incomplete simulations or corrupted data files. Key checks verify the physical plausibility of outputs (*e.g.*, non-zero voltages) and the consistency between conductivity maps and voltage measurements. Corrupted data samples are automatically identified and flagged for regeneration.

Multi-Resolution Processing and Data Aggregation: To prepare the data for AI model training, the framework processes the high-resolution ground-truth conductivity maps (*e.g.*, 256×256) into multiple lower resolutions (*e.g.*, 128×128, 64×64, and 32×32). This multi-resolution output is generated using configurable interpolation methods (*e.g.*, bilinear, bicubic) and supports processing in both linear and logarithmic domains to handle wide conductivity ranges. The resulting data is aggregated into memory-efficient chunks and stored in HDF5 files, ensuring compatibility with various deep learning architectures and computational constraints.

Data Visualization and Storage: The framework provides integrated tools for data validation and management. Visualization modules generate statistical summaries of dataset properties (*e.g.*, object counts, conductivity distributions) and allow for inspection of individual samples, including FEM models and ground-truth maps. This enables systematic quality control and bias detection. For efficient handling, datasets are stored in the HDF5 format, which uses chunked compression to minimize storage size and supports hierarchical organization for scalable batch processing. All simulation parameters are stored as metadata, ensuring fulls reproducibility.

3. GENERATED DATASETS

This section details the validation of our framework through generating two large-scale synthetic EIT datasets using an adjacent current injection pattern. It describes the efficient fixed-meshing strategy enabling high-throughput simulation, presents the characteristics and statistical distributions of the dataset, showcases sample diversity, and outlines framework features like multi-resolution support and PyTorch integration, addressing the critical need for scalable, open-source EIT data. The datasets were generated using MATLAB 2020b on a high-performance computing (HPC) cluster with distributed processing via SLURM. Both datasets are publicly available to support reproducible EIT research.

Dataset 1: Mixed Shapes with Diverse Geometries: The first dataset contains 100,000 samples featuring four distinct inclusion shapes: triangles, rectangles, circles, and ellipses. Each shape category exhibits configurable geometric degrees of freedom (DOF), such as position, size, and aspect ratio. For example, circles are defined by center coordinates (x, y) and radius (3 DOF), while ellipses require additional axes and rotation parameters (5 DOF). The maximum per-shape DOF was capped at 7 to maximize shape diversity while maintaining computational tractability. The statistical properties of the generated dataset, summarized in Figure 2, confirm its controlled diversity. The number of inclusions per sample

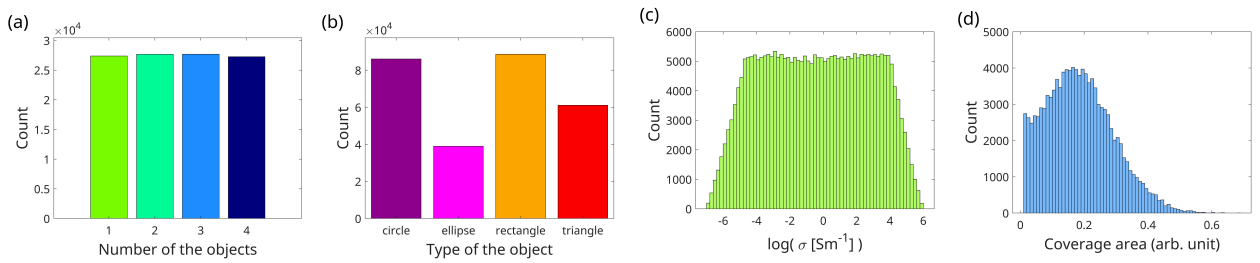

Fig. 2. Statistical distributions of key parameters in the first generated EIT dataset. (a) Number of objects. (b) Type of object geometries. (c) Logarithmic conductivity values of inclusions. (d) Fractional coverage area of objects within the domain.

is uniformly distributed from one to four (a), ensuring balanced complexity. The distribution of shapes is varied, with rectangles and circles being most frequent (b). Inclusion conductivities follow a near-uniform logarithmic distribution (c), offering a wider range than typical phantoms. The fractional area covered by objects is right-skewed, prioritizing samples with lower object density while still including high-density cases (d).

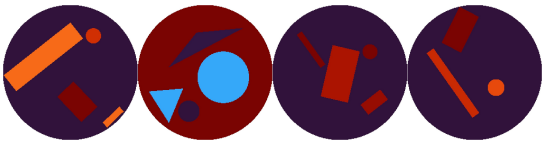

Fig. 3. Example inclusion geometries and conductivity distributions. Each domain shows unique configurations of shapes with varying sizes, orientations, arrangements, and conductivity values (color-coded).

Samples of Dataset 1: Our framework procedurally generates EIT datasets with morphological diversity. Each sample represents a circular domain containing randomized inclusions of circles, triangles, rectangles, and ovals with heterogeneous sizes, orientations, and spatial distributions (Figure 3). Conductivity values σ follow a logarithmic distribution spanning multiple orders of magnitude, creating challenging physical scenarios. To accommodate different computational requirements, conductivity maps are generated at 256×256 pixel resolution and downsampled to 128×128, 64×64, and 32×32 resolutions. A PyTorch DataLoader integrates these datasets into deep learning workflows, while built-in circular masks exclude extraneous regions outside the EIT domain boundary, ensuring only relevant pixels are processed.

Dataset 2: Circular Inclusions with Variations: This dataset comprises 100,000 samples containing exclusively circular inclusions. Each circle is defined by its center coordinates (x, y) and radius, resulting in 3 DOF per inclusion. The number of circles per sample varies from 1-4, with their radii and conductivities sampled from user-defined distributions. This dataset focuses on parametric variations of circular shapes to facilitate targeted studies on the impact of inclusion size and conductivity on EIT reconstruction performance.

Code and Data Availability: The SimEIT framework is publicly available on the project page, including code, documentation, and examples. Two large-scale synthetic EIT datasets generated using SimEIT are hosted on Hugging Face, complete with metadata files for traceability. A live demo is also accessible, allowing users to interactively explore the generated datasets. SimEIT's capabilities can be extended by utilizing EIDORS implemented methods, such as different 3D environments, such as lung monitoring, etc.

4. CONCLUSION

We introduced **SimEIT**, an open-source, parallelized framework designed to address the critical need for large-scale, customizable datasets in EIT. Built on the validated EIDORS engine, SimEIT enables the generation of diverse synthetic data with user-defined geometries, conductivity distributions, and noise models. Its modular framework ensures reproducibility through deterministic seeding, computational efficiency via parallel processing and cached base meshes, and AI-readiness through multi-resolution outputs and HDF5 storage. By providing open access to scalable and traceable data generation, SimEIT accelerates the development and robust benchmarking of AI-driven EIT reconstruction algorithms.

5. REFERENCES

[1] Ayman A. Ameen, Achim Sack, and Thorsten Pöschel, "TSS-ConvNet for electrical impedance tomography image reconstruction," vol. 45, no. 4, pp. 045006.

[2] Shuaikai Shi, Ruiyuan Kang, and Panos Liatsis, "A Conditional Diffusion Model for Electrical Impedance Tomography Image Reconstruction," vol. 74, pp. 1–16.

[3] Zhou Chen and Yunjie Yang, "Structure-Aware Dual-Branch Network for Electrical Impedance Tomography in Cell Culture Imaging," vol. 70, pp. 1–9.

[4] Andy Adler and William R B Lionheart, "Uses and abuses of EIDORS: An extensible software base for EIT," vol. 27, no. 5, pp. S25–S42.

Supplementary Materials

A. RELATED WORK ON EIT DATASETS

Training AI models necessitates substantial datasets, which are derived from either simulated or experimental sources. In electrical impedance tomography (EIT), many training datasets are closed-source, such as those generated via EIDORS simulations featuring geometric configurations like circles and triangles [1], or specialized datasets for lung diseases containing 10,000 samples simulated using PyEIT without public access to data or code [2]. Similarly, closed-source datasets exist for applications like brain haemorrhage detection.

Open-source simulated datasets are scarce. The CDEIT model dataset [3] offers 57,600 samples but restricts object placement/conductivity parametrization and lacks generation code. The Edinburgh mfEIT Dataset [4], widely used for multi-frequency imaging, contains objects with continuous conductivity variations solved via COMSOL and MATLAB, yet its closed-generation code impedes sample expansion. Likewise, Edinburgh's cell-culture phantom dataset [5] and datasets with varying conductivity levels remain without accessible generation source code. Experimental open-source datasets are more prevalent but smaller in scale. Examples include geometric phantoms in water tanks (more than 125,000 samples), animal models (*e.g.*, pigs [6]), and human studies: respiratory data under CPAP ventilation [7], cardiorespiratory monitoring [8], and stroke/brain observations [9]. However, experimental data acquisition faces significant challenges, including the complexity of acquiring large-scale datasets and reduced controllability compared to simulations.

B. METHODOLOGY DETAILS

B.1. Framework Innovations

Our architecture advances EIT methodology by resolving critical limitations in conventional dataset synthesis. Modular parameterization supports systematic exploration of anatomical geometries and conductivity distributions, enhancing dataset diversity beyond empirical phantom constraints. Deterministic seed control coupled with configurable noise models enables precise quantification of inverse solver sensitivity to measurement artifacts, a critical requirement for robust algorithm. Furthermore, scale-adaptive outputs bridge the resolution gap between simulation meshes and neural network input formats, directly addressing the domain shift problem in learned reconstruction approaches.

Computational efficiency is achieved through parallelization, task-level distribution across framework stages combined with data-parallel execution within finite element solvers. Validation against experimental phantoms facilitates systematic investigation of the EIT inverse problem's ill-posedness across parameter spaces previously inaccessible to data-driven approaches. HDF5-based storage with lossless compression further ensures efficient handling of datasets.

B.2. Forward Problem Formulation and Finite Element Implementation

The Electrical Impedance Tomography (EIT) forward problem is modeled using the complete electrode model (CEM), which governs the relationship between conductivity distributions and boundary voltage measurements. Let Ω denote the imaging domain with boundary $\partial\Omega$, N electrodes $\{\Gamma_l\}_{l=1}^N$, and contact impedances $\{\rho_l\}_{l=1}^N$. For an applied current pattern $\{I_l\}_{l=1}^N$, the potential distribution $u(r)$ within Ω and electrode voltages $\{V_l\}_{l=1}^N$ satisfy the following system:

Governing Equations.

1. *Conservation of charge*:

$$\nabla \cdot (\sigma(r)\nabla u(r)) = 0, \quad r \in \Omega \tag{4}$$

2. *Electrode boundary conditions*:

$$\int_{\Gamma_l} \sigma(r)\frac{\partial u(r)}{\partial \mathbf{n}} dS = I_l, \quad \sigma(r)\frac{\partial u(r)}{\partial \mathbf{n}} = 0 \text{ on } \partial\Omega \setminus \cup_l \Gamma_l \tag{5}$$

$$u(r) + \rho_l \sigma(r)\frac{\partial u(r)}{\partial \mathbf{n}} = V_l \text{ on } \Gamma_l \tag{6}$$

3. *Conservation laws*:

$$\sum_{l=1}^{N} I_l = 0, \quad \sum_{l=1}^{N} V_l = 0 \tag{7}$$

Here, $\sigma(r)$ represents the conductivity distribution, and $\mathbf{n}$ is the outward unit normal vector.

B.3. Finite Element Discretization.

The domain Ω is discretized into finite elements (*e.g.*, triangles in 2D). The discretization configuration is fixed across samples to enable efficient parameter updates without remeshing, featuring:

- A reinforced finite element mesh with enhanced electrode contact zone refinement.
- Uniform baseline conductivity applied to all mesh elements.
- Precomputed forward solution for adjacent current injection patterns.
- Electrode contact impedance values are set.
- Solve the forward problem for the homogeneous image.

Domain discretization is illustrated in Figure 4. The forward problem is solved using the EIDORS toolbox.

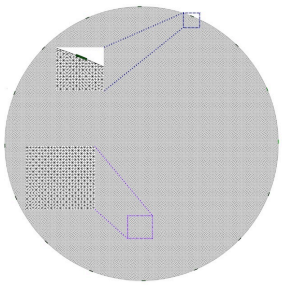

Fig. 4. Domain discretization from the base cached image, enabling rapid parameter updates without remeshing. The meshing is fixed for all the generated samples, while only the selected elements with different conductivities are changed, which saves significant computational power and time.

B.4. Meshing and Simulation

Our framework employs a fixed-domain meshing strategy to efficiently generate synthetic Electrical Impedance Tomography (EIT) datasets. The region of interest (ROI) for the two generated datasets is a circular domain representing an EIT phantom. The ROI is discretized using a uniform square grid with static resolution (*e.g.*, 256×256, illustrated). This approach eliminates computationally expensive remeshing between simulations. Instead, conductivity distributions are varied by modifying element-wise properties within the fixed mesh, enabling high-throughput dataset generation with minimal overhead.

Electrode-adjacent elements feature refined mesh reinforcement to ensure accurate modeling of current injection and voltage measurements under the Complete Electrode Model (CEM). This enhancement preserves simulation fidelity at electrode-ROI interfaces during finite element analysis (FEA).

Forward simulations leverage the established EIDORS package for CEM-based FEA.

The fixed-mesh paradigm ensures consistent topology across all samples, streamlining training for both structured (pixel-based) and unstructured (element-based) data representations. This consistency addresses the scarcity of large-scale open-source EIT datasets while supporting scalable synthesis for machine learning applications.

B.5. Efficient Simulation via Fixed-Mesh Conductivity Mapping

SimEIT employs a fixed finite element mesh across all simulations to avoid the computational overhead of repeated mesh generation. The mesh, defined by nodes and elements in the forward model (`img.fwd_model`), remains constant. Instead, conductivity distributions are updated by mapping geometric objects onto the mesh. This is achieved through a **pre-indexing strategy** that efficiently links high-resolution image pixels to mesh elements.

Approach

1. **High-Resolution Grid**: A 2D grid (resolution $N \times N$) defines pixel coordinates `(X, Y)` spanning the domain $[-1, 1]^2$.
2. **Mesh Pre-indexing**: For each pixel, identify the four corner nodes of its bounding square (side length = grid spacing). These nodes are precomputed and rounded to floating-point precision for lookup.
3. **Element Mapping**: Mesh elements containing ≥ 3 of a pixel's corner nodes are flagged as "covered" by that pixel. This mapping is reused for all simulations.
4. **Conductivity Assignment**: When an object overlaps a pixel, the conductivity of all mapped elements is updated to the object's value.

B.5.0.1. Pseudo-Code

```
# precomputed a fixed mesh (once)
nodes = round(img.fwd_model.nodes, precision)
elements = img.fwd_model.elems
v_homogeneous = fwd_solve(elements)


for sample in dataset (Parallelizable):
    # Initialize conductivity (background)
    elem_cond = background_value * ones(n_elements)

    for obj in objects of the sample:
        # Get object's shape & conductivity
        shape_fn = define_shape(obj)
        obj_cond = obj.conductivity

        # Identify pixels inside object (high-res mask)
        in_obj = shape_fn(X, Y)

        # Update elements: use pre-indexed node mapping
        for pixel in pixels_where(in_obj):
            # Retrieve precomputed corner nodes
            corners = get_corner_nodes(pixel)

            # Find elements containing >=3 corners
            covered_elems = find_elements_with_nodes(
                elements, corners, min_nodes=3)

            # Update conductivity for covered elements
            elem_cond[covered_elems] = obj_cond - \
                                       background_value

    # Solve EIT forward problem
    v_target = fwd_solve(elem_cond)
    v_diff = v_target - v_homogeneous
```

This fixed mesh reuse with pre-indexed element mapping enables rapid conductivity updates without remeshing, significantly accelerating dataset generation while maintaining simulation fidelity. The approach is particularly effective for high-throughput synthesis of diverse EIT phantoms.

B.6. Metadata Integration.

Generated geometries are paired with structured metadata, including inclusion positions, radii, conductivity values, and spatial coverage ratios. These parameters are stored in CSV files, establishing traceability between forward simulations and inverse problem benchmarks.

By combining constraint-based randomization with deterministic seeding, the framework balances diversity and reproducibility, essential for validating reconstruction algorithms. Parallel execution across batches ensures computational efficiency, while modular parameterization supports both standardized and user-customized datasets.

B.7. Data Integrity Validation and Quality Assurance

To ensure dataset reliability and reproducibility, the framework implements a multi-stage validation protocol. An automated patch-wise verification mechanism systematically scans precomputed entries for anomalies, including incomplete simulations, corrupted conductivity maps, or malformed finite element meshes. Integrity checks are optimized through parallel batch processing of the data chunks, minimizing computational overhead while validating large-scale datasets.

Key validation steps include: (1) **Null Value Detection**, which flags voltage measurements or conductivity values equal to zero, indicating incomplete simulations; (2) **Dimensional Consistency Checks** to confirm alignment between mesh geometries and corresponding electrical measurements; and (3) **Metadata Integrity Verification**, ensuring seed values and simulation parameters are correctly logged. Suspect entries are isolated using a hash-based indexing system that identifies corrupted batches for selective re-execution, avoiding full framework reruns.

This approach automated anomaly detection coupled with physics-based consistency checks, guarantees dataset coherence, supporting reproducible training and evaluation of AI-driven EIT models.

B.8. Multi-Resolution Processing and Data Aggregation

The framework incorporates a multi-resolution processing stage to harmonize simulated data with varying spatial scales, ensuring compatibility with AI models of differing architectures and computational constraints. High-resolution conductivity maps (*e.g.*, 256×256 pixels) generated from finite element simulations are systematically downsampled to lower resolutions (eg. 32×32, 64×64, 128×128) through adaptive interpolation methods. This step addresses the inherent disparity between the high dimensionality of reconstructed images and the comparatively sparse voltage measurements, enabling efficient training of real-time capable models while preserving simulation fidelity. Key operations include:

- **Resolution scaling**: Downsampling via bilinear, bicubic, or nearest-neighbor interpolation in linear or logarithmic domains, accommodating conductivity variations spanning multiple orders of magnitude.
- **Domain transformation**: Optional log-to-linear conversions to mitigate numerical instability during AI model training.
- **Batch normalization**: Per-resolution statistical normalization to ensure dataset consistency.

Data aggregation employs a chunk-based workflow to partition large datasets into memory-efficient segments, minimizing overhead during parallel processing. Downsampled conductivity maps and voltage measurements are stored hierarchically in HDF5 format, preserving metadata linkages between geometric configurations, simulation parameters, and outputs.

B.9. Dataset Visualization and Storage Management

The framework integrates robust visualization tools and efficient storage protocols to ensure dataset quality and accessibility. Visualization modules provide: (1) statistical summaries of dataset properties, including object counts, conductivity distributions, and spatial coverage ratios; (2) Automated aggregation of metadata to generate histograms and bar plots for bias detection and parameter distribution analysis; and (3) Stratified sampling to visualize finite element models, ground-truth conductivity maps, and voltage difference patterns. These tools enable systematic validation of geometric configurations, electrode placements, and boundary condition alignments across simulations.

For storage, the framework employs HDF5 files with chunked compression and optional zip archiving to minimize storage footprints. Dual validation stages pre-downscaling and post-downscaling ensure data integrity, while normalized physical quantities and metadata-derived masks preserve simulation consistency. Comprehensive metadata packaging traces all simulation parameters, and HDF5's hierarchical structure supports scalable batch processing. This architecture facilitates rapid dataset iteration and reproducible large-scale EIT research with minimal manual intervention.

B.10. Additional Figures

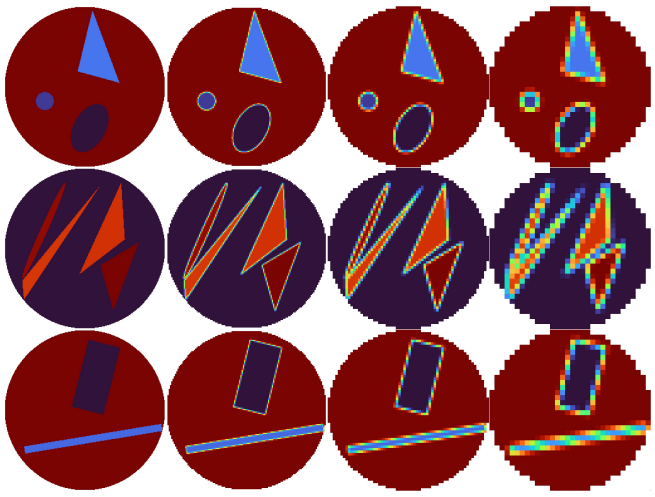

Fig. 5. Multi-resolution outputs (256×256 to 32×32 pixels) for flexible integration with neural network architectures.

C. GENERATED DATASETS DETAILS

C.1. Meshing and Simulation

Our framework employs a fixed-domain meshing strategy to efficiently generate synthetic Electrical Impedance Tomography (EIT) datasets. The region of interest (ROI) for the two generated datasets is a circular domain representing an EIT phantom. The ROI is discretized using a uniform square grid with static resolution (*e.g.*, 256×256, illustrated). This approach eliminates computationally expensive remeshing between simulations. Instead, conductivity distributions are varied by modifying element-wise properties within the fixed mesh, enabling high-throughput dataset generation with minimal overhead.

Electrode-adjacent elements feature refined mesh reinforcement to ensure accurate modeling of current injection and voltage measurements under the Complete Electrode Model (CEM). This enhancement preserves simulation fidelity at electrode–ROI interfaces during finite element analysis (FEA).

Forward simulations leverage the established EIDORS package [10] for CEM-based FEA.

The fixed-mesh paradigm ensures consistent topology across all samples, streamlining training for both structured (pixel-based) and unstructured (element-based) data representations. This consistency addresses the scarcity of large-scale open-source EIT datasets while supporting scalable synthesis for machine learning applications.

To validate our computational framework, we generated two large-scale electrical impedance tomography (EIT) datasets, each comprising $> 100{,}000$ unique samples. The first dataset includes four distinct inclusion shapes: triangles, rectangles, circles, and ellipses. Each shape category exhibits configurable geometric degrees of freedom, such as position, size, and aspect ratio. For example, circles are defined by center coordinates (x, y) and radius (3 DOF), while ellipses require additional axes and rotation parameters (5 DOF). The maximum per-shape DOF was capped at 7 to maximize shape diversity while maintaining computational tractability.

C.2. Dataset 1 Statistical Analysis

Our framework automates dataset generation and statistical visualization, as summarized in Figure 2. Panel (a) shows the distribution of inclusion counts per sample, which follows a uniform distribution across 1–4 objects. This design ensures balanced algorithm evaluation across structural complexities. Panel (b) illustrates the frequency of each shape, revealing rectangles and circles as predominant, followed by triangles and ellipses. While users may enforce uniform shape distributions, no such constraint was applied here. Panel (c) characterizes the logarithmic conductivity ($\log \sigma$) of inclusions. Values span approximately -6 to 5 with near-uniform distribution and minor edge tapering, exceeding typical empirical phantoms and enabling broad conductivity-space exploration. Additionally, the conductivity distribution of the inclusions can be changed by the user. Panel (d) quantifies areal coverage, defined as the fractional domain area occupied by objects. This distribution is right-skewed, indicating prevalent partial coverage scenarios while including rarer high-coverage cases.

Collectively, these statistics confirm that our datasets feature controlled diversity in object count, geometry, conductivity, and spatial coverage. This structured variability establishes a rigorous foundation for developing and benchmarking robust EIT reconstruction algorithms.

The second dataset comprises exclusively circular inclusions with parametrically varied radii, conductivities, and counts per sample. Both datasets are publicly available to support reproducible EIT research.

D. OVERVIEW OF SIMEIT FRAMEWORK

This paper introduces **SimEIT**, an open-source, parallelized framework designed to address the critical scarcity of large-scale, customizable datasets for Electrical Impedance Tomography (EIT) research. SimEIT bridges the existing gap in the EIT datasets frameworks by providing a flexible, efficient framework built on the validated EIDORS simulation engine (compatible with MATLAB/Octave), enabling fully customizable phantom generation with diverse shapes (circles, ellipses, rectangles, triangles), user-defined conductivity distributions, spatial constraints, and electrode configurations. Its modular architecture features three key phases:

1. **Initialization** ensuring deterministic reproducibility via seed control and baseline caching;
2. **Core Execution** generating geometries with randomized inclusions and performing parallelized finite element method (FEM) simulations using the Complete Electrode Model (CEM); and
3. **Post-processing** with multi-resolution downscaling, rigorous physics-based validation, and anomaly detection.

Innovations include batch-specific seeds for identical reproducibility, reuse of base meshes for efficiency, domain-shift mitigation via adaptive conductivity map scaling (32×32 to 256×256), and HDF5 storage with lossless compression and metadata traceability. By enabling large-scale, customizable dataset synthesis with open access, SimEIT facilitates systematic studies of boundary effects, geometry impacts, and measurement artifacts, advancing AI model training for EIT reconstruction. A real-time HuggingFace demo enhances accessibility.

E. REFERENCES

[1] Xuanxuan Yang, Yangming Zhang, Haofeng Chen, Gang Ma, and Xiaojie Wang, "A Two-Stage Imaging Framework Combining CNN and Physics-Informed

Neural Networks for Full- Inverse Tomography: A Case Study in Electrical Impedance Tomography (EIT)," vol. 32, pp. 1096–1100.

[2] Areen K. Al-Bashir, Duha H. Al-Bataiha, Mariem Hafsa, Mohammad A. Al-Abed, and Olfa Kanoun, "Electrical impedance tomography image reconstruction for lung monitoring based on ensemble learning algorithms," vol. 11, no. 5, pp. 271–282.

[3] Shuaikai Shi, Ruiyuan Kang, and Panos Liatsis, "A Conditional Diffusion Model for Electrical Impedance Tomography Image Reconstruction," vol. 74, pp. 1–16.

[4] Zhou Chen, Jinxi Xiang, Pierre-Olivier Bagnaninchi, and Yunjie Yang, "MMV-Net: A Multiple Measurement Vector Network for Multifrequency Electrical Impedance Tomography," vol. 34, no. 11, pp. 8938–8949.

[5] Zhou Chen and Yunjie Yang, "Structure-Aware Dual-Branch Network for Electrical Impedance Tomography in Cell Culture Imaging," vol. 70, pp. 1–9.

[6] Dani Bodor, Peter Somhorst, Annemijn Jonkman, Walter Baccinelli, Jantine Wisse-Smit, and Juliette Francovich, "Eitprocessing," .

[7] Ella Frances Sophia Guy, Jaimey Anne Clifton, Trudy Caljé-van der Klei, Rongqing Chen, Jennifer Knopp, Knut Moeller, and James Geoffrey Chase, "Respiratory dataset from PEEP study with expiratory occlusion," .

[8] Ella Frances Sophia Guy, Isaac Flett, Jaimey Anne Clifton, Trudy Caljé-van der Klei, Rongqing Chen, Jennifer Knopp, Knut Moeller, and James Geoffrey Chase, "Respiratory and heart rate monitoring dataset from aeration study," .

[9] Nir Goren, James Avery, Thomas Dowrick, Eleanor Mackle, Anna Witkowska-Wrobel, David Werring, and David Holder, "Multi-frequency electrical impedance tomography and neuroimaging data in stroke patients," vol. 5, no. 1, pp. 180112.

[10] Andy Adler and William R B Lionheart, "Uses and abuses of EIDORS: An extensible software base for EIT," vol. 27, no. 5, pp. S25–S42.