A THESIS

FOR THE DEGREE OF MASTER OF ENGINEERING

Generative Adversarial Networks Methods for Electrical Impedance Tomography Reconstruction and Post-processing

Solano Sánchez Felipe Alberto

Department of Electronic Engineering

GRADUATE SCHOOL

JEJU NATIONAL UNIVERSITY

REPUBLIC OF KOREA

2023. 12

Generative Adversarial Networks Methods for Electrical Impedance Tomography Reconstruction and Post-processing

Solano Sánchez Felipe Alberto

(Supervised by Professor Kim Kyung Youn, PhD)

A thesis submitted in partial fulfilment of the requirement for the

Degree of Master of Engineering

2023. 12

The thesis has been examined and approved.

Kyung Youn Kim

Kim, Kyung Youn, PhD

Professor, Department of Electronic Engineering

Seokjun Ko

Ko, Seok Jun, PhD

Professor, Department of Electronic Engineering

Woo Young Kim

Kim, Woo Young, PhD

Associate Professor, Department of Electronic Engineering

December 2023

Date

Department of Electronic Engineering

GRADUATE SCHOOL

JEJU NATIONAL UNIVERSITY

REPUBLIC OF KOREA

Table of Contents

Abstract

Electrical Impedance Tomography (EIT) is a versatile method applied for imaging in fields such as phase flow analysis, medical diagnostics, and sensing materials imaging. It provides real-time, non-invasive cross-sectional imaging, aiding in pipeline optimization, energy reduction, and environmental impact mitigation. For medical purposes, EIT has been integrated with conductive fabrics for artificial skin, enabling simultaneous monitoring of electrical conductivity and mechanical properties. This thesis proposes Generative Adversarial Network (GAN) models for EIT reconstructions related to the mentioned fields. These neural networks based models effectively handle complex conductivity distributions, learn intricate features directly from data, and support learning by simulation cases. Their adaptability to various scenarios, data-driven nature, and ability to incorporate regularization techniques make them promising tools for EIT applications.

The results demonstrate that the designed GAN-based models successfully reconstruct target positions and a more uniform background. The performance is also contrasted to alternative neural network approaches, including neural network and deep neural network models. The evaluation metrics indicate superior performance of the GAN models in comparison.

Keywords: Electrical Impedance Tomography (EIT); Two-phase Flow; Inverse Problem; Reconstruction Algorithms; Generative Adversarial Networks (GAN); Piezoresistive Fabric; Attention Mechanism.

Abbreviations

EIT	Electrical Impedance Tomography
GAN	Generative Adversarial Network
AWGN	Additive White Gaussian Noise
pix2pix	Pixel to pixel
mNR	Modified Newton Raphson
ADAM	Adaptive Moment Estimation optimizer
SGD	Stochastic Gradient Descent
RNN	Recurrent Neural Network
EIDORS	Electrical Impedance and Diffuse Optical Tomography Reconstruction Software
CC	Correlation Coefficient
SSIM	Structural Similarity Index
RIE	Relative Image Error
RCR	Relative Size Coverage Ratio
LAPGAN	Laplacian Generative Adversarial Network
NN	Neural Network
DNN	Deep Neural Network

List of Figures

List of Tables

1. Introduction

Electrical impedance tomography (EIT) is an imaging method that provides insight into an object's interior by the injection of currents at its boundary and the measurement of voltages at the same points. Some of its application fields are phase flow, medicine and materials imaging. In the case of phase flow procedures, two-phase flow is one of its basic types, which includes various combinations between liquid, solid or gas flows. It is present in both nature or industry settings, encompassing scenarios such as the movement of particles in oils, as well as the flow of liquids in pipes. Gaining insight into the behavior of two-phase flow is essential for optimizing pipelines, reducing energy usage, and mitigating environmental effects [1]. To address this challenge, EIT offers a straightforward and non-invasive cross-sectional method that has the capability to capture real-time images of the dynamics of two-phase flow [2]. Regarding medical applications, the combination of EIT with conductive fabrics for artificial skin applications has been explored in recent studies [3-4]. This integration allows for the simultaneous monitoring of both the electrical conductivity and mechanical properties of the fabrics. At the same time, a more comprehensive assessment of the monitored area's health status is possible and the chance for an early intervention with the enhancement of diagnostic capabilities, contributing to improved healthcare outcomes.

However, only for simple geometry cases the reconstruction problem can be obtained by an analytical solution. For complex cases, numerical methods are required since the inverse problem to be solved is ill-posed. This means that the solution is highly unstable to data

perturbations [5]. To deal with this, regularization methods such as modified Newton-Raphson or Landweber are applied since they improve the precision of the reconstruction.

Another field around EIT reconstruction has been the study of neural network architectures. With the capability of managing non-linear relationships in data, they show good performance when dealing with complex conductivity distributions. These models have the ability to learn intricate features directly from the data, which can be essential for capturing patterns of EIT problems. Additionally, neural networks allow end-to-end learning, enabling the entire reconstruction process to be optimized as a whole. Their flexibility in terms of architecture, loss functions, and hyperparameters makes them adaptable to various EIT scenarios. Furthermore, neural networks are data-driven, making them well-suited for leveraging large datasets often available in EIT applications by simulations. Finally, they can incorporate regularization techniques within their architecture, upgrades of the model complexity, and techniques for overfitting prevention.

1.1 Scope of the Thesis

This thesis proposes two approaches that employ Generative Adversarial Networks (GAN) for EIT reconstruction and enhancement. For the reconstruction model, the GAN model takes voltages obtained at the border of the object as input data. The model's outcome is generated through a multi-step process. Initially, a wide dataset is compiled, consisting of voltages and images with the corresponding target. During GAN training, the voltage signal serves as the input, while the expected output is images with the target. Through training, the model gradually refines itself to closely resemble the images that contain the target location. To assess the results of the model, simulations that take into account additive white Gaussian noise (AWGN) and measurements taken from experiments are utilized. Also, evaluation of its performance is done by comparing results against those produced by conventional basic models that are based on neural networks and deep neural networks.

Regarding the image enhancement model, its objective is to enhance the quality of results obtained through the modified Newton-Raphson (mNR) method by pix2pix GAN with an attention mechanism. Based on the generative adversarial networks (GAN) approach that consists of learning the mapping between domains, the reconstructed mNR image and the corresponding image depicting the target's location are used. By integrating attention mechanisms into the mentioned model’s architecture, the proposed method effectively captures spatial details of the images, leading to improved outcomes in the reconstruction process. The effectiveness of this method is assessed by applying it to out-of-dataset and noisy mNR images, demonstrating its capability to reconstruct enhanced backgrounds and target representations based on the initial mNR input images.

2. Theoretical Framework

2.1 Electrical Impedance Tomography

Electrical Impedance Tomography (EIT) is an imaging technique in which electrodes are located around a region of interest (Ω) and at the same points voltages (u) are measured while alternating currents are applied in a defined pattern, to relate them and obtain the object's resistivity (ρ). The main used patterns are opposite, adjacent, and trigonometric. This imaging technique is advantageous for several reasons such as the employment of nonionizing radiation, easy portability, and the fact that is relatively inexpensive compared to other tomography methods, allowing it to produce a basic system at a reasonably low cost [6].

Fig. 1 presents a diagram of an EIT basic configuration system. As shown, the main parts of an EIT system consist of the measurement, data acquisition, and data processing sections. The measurement stage comprises both the current injection and voltage measuring electrodes. In this first stage, the data dimension is defined by the current pattern. In the case of the data acquisition stage, the system usually relies on multiplexers and demultiplexers for simpler hardware implementation. Finally, the data processing stage applies a reconstruction algorithm to obtain a conductivity or resistivity estimation to display. This algorithm is a crucial part of the reconstruction since the problem to solve suffers from non-linearity and ill-posedness.

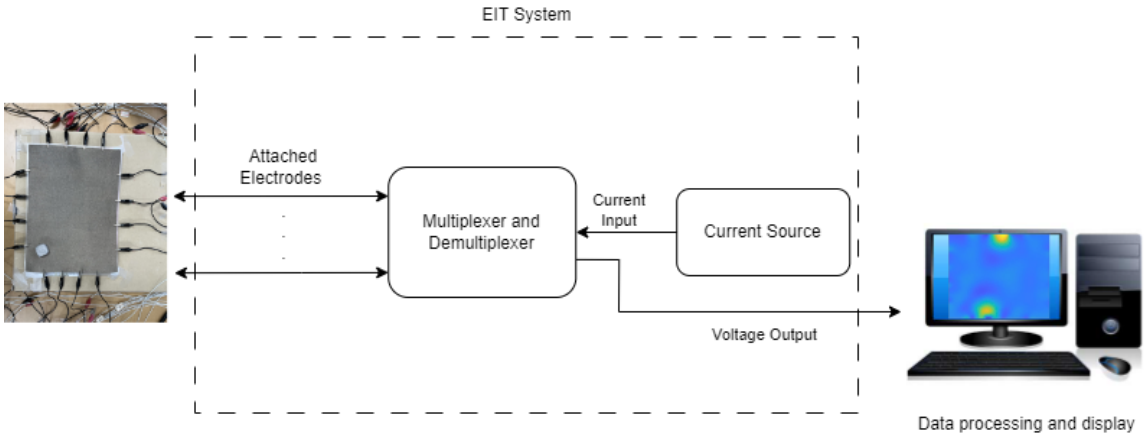


Fig. 1. Basic EIT system

Regarding the mentioned electrodes, diverse physical models have been formulated, varying on the boundary conditions. From all of them, here are presented the shunt model and the complete electrode model, since they are the ones that take most into account the physical effects that are present in experimental setups. For the shunt model, it assumes the used metal for the electrode is a perfect conductor, so the voltage is constant in all the material. This leads to the boundary conditions and constraints [7]:

$$\int_{e_l} \frac{1}{\rho}\frac{\partial u}{\partial v} dS = I_l \qquad l = 1,2,\ldots,L \qquad (1)$$

$$\frac{1}{\rho}\frac{\partial u}{\partial v} = 0 \qquad \text{off} \bigcup_{l=1}^{L} e_l \qquad (2)$$

$$u = U_l \quad on \quad e_l \qquad l = 1,2,\ldots,L, \qquad (3)$$

here e_l is the electrode's surface, v the unit normal in the outward direction, I_l as the currents that are injected, L as the electrode's quantity, and U_l as the boundary measured voltages. Experimentally, this model has shown an underestimation of the resistivity values since it does not take into account the contact impedances at the electrodes' interface.

For the complete electrode model, the term z_l is added as the contact impedance that is

present at the boundary of the electrode and the measured material or tissue. Equations and conditions for this model are[8-10]:

$$u + z_l \frac{1}{\rho}\frac{\partial u}{\partial v} = U_l, \quad x \in e_l, l = 1,2,\dots,L. \qquad (4)$$

$$\int_{e_l} \frac{1}{\rho}\frac{\partial u}{\partial v} dS = I_l, \quad x \in e_l, l = 1,2,\dots,L \qquad (5)$$

$$\frac{1}{\rho}\frac{\partial u}{\partial v} = 0, \quad x \in \partial\Omega \backslash \bigcup_{l=1}^{L} e_l \qquad (6)$$

$$\sum_{l=1}^{L} I_l = 0 \qquad (7)$$

$$\sum_{l=1}^{L} U_l = 0. \qquad (8)$$

To address the complexities to solve (4-8) in intricate scenarios, the finite element method (FEM) is employed. This method is based on a discretization over the object (Ω) to estimate the voltages at the electrodes (U) and the potential distribution (u). This calculation is part of the "forward problem." The explained approach is made by taking these approximations [11]

$$u \approx u^h(x,y) = \sum_{i=1}^{N} \alpha_i \phi_i \qquad (9)$$

$$U \approx U^h = \sum_{j=1}^{L-1} \beta_j n_j. \qquad (10)$$

Here N is the node quantity within the mesh, while ϕ_i represents a two-dimensional basis function of first order. The patterns for measurement, labeled as n represent $n_1 = (1,-1,0,\dots,0)^T$, $n_2 = (1,0,-1,\dots,0)^T, etc. \in \mathbb{R}^L$. The values to determine consist of the nodes and the boundary voltages, respectively α_i and β_j. By applying (9) and (10), the EIT

forward solution can be expressed as the equation of matrices

$$Ab = f \qquad (11)$$

in which A stands for the matrix of coefficients, b represents the solution vector, and f the data vector.

2.2 Modified Newton-Raphson Method

The modified Newton-Raphson (mNR) method is employed to solve iteratively nonlinear equations such as equation (11). In the context of Electrical Impedance Tomography (EIT), this method is used to improve initial estimates of the electrical impedance distribution so that they match the measured data [12]. In each iteration of mNR, the Jacobian matrix is calculated, which represents how voltage data responds to changes in impedance. This matrix is used to update impedance approximations using the Newton-Raphson method, which involves the solution of a linear system of equations. To maintain stability and avoid a not converging point, a regularization term is introduced [13]. The equation for the resistivity approximation is [14]

$$\rho_{i+1} = \rho_i + \left(J_i^T J_i + \alpha W\right)^{-1} \left(J_i^T \left(U - V(\rho_i)\right)\right) \quad (12)$$

here:

- ρ_i: ith iteration resistivity
- J_i: ith iteration Jacobian
- U: obtained voltage
- V: estimated voltage
- α: regularization term
- W: regularization matrix, which in the case of standard Tikhonov regularization is the identity matrix

The drawbacks of applying this method in EIT reconstruction include its high computational demands, sensitivity to initial guesses for impedance distribution, potential linearization errors when modeling impedance changes, non-guaranteed convergence, dependency on choosing an appropriate regularization term and susceptibility to measurement noise. An accurate parameter choice is vital to secure the algorithm's precision [15].

2.3 Generative Adversarial Networks

To understand the process of GAN design and its architecture, it is valuable to understand the architecture and operation of autoencoders. Beginning with their fundamental components, autoencoders consist of nodes, with their corresponding inputs, weights, biases, activation functions, and outputs. These fundamental node components are organized into layers and trained for learning and mapping complex relations through the modification of their weights. Consequently, the neural network contains three primary sections: the input layer, the output layer, and the hidden layers [16].

In terms of their design, the autoencoders consist of a pair of linked networks, known as the decoder and encoder. The operation of these two parts is [17]:

1. The encoder processes a vector $x \in R^d$ with input features into $y^* \in R^p$, which is a vector in a latent space.
2. From this, the decoder block uses a generated vector $y = y^* \in R^p$ to process it into $x^* \in R^d$.

The desired outcome is that this result closely resembles the initial input. Both the encoder and decoder can consist of multiple hidden layers that incorporate nonlinearities, creating a deep or stacked autoencoder. This architecture enhances its ability to learn and capture

complex nonlinear relationships, which is particularly useful in tasks like EIT reconstruction. Fig. 2 illustrates a basic representation of an autoencoder's basic structure.

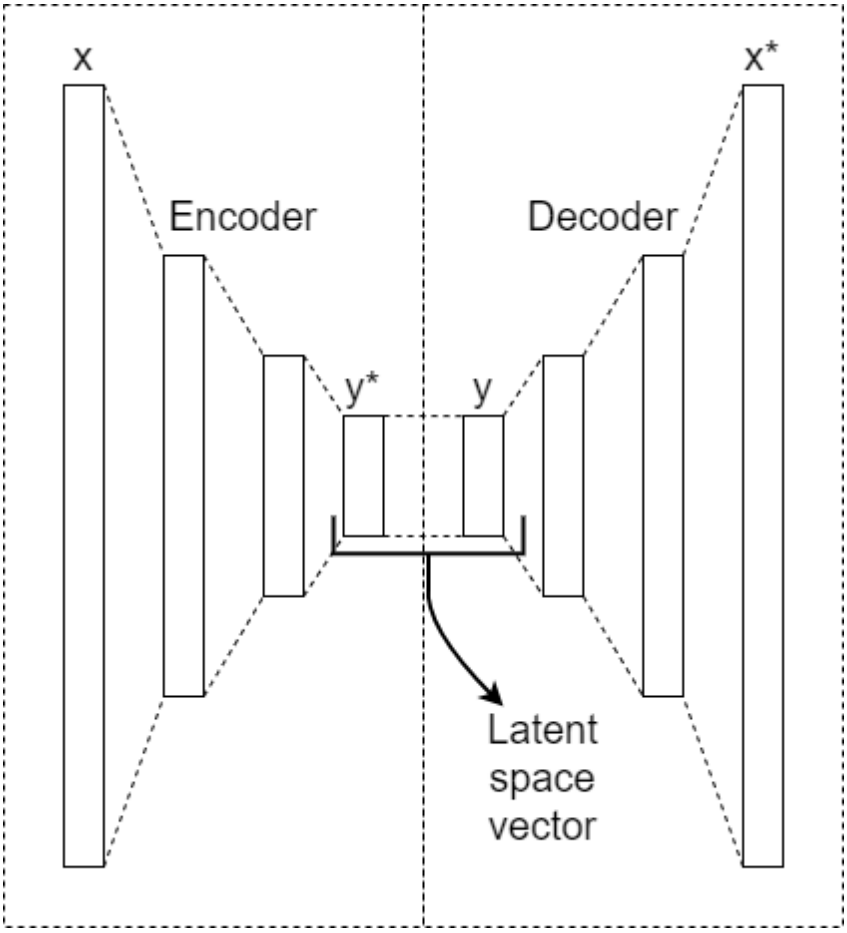


Fig. 2. Basic autoencoder structure

For the GAN architecture, the same decoder works as a generative model, often referred to as the generator block. Its role is to take a latent vector and produce an output of a higher dimension [18]. To evaluate the quality of the generated images, the architecture includes a discriminator block. The overall structure undergoes progressive training where, initially the generator outputs a synthetic vector x^* aiming to closely resemble the true conductivity at the domain. The image is produced from an input z, which for implemented GAN, corresponds to the obtained voltages at the electrodes. Subsequently, the synthetic input x^* is evaluated by the discriminator and a classification error is obtained. Here the

generator's task is to enhance the classification error by modifying its internal parameters through the overall error backpropagation. Throughout these iterations, an improvement in the generator's ability to produce data that convincingly resembles the data provided as expected output is done. The improvements allow it to mislead the discriminator effectively with realistic outputs [19].

The training procedure and internal structure of a basic GAN are depicted in Fig. 3. Here the backpropagation for training is represented as the discriminator output.

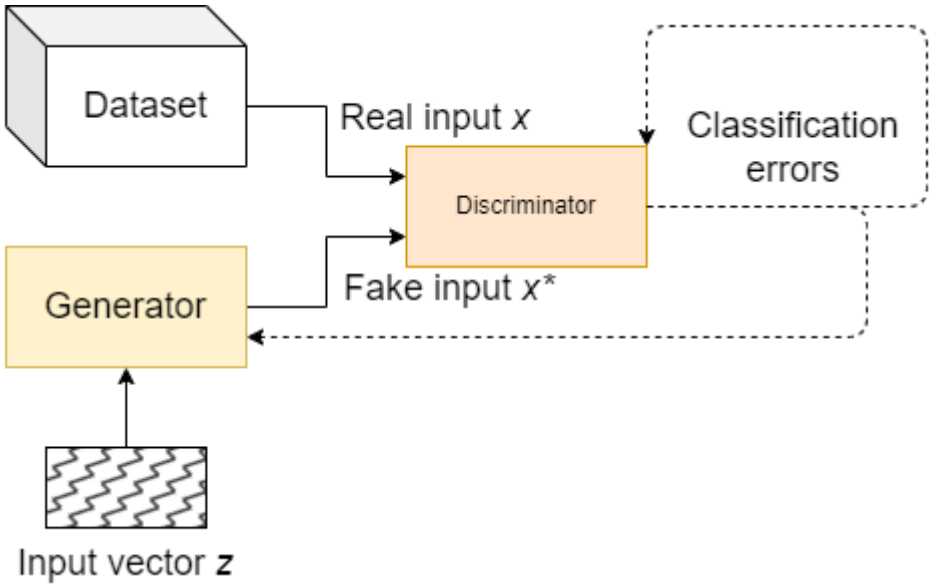


Fig. 3. Architecture and training process of GAN

The training objective function follows the minmax relationship [20]

$$minmax\, V(D,G) = E_{x\sim p_{data}(x)}\left[log\left(D(x)\right)\right] + E_{z\sim p_z(z)}\left[log\left(1-D(G(z))\right)\right] \quad (13)$$

here $p_{data(x)}$ represents the distribution of the real data, $p_z(z)$ the distribution from the noise, E the average value, G and D represents the generator and discriminator blocks correspondingly. The primary objective of the D block is to improve its capability to differentiate between generated and fake samples, whereas the G block aims to minimize

the performance of D. The D block updates involve adjusting its parameters using the loss function gradient with respect to G and updating G by the gradient of the loss with respect to D. The iterative process continues to a point where G generates data indistinguishable from the original dataset.

To update the parameters of G and D, the ADAM optimization method is employed. ADAM, as a better alternative to stochastic gradient descent (SGD), incorporates adaptive learning and momentum for faster convergence. It calculates its learning rates by maintaining an exponential average of square and past gradients. Additionally, ADAM incorporates a momentum component to mitigate fluctuations of gradients in the training. The mathematical expressions for the ADAM optimization method are as follows [21]:

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t \quad (14)$$

$$v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2 \quad (15)$$

$$\widehat{m_t} = \frac{m_t}{1 - \beta_1^t} \quad (16)$$

$$\widehat{v_t} = \frac{v_t}{1 - \beta_2^t} \quad (17)$$

$$\theta_t = \theta_{t-1} - \frac{\alpha \cdot \widehat{m_t}}{\sqrt{\widehat{v_t}} + \varepsilon} \quad (18)$$

where:

- m_t: mean(1st moment estimate) for gradients during tth iteration
- v_t: uncentered variance(2nd moment estimate) for gradients during tth iteration
- g_t: gradient during tth iteration
- β_1 and β_2: respectively m_t and v_t rates of decay
- $\widehat{m_t}$ and $\widehat{v_t}$: estimates m_t and v_t with bias-correction during tth iteration

- θ_t: tth iteration weight value
- α: step size or learning rate
- ε: constant to prevent the division by zero

Both the loss functions of G and D are differentiable, making it possible to utilize their gradients for parameter updates with the ADAM optimizer. Employing ADAM to optimize, enhances the GAN response by providing more efficient learning with a faster convergence of G and D towards a solution. ADAM has demonstrated effectiveness in managing complex data that address challenges associated with sparse gradients, a common issue during GAN training. As a summary, ADAM optimization holds a crucial place for the GAN process of training, by simplifying the process of optimization, and enhancing the overall performance of its networks [22].

2.3.1 pix2pix GAN

The pix2pix GAN is a conditional model that learns how an input image corresponds to its corresponding output image by optimizing a loss function to strengthen this connection. This model is renowned for its versatility in tasks such as reconstructing objects from edge maps, adding color to grayscale images, and performing various other image processing tasks [23]. The architecture diagram for the pix2pix GAN is illustrated in Fig. 4.

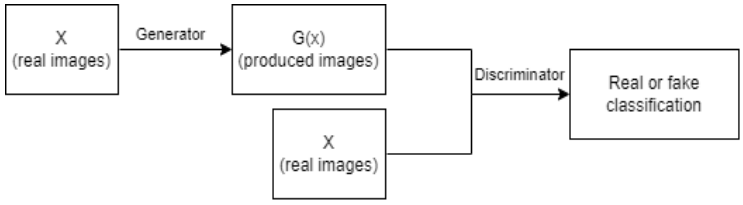


Fig. 4. pix2pix GAN architecture diagram

The pix2pix GAN model employs a U-Net style generator, which sets it apart from other

GANs. Instead, it utilizes a reference image and a target image as inputs, integrating dropout layers to introduce noise during both the training and testing phases. In the case of the discriminator model, a PatchGAN architecture is employed. This deep convolutional neural network specializes in classification. However, rather than evaluating the whole image, it processes segments of the image. Typically, a patch of 70 by 70 pixels is used, and the overall classification results are derived by averaging the classifications obtained for each patch [24]. The loss function for the pix2pix GAN is

$$L(G, D) = L_{cGAN}(G, D) + \lambda L_{L1}(G) \qquad (20)$$

where:

- L_{cGAN}: conditional adversarial loss between generator and discriminator
- L_{L1}: cycle-consistency loss using L1 norm
- λ: hyperparameter to regulate how much L1 error weight is enforced

2.3.2 Attention Mechanism

The attention mechanism, introduced in 2014, presented an innovative approach to overcome the limitations of traditional machine translation models. In this model, the encoder is built using a bidirectional recurrent neural network (RNN), while the decoder incorporates a module for both decoding and alignment. What sets the attention mechanism apart is its ability to dynamically focus on different segments of the source sentence during translation, allowing it to adapt and attend to the relevant information as needed [25]. This joint learning of alignment and translation results in a significant improvement in translation quality, making it more adept at handling lengthy sentences

and capturing complex dependencies. Fig. 5 illustrates the original attention mechanism model for the processing of natural language [24].

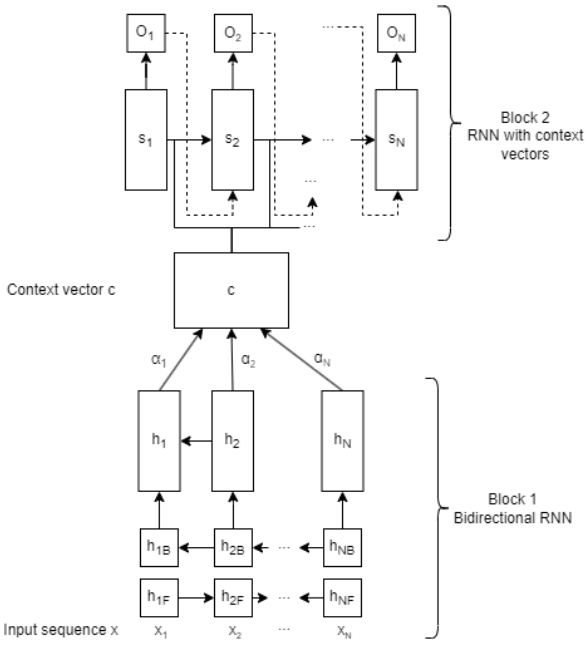


Fig. 5. Proposed model in [24] with a sequence(x) as input

The model takes an input sequence (x) and assigns a weight to each of its elements. Subsequently, the bidirectional RNN (represented as block 1) processes the sequence x in both forward and reverse directions, allowing it to capture any dependencies within the sequence. This process yields two hidden states, h_B and h_F, resulting from the forward and reverse processing, respectively. These two states are concatenated into a single state, h. All elements in h are then multiplied by attention weights α, which are determined by an alignment model that assesses the relationship between positions in the input and output [25]. The combination of h and α produces a context vector, which serves as input for block 2 as depicted in Fig. 5. In this block, a RNN utilizes the context vector, previous outputs, and hidden states to generate predictions.

3. Materials and methods

3.1 Two-Phase Flow

To assess the effectiveness of the developed model, a combination of simulations and experiments is employed. The study involves a cylindrical container with a 4 cm radius, simulating a two-phase flow scenario with voids and bubbles. For numerical simulations, a domain with 32 electrodes is employed, with a contact impedance of 0.005 Ωcm^2. A pair of different mesh structures are used to prevent the "inverse crime" issue [11]. Boundary voltage measurements are taken using an unstructured mesh consisting of 2409 nodes and 4560 triangular elements. Resistivity distribution estimations are carried out using a structured mesh with 635 nodes and 1140 triangular elements. This approach is applied in both numerical simulations and experimental setups.

3.2 Piezoresistive fabric

To characterize a piezoresistive material, a Medtex p130b piezoresistive fabric is taken as a model. This is a silver plated knitted fabric with polyamide and elastomer as raw materials. Numerical simulations are conducted within a rectangular domain with 16 electrodes, each with a contact impedance of 0.005 $\Omega\mathrm{cm}^2$. To prevent the "inverse crime," two different mesh structures are employed. Data processing tools include the Eidors toolbox [26], as well as MATLAB and Python. To generate various training cases with identical dimensions and parameters as the piezoresistive fabric, a MATLAB code is developed. The generated simulation cases serve as inputs for the proposed neural network models.

3.3 Evaluation Metrics

To evaluate the performance of the proposed GAN model in image reconstruction, several performance metrics are employed, including the relative size coverage ratio (RCR), structural similarity index (SSIM), relative image error (RIE) and correlation coefficient (CC). The SSIM considers differences in structure, luminance and contrast between two images, where values that approximate 1 indicate both images are similar. RCR gives a quantitative assessment of the inclusion volumes recovery, where 1 represents a perfect match between the real and obtained volumes. CC assesses a linear relationship between both the reconstruction and the actual target. Finally, RIE makes a comparison of the same pair of images but calculates the norm, with smaller values indicating better reconstruction quality. Definitions for these metrics are [27-29]

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (21)$$

$$RCR = \frac{CR}{CR_{True}} \quad (22)$$

$$CR = \frac{Inclusion\ volume}{Target\ volume} \times 100\% \quad (23)$$

$$CC = \frac{\sum_{i=1}^{N}[(p_i - \underline{p})(\widehat{p_i} - \underline{\hat{p}})]}{\sqrt{\sum_{i=1}^{N}(p_i - \underline{p})^2 \sum_{i=1}^{N}(\widehat{p_i} - \underline{\hat{p}})^2}} \quad (24)$$

$$RIE = \frac{\|x - \hat{y}\|}{\|x\|} \quad (25)$$

where:

- x: original image
- y: generated image

- μ_i: image i mean value
- σ_i: image i variance
- σ_{ij}: images i and j covariance
- C_1 and C_2: variables to avoid division by zero
- N: quantity of pixels
- p_i: pixel i value, from original image
- $\widehat{p_i}$: pixel i value, from reconstructed image
- $\underline{p}$: variable p mean

4. Two-phase flow study

Table 1 offers information regarding the training setup for the implementation of the proposed GAN model. This includes details on the dataset used, the architectural design, and the hyperparameters employed. The model's base is the conventional GAN, but it has been customized in its layers to work with the vector that contains the voltage measurements and the desired output image dimensions.

Table 1. Information on the dataset, hyperparameters, and numerical and experimental model

Parameter	Value
Batch size	16
Input layer neurons	128
Hidden layer neurons	12 500
Output layer neurons	16 384
Numbe of epochs	200
Numerical cases training set size	30 132
Experimental cases training set size	25 792
α	0.002
β_1	0.05
β_2	0.999

The training datasets are created through the use of MATLAB and EIDORS software [26], incorporating different target positions and varying sizes and shapes. In the case of the experimental model, a smaller number of cases are generated due to the predefined shapes. The training datasets can be enlarged as necessary to introduce additional variations in shape reconstruction.

In the simulation scenarios, the background is depicted as a 300 Ωcm liquid, while targets are defined as air bubbles or voids with a considerably higher 2000 Ωcm resistivity. The research explores situations featuring one, two, and four circular targets within the region to evaluate the performance of the proposed method in reconstruction. Furthermore, an additional case is examined, involving a shape not present in the training, such as a square.

The evaluation involved experimenting with different neuron quantities in the hidden layers of both the generator and discriminator. The results reveal that employing 12,500 neurons produces effective reconstruction results. Fewer than 12,000 neurons fail to lead to the outputs of average probability to 0.5 in the training instances while using more than 13,000 neurons leads to model overfitting. In Fig. 6(a), the two average probabilities when using 10,000 neurons are presented. In Fig. 6(b), a model overfitting is showcased, where distinct inputs (as shown in row 1) result in an identical output (as seen in row 2).

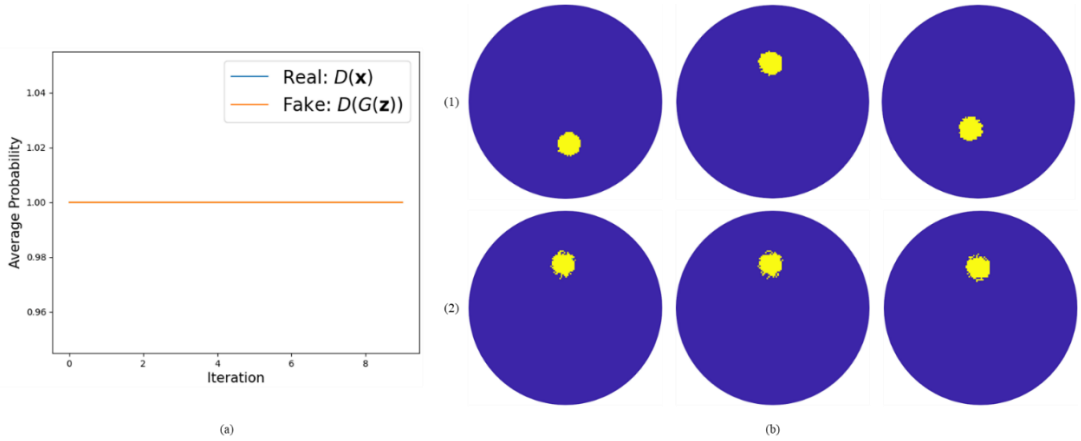


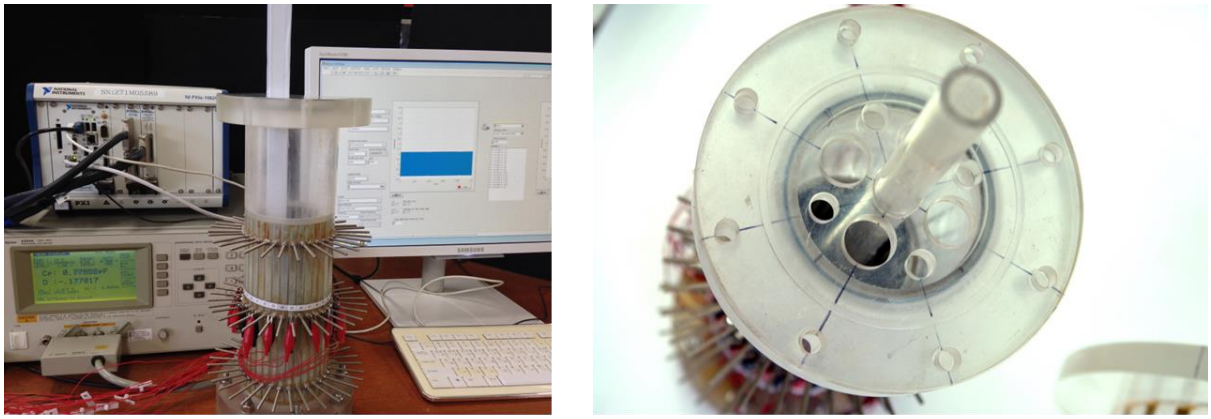
Fig. 6. Behavior of model training with varying the quantity of neurons (a) 10,000 neurons average probability (b) 13 000 neurons model result, where (1) represents the target location and (2) shows similar outputs due to model overfitting

For hyperparameters, the tuning of the ADAM optimizer involves adjusting α and β_1 based on the convergence of model probabilities, while β_2 remains at its default value since it has minimal impact on the outcomes. During testing, the trained model demonstrates a reconstruction speed of 100 ms, and the process of training is carried out on the Google Colab platform [30].

Regarding the neural network (NN), is composed of densely connected layers with a neuron configuration of 128-256-128, using ReLU activation for most layers and linear activation for its output layer. The deep neural network (DNN) incorporates densely connected layers along with dropout layers and features a neuron configuration of 128-512-1024-256-128. Both the NN and DNN utilize as optimizer the Adam model with a learning rate set at 0.002. Their reconstruction speeds are approximately 138 ms for the NN and 153 ms for the DNN.

Furthermore, to evaluate the GAN model's performance in a real-world scenario, experimental data collected from a setup designed at a Jeju National University laboratory

was employed. This setup featured a cylindrical phantom with a 4 cm radius, with 32 electrodes on its internal surface. Within the phantom, cylindrical targets (plastic rods) with high impedance and a diameter of 1 cm and 2 cm to simulate defect locations, were positioned. Between these targets, it was filled with a 0.15% saline solution having a 330 Ωcm resistivity. The electrodes applied current in an opposite pattern, and the experiment was carefully conducted to keep the error rate for both applied current and measured voltages below 1%. The equipment used for the experiment included the Agilent Precision LCR meter 4284A for current injection and the Agilent 43970A for data acquisition. Additional information about the experimental setup is presented in Fig. 7.



(a) (b)

Fig. 7. Set-up for experimental model. (a) used phantom and processing hardware (b) top view of possible target locations for the experimental tank

4.1 Numerical Results

Fig. 8 presents different numerical methods used to produce target locations and corresponding reconstructions in various scenarios. These methods include modified Newton-Raphson (mNR), neural network (NN), deep neural network (DNN), and Generative Adversarial Network (GAN).

The neural network (NN) uses densely connected layers with a configuration of 128-256-128 neurons, employing Rectified Linear Unit (ReLU) activation and concluding with a linear activation at the output. On the other hand, the deep neural network (DNN) incorporates densely connected layers having in between dropout layers, featuring a neurons model of 128-512-1024-256-128. Both the NN and DNN utilize the Adam optimizer with 0.002 as the learning rate, and their respective processing speeds are approximately 138 ms and

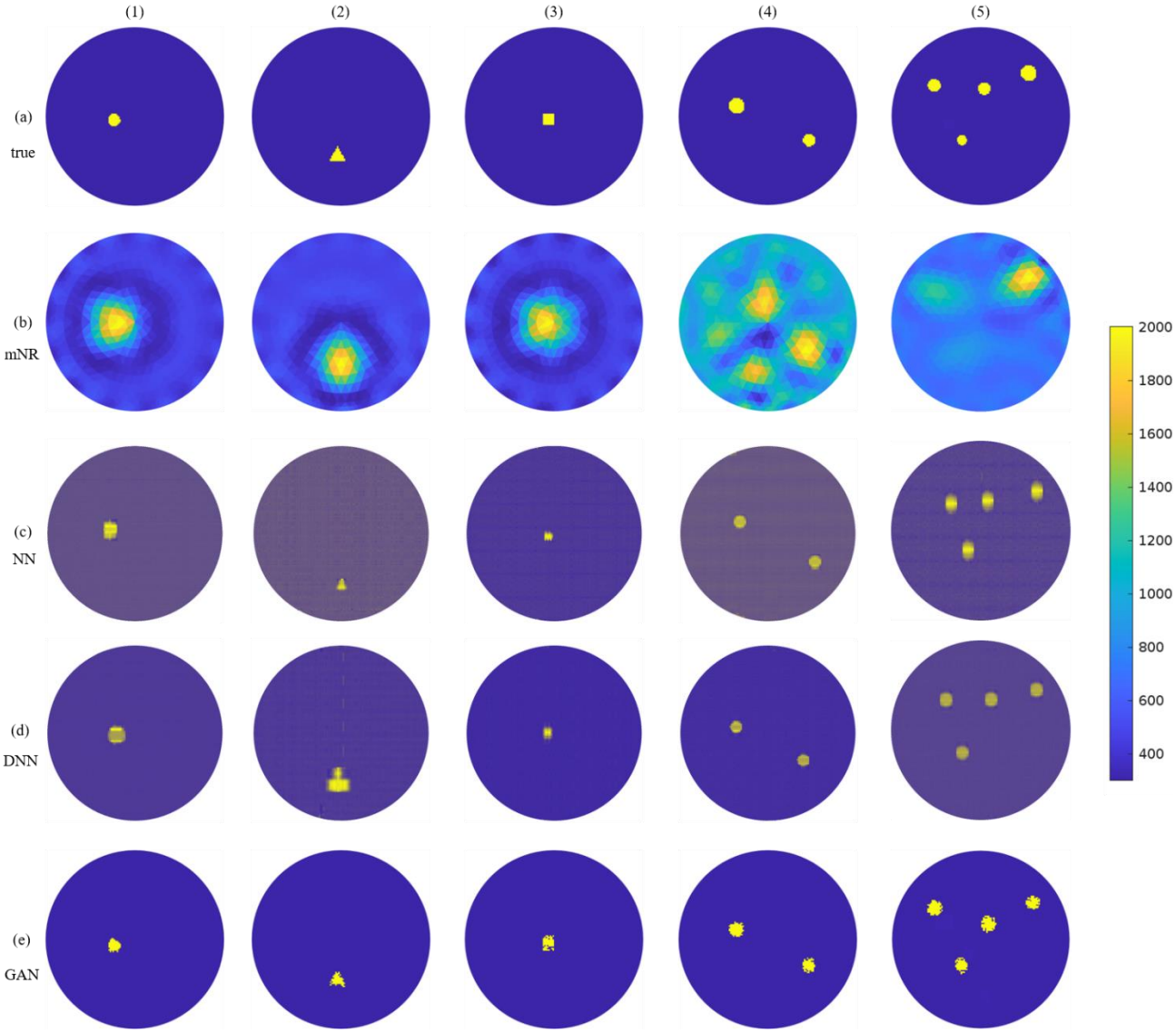


Fig 8. Numerical results for various two-phase flow shapes (a) location of the target (b) mNR reconstruction (c) neural network reconstruction (d) deep neural network reconstruction (e) GAN reconstruction, colorbar is included for reference in the reconstruction images, case (3) involves a shape not present in the dataset

153 ms.

Fig. 8 displays various validation scenarios. Cases (1) and (2) mimic defects resembling those encountered during the training phase. In case (3), a target shaped as a square, which was not part of the training dataset, is introduced. Cases (4) and (5) involve the presence of multiple targets within the domain.

To obtain the mNR reconstructions, five iterations are performed, with the inclusion of a regularization parameter α set at 5×10^{-6}. Additional iterations are not found to significantly enhance the quality of reconstruction. The reconstruction process with mNR takes approximately 18.365 seconds.

The mNR reconstructions shown in Fig. 8(b) reveal that the target's shape is unclear and the background is not homogeneous. In the scenario with two targets (case 4), the mNR reconstruction exhibits incorrect artifacts in areas that display uniformity in the real modeling. Furthermore, with the introduction of additional defects (case 5), the mNR reconstruction encounters difficulties in accurately representing all of them, as evident in the results.

On the other side, the NN and DNN models accurately identify the target locations but display slight blurriness and a non-uniform background. In the reconstructions produced by the GAN model, the background appears more uniform, primarily because the GAN learns from target location images. Furthermore, the shapes closely resemble those of the validation cases. Notably, even in the situation where a target shape is outside the dataset (case 3), the GAN model makes an effort to approximate the target shape based on the patterns it has learned. Consequently, it correctly identifies the target location and partially

reconstructs the shape, with some minor noise near the target.

Figure 9 presents two situations in which noise is added to the input signal during the GAN model's reconstruction process. In both cases, noise levels vary from 0% to 4%. The evaluation considers the Structural Similarity Index (SSIM), which accounts for elements like image similarity and texture. The SSIM difference between the noise-free scenario and the 4% noise scenario is determined to be less than 1.2% for both cases.

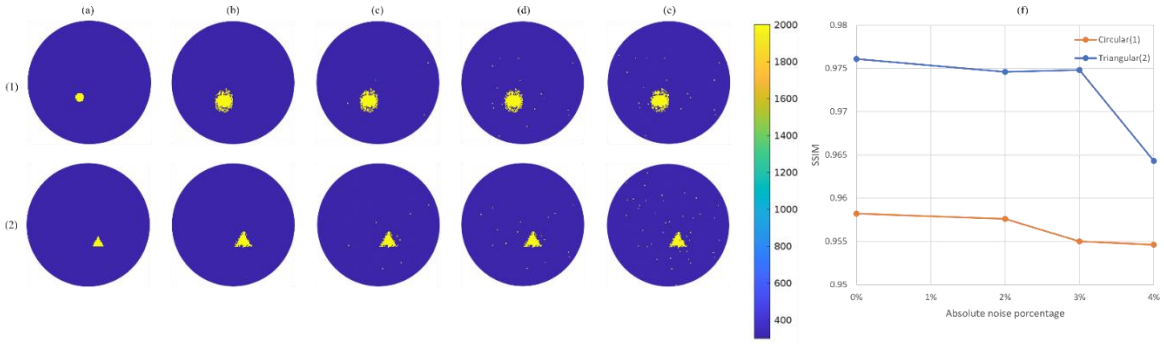


Fig. 9. Reconstruction result of GAN model when input signal is subjected to added noise (a) target location (b) 0% (c) 2% (d) 3% (e) 4% (f) SSIM under the various levels of noise, a color reference bar is included for the reconstructions

In summary, the reconstructions produced by the GAN model and deep learning techniques reveal a drawback characterized by the presence of noise near the target's boundary. Nevertheless, this challenge has the potential to be mitigated by integrating advanced modules or adopting other proposed GAN architectures, such as CycleGAN and LAPGAN.

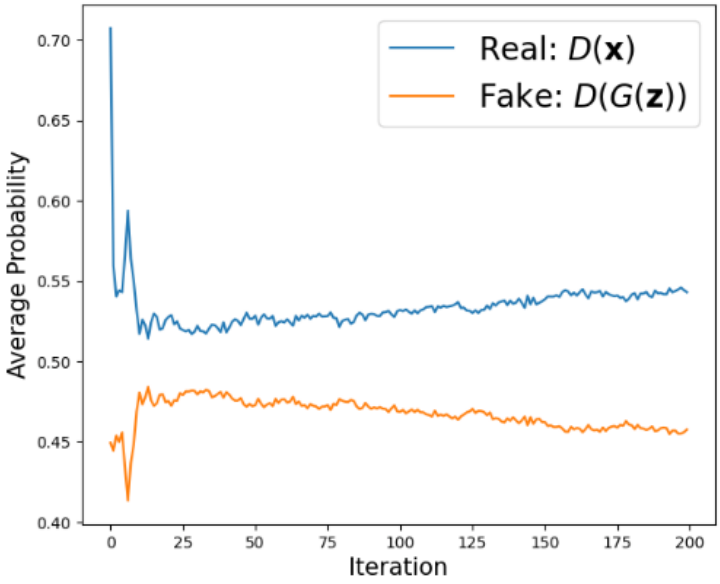


Fig. 10. Output of the discriminator's probability during training of the numerical model

Figure 10 illustrates the discriminator probability outputs during the training of the GAN model. As expected, they converge to 0.5, which means the discriminator has difficulty in confidently discerning between real and generated outputs.

To perform a comparative assessment of the numerical results, Table 2 provides the evaluation metrics values of the dataset for validation. As expected, considering the notably uniform backgrounds depicted in Figure 8(c), (d), and (e), the metrics for the deep learning models typically yield superior results compared to mNR. It's worth noting that the RIE metric highlights a significant difference, with GAN outperforming the other methods by an order of magnitude.

Table 2. Evaluation metrics for the numerical study

Case	mNR				NN				DNN				GAN			
	SSIM	RCR	CC	RIE	SSIM	RCR	CC	RIE	SSIM	RCR	CC	RIE	SSIM	RCR	CC	RIE
1	0.8715	0.1555	0.918	0.2792	0.6901	0.424	0.847	0.321	0.874	0.155	0.889	0.272	0.996	0.3856	0.9953	0.0541
2	0.8662	0.206	0.918	0.2883	0.5983	0.265	0.815	0.294	0.857	0.365	0.869	0.355	0.997	0.4165	0.9972	0.0419
3	0.8685	0.1792	0.916	0.2783	0.8559	0.259	0.885	0.261	0.913	0.28	0.896	0.275	0.996	0.3932	0.9959	0.0503
4	0.5188	0.1843	0.656	0.5231	0.5104	0.352	0.885	0.356	0.897	0.341	0.812	0.273	0.99	0.6508	0.9983	0.0606
5	0.7342	0.1803	0.81	0.4256	0.7885	0.218	0.908	0.246	0.815	0.386	0.921	0.231	0.943	0.636	0.9456	0.0852
6	0.7356	0.8213	0.915	0.3556	0.7535	0.655	0.865	0.234	0.814	0.597	0.857	0.239	0.848	0.6011	0.8854	0.0217
7	0.8662	0.8351	0.92	0.4585	0.3274	0.619	0.848	0.255	0.851	0.624	0.848	0.215	0.83	0.5947	0.9653	0.0221
8	0.7231	0.7221	0.643	0.2574	0.6471	0.456	0.799	0.268	0.819	0.345	0.867	0.292	0.95	0.4875	0.7845	0.0259
9	0.8274	1.0012	0.652	0.2451	0.5947	0.397	0.801	0.301	0.825	0.385	0.818	0.228	0.914	0.5646	0.7927	0.0294

Regarding the deep learning models metrics, they exhibit superior performance compared to mNR, as anticipated from the smoother backgrounds evident in Figure 5(c), (d), and (e). Notably is the considerable distinction of difference of one magnitude, revealed by the RIE metric between the GAN model and the other methods. In the scenarios displayed in Figure 7, which involve varying levels of noise, the SSIM difference between the noise-free scenario and the 4% noise scenario is minimal, registering a value below 1.2%.

4.2 Experimental Results

Figure 11 presents the reconstructions obtained from the different implemented methods in various experimental configurations, highlighting the location of the target. For mNR, the reconstructions were generated with a regularization parameter set at $\alpha=5\times10^{-6}$ and five iterations.

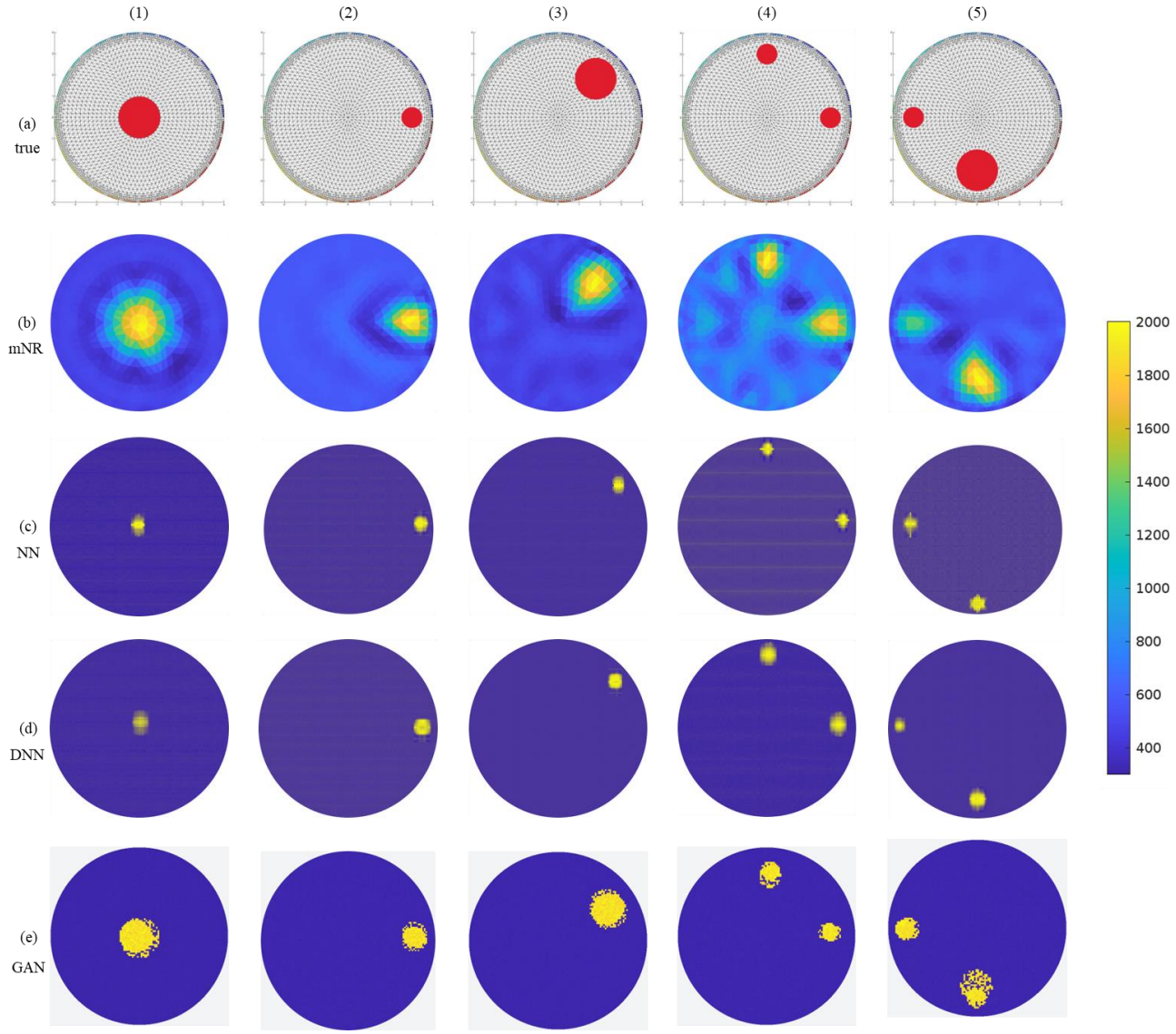


Fig. 11. Experimental results for two-phase flow (a) location of the target (b) mNR reconstruction (c) neural network reconstruction (d) deep neural network reconstruction (e) GAN reconstruction, a color reference bar is included for the reconstructions

Figure 11(a) displays the positions of the targets in the experimental dataset. However, the reconstructions produced by mNR, shown in Figure 11(b), exhibit an uneven background, similar to what was seen in the numerical cases presented in Figure 8(b).

In the case of NN and DNN, Figures 11(c) and 11(d) illustrate the experimental data reconstructions. As in the numerical scenarios for these two methods, the correct

identification of target positions is correct. However, in most of the reconstructions, noticeable irregularities in the background can be observed. Additionally, there is a variation in the shape of the results based on the target's position, exemplified in Figure 11(c) and Figure 11(d).

For the reconstructions produced by the GAN model in Figure 11(e), they display a smoother and more uniform background, which can be attributed to the model's capacity to learn from images of target locations. Additionally, targets' shapes closely approximate those in various validation scenarios.

Figure 12 illustrates the discriminator probability output during the training of the GAN model. As observed in Figure 10, these values tend to be around 0.5. This suggests that the discriminator cannot confidently distinguish between whether the outputs from the generator are real or fake.

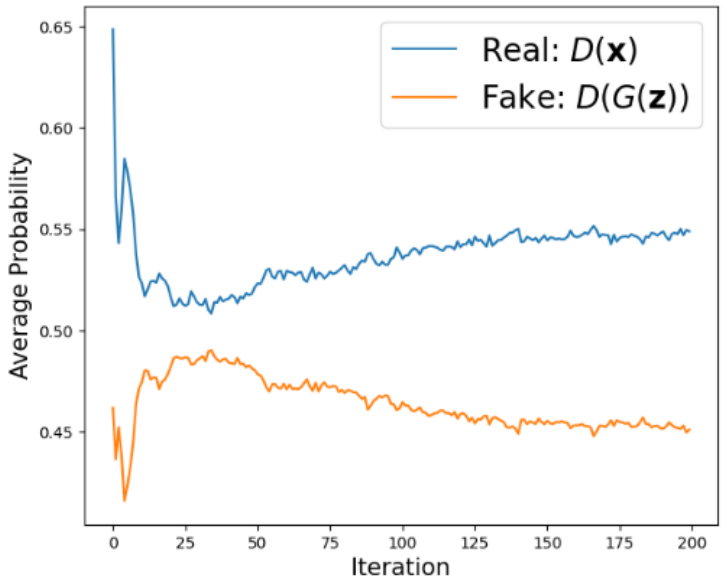


Fig. 12. Output of the discriminator's probability during training of the numerical model

Table 3. Evaluation metrics for the experimental study

Case	mNR		NN		DNN		GAN	
	SSIM	RCR	SSIM	RCR	SSIM	RCR	SSIM	RCR
1	0.8715	0.1555	0.6901	0.4241	0.8742	0.1548	0.9955	0.3856
2	0.8662	0.206	0.5983	0.2647	0.8574	0.3653	0.9968	0.4165
3	0.8685	0.1792	0.8559	0.2589	0.9129	0.28	0.9958	0.3932
4	0.5188	0.1843	0.5104	0.352	0.8971	0.3412	0.9901	0.6508
5	0.7342	0.1803	0.7885	0.2184	0.8154	0.3861	0.9433	0.636

Table 3 presents a comparison of the results from the validation cases in the experimental setting. In these experimental scenarios, the proposed GAN model demonstrates superior performance, with approximately a 10% improvement in SSIM compared to the other methods. In some cases, RCR shows a similar difference of 0.11 to 1 between mNR and GAN. However, it's worth noting that RCR may not necessarily reflect similarity in quality, and GAN results can be considered closer to the true target in terms of color and shape.

In terms of quantitative performance, the GAN model outperforms the other methods in experimental cases, achieving around a 10% improvement in SSIM. While RCR values show a similar difference of 0.11 to 1 between mNR and GAN in some instances, it is important to note that RCR may not necessarily reflect similarity in quality, so results obtained from the GAN model can be considered closer to the true target in terms of color and shape. CC and RIE metrics are not considered for experimental cases as the exact impedance values of targets from experimental measurements are not known.

5. Piezoresistive fabric study

For numerical simulations, rectangular meshes with the same dimensions as the fabric used in experimental experiments are employed. To avoid the "inverse crime" issue, distinct meshes are used for forward and inverse computations. Initially, reconstruction is performed using mNR with 5 iterations and a regularization parameter of $\alpha=5x10^{-6}$, determined through the evaluation of reconstruction results. After the mNR reconstruction, the resulting image serves as input for a trained pix2pixGAN model. Various numerical cases are assessed, including single-target scenarios, out-of-dataset target shapes, and cases with two targets.

To simulate resistance changes, conductivity values are increased six times in areas with pressure points. Electrodes are evenly spaced based on fabric dimensions, and a 100mA alternating current is applied between adjacent electrode pairs. Voltages resulting from these injections are recorded for each electrode, totaling 16 electrode pairs and 256 voltage measurements per data frame. Numerical cases also explore scenarios with the addition of 0.5 relative noise to model instrument and measurement errors.

The proposed pix2pix GAN model architecture is based on the model introduced in [24], with modifications in layer dimensions and the inclusion of an attention mechanism. Detailed hyperparameters and dataset-related information can be found in Table 4.

Table 4. pix2pix GAN architecture and dataset details

Hyperparameter/Data set detail	Value
Input layer size	256x256x3
Output layer size	256x256x3
Encoder layers	C64-C128-C256-C512-C512-C512-C512-C512
Decoder Layers	TC512-TC512-TC512-TC512-TC256-TC128-TC64
Dataset size	9787

[1] *C: convolution, TC: transposed convolution

Fig. 13 presents evaluation cases taken each 200 iterations during the training of the pix2pix GAN with the attention mechanism model. It can be seen in this iterative training that the model progressively learns to generate the shape and location of the true target, taking the mNR reconstructions as input.

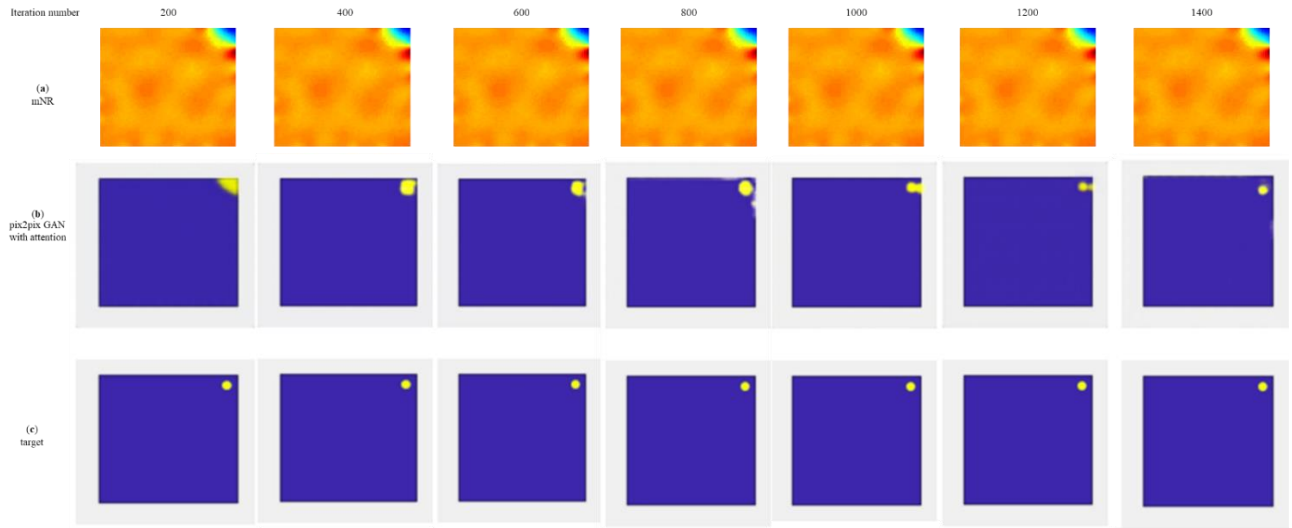


Fig. 13. Evaluation cases for pix2pixGAN with attention mechanism model during training (a) reconstruction by mNR (b) reconstruction by pix2pix GAN with attention mechanism (c) real target location

The experimental arrangement is depicted in Fig. 14 and comprises a Medtex p130b configured to allow the application of pressure points while the fabric remains stationary. The fabric's dimensions are 21x29, which is appropriate for a real application sensor.

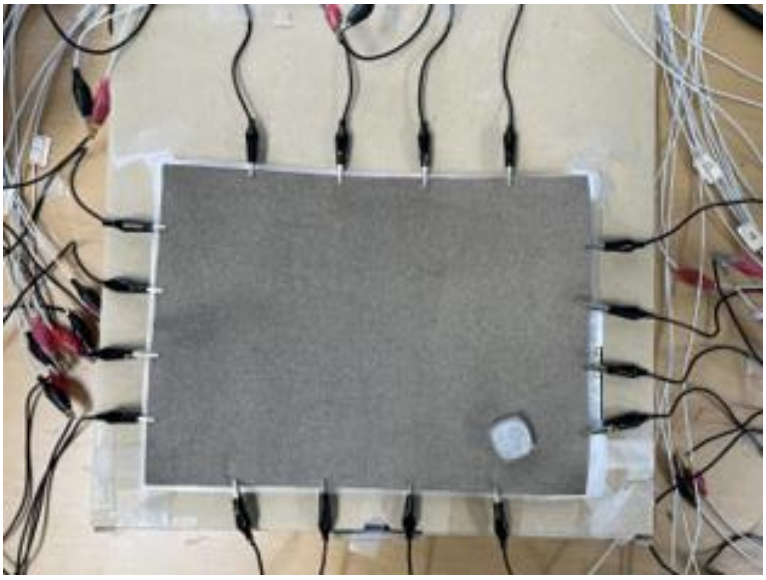


Fig. 14. Experimental setup

Laboratory experiments are carried out with an EIT setup consisting of an Agilent 4284A precision LCR meter serving as a stable current source and a National Instruments system (NI PXI-1042Q, National Instruments Corporation, Austin, TX, USA) utilized for data acquisition to measure the resulting voltages. This setup incorporates a total of 16 electrodes, evenly distributed along the fabric's boundary, with four electrodes placed on each side of the fabric.

The reconstruction process employs difference imaging, which begins with obtaining an initial measurement in the absence of pressure points to serve as a reference. To simulate the presence of pressure points, small objects are positioned on the fabric.

5.1 Numerical Results

The results of post-processing for mNR reconstruction are depicted in Figure 15. In Figure 15(a), each row displays the actual locations and shapes of one or two targets. Figure 15(b) shows the outcomes of mNR reconstruction, while Figure 15(c) and Figure 15(d) display the results of post-processing using pix2pix GAN without and with an attention mechanism,

respectively.

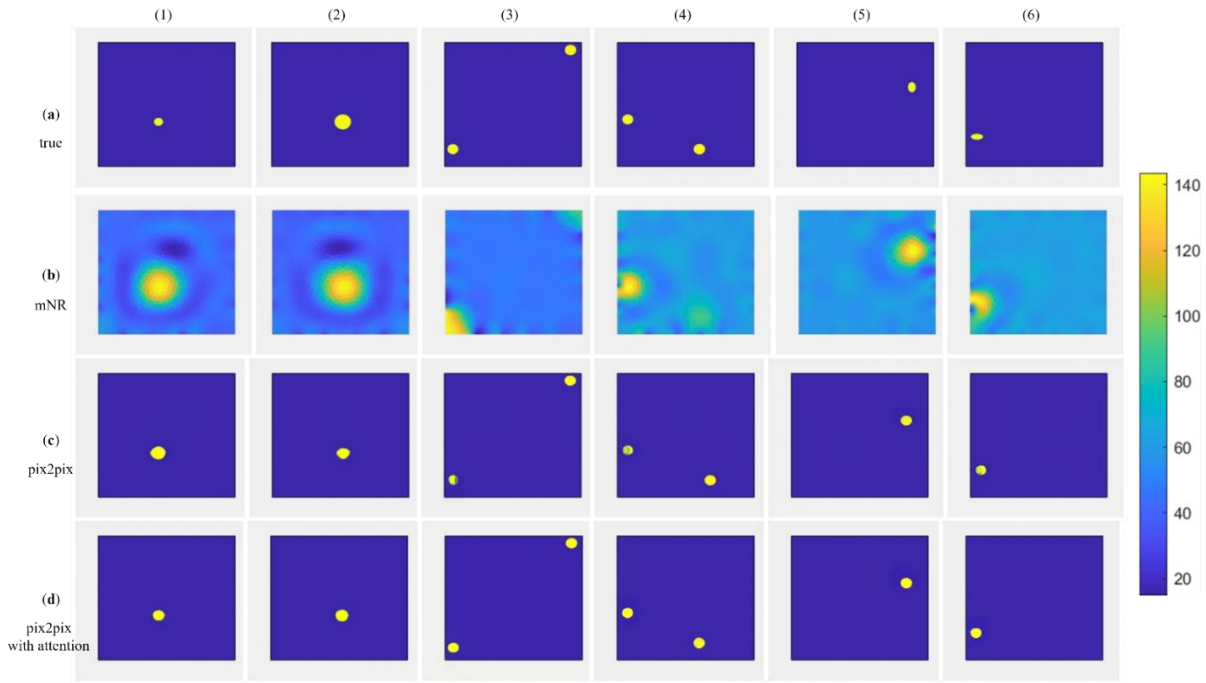


Fig. 15. Numerical results for the pix2pix GAN models (a) target location, (b) the reconstruction achieved using mNR, (c) the reconstruction obtained via the pix2pix model, and (d) the reconstruction accomplished using the pix2pix model, and (d) the reconstruction accomplished using the pix2pix model with an attention mechanism, a color reference bar is included for the reconstructions

For cases (1-2) with one target, (3-4) with two targets, and (5-6) with elliptical-shaped targets, mNR reconstruction yields indistinct target shapes and uneven background, consistent with the limitations of this method. In the two-target cases (3-4), both targets do not align with the same colorbar scale. When dealing with elliptical targets, mNR fails to represent the target's shape compared to circular targets.

Pix2pix GAN reconstructions, in all cases, correctly position the target but exhibit challenges in accurately reproducing the target's shape and clarity. Notably, in cases (5-6) featuring out-of-dataset elliptical-shaped targets, pix2pix GAN reconstructs them as circular targets. However, when employing pix2pix GAN with an attention mechanism, there is an improvement in target clarity compared to Figure 15(c). Metrics for Figure 15 are presented

in Table 5.

Table 5. Evaluation metrics for Figure 15 evaluation cases

Case	pix2pix GAN			pix2pix GAN with Attention Mechanism		
	SSIM	CC	RIE	SSIM	CC	RIE
1	0.9929	0.9947	0.0492	0.9963	0.9982	0.0281
2	0.9937	0.9956	0.0447	0.994	0.9965	0.0399
3	0.9897	0.9936	0.0539	0.9967	0.9988	0.0239
4	0.9862	0.9879	0.074	0.9979	0.9997	0.0109
5	0.9944	0.9967	0.0433	0.9948	0.9973	0.0373
6	0.9943	0.9965	0.0391	0.9947	0.9975	0.0325

Results obtained using the same model but utilizing mNR images from a signal with 2% added noise as input are presented in Figure 16. The row in Figure 16(a) displays the actual positions and shapes of either one or two targets. Figure 16(b) shows the reconstruction achieved through mNR. In Figure 16(c) and Figure 16(d), the results of post-processing using pix2pix GAN and pix2pix GAN with an attention mechanism are presented, respectively.

In cases (1-2) involving a single target, (3-4) with two targets, and (5-6) featuring targets with an elliptical configuration, mNR reconstruction exhibits a less uniform background compared to the reconstructions in Figure 16(b). Pix2pix GAN demonstrates variable target clarity among different cases and occasionally includes an unclear additional target near the true target position, as seen in cases (2-3) and (5-6). Pix2pix GAN, when coupled with an attention mechanism, accurately localizes the targets and provides better target definition, with the exception of cases (5-6) where the target is less clearly defined compared to other scenarios. In Table 6 the metrics for Figure 16 are presented.

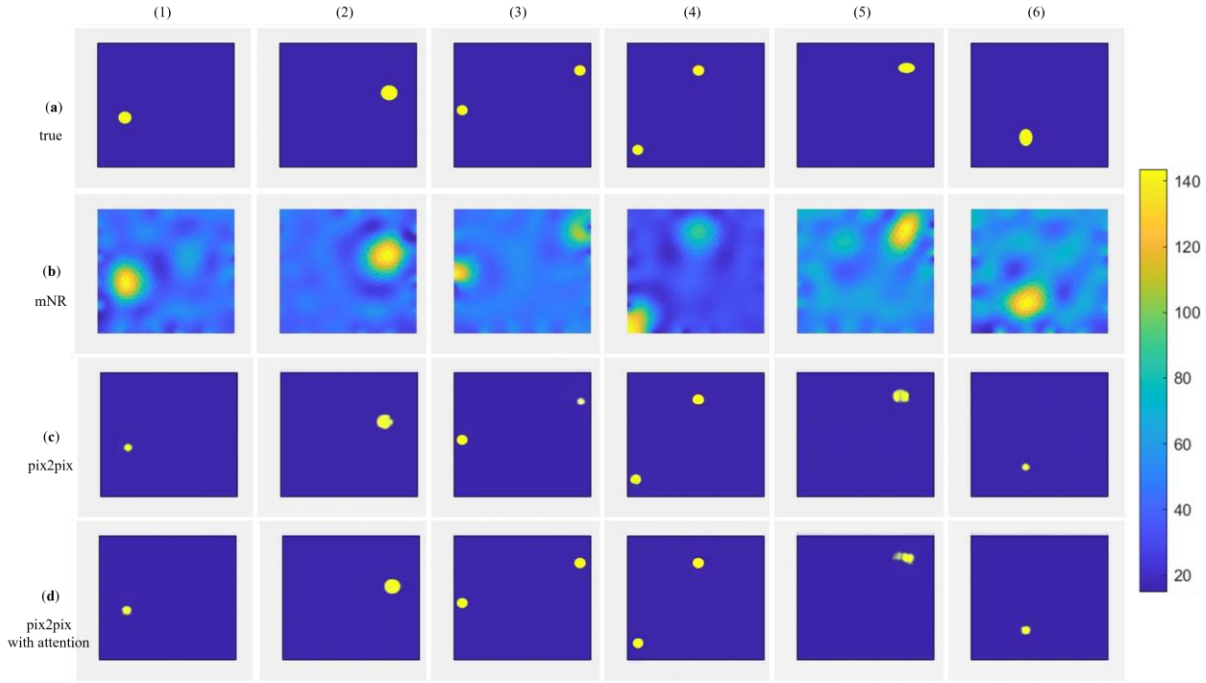


Fig. 16. Numerical outcomes with the inclusion of noise for the pix2pix GAN models (a) the actual target location, (b) the reconstruction achieved using mNR, (c) the reconstruction obtained via the pix2pix model, and (d) the reconstruction accomplished using the pix2pix model with an attention mechanism, a color reference bar is included for the reconstructions

Table 6. Evaluation metrics for Figure 16 evaluation cases

Case	pix2pix GAN			pix2pix GAN with Attention Mechanism		
	SSIM	CC	RIE	SSIM	CC	RIE
1	0.9924	0.9958	0.0436	0.9946	0.997	0.037
2	0.9873	0.9906	0.0652	0.9957	0.9988	0.021
3	0.9924	0.9963	0.0409	0.997	0.9997	0.011
4	0.9956	0.998	0.0286	0.9971	0.9994	0.015
5	0.9885	0.9928	0.0569	0.9877	0.9945	0.05
6	0.9879	0.9917	0.0613	0.9874	0.9929	0.057

5.2 Experimental Results

Figure 17 illustrates the target positions and the resulting reconstructions obtained using mNR with difference imaging and its post-processing with pix2pix GAN and pix2pix GAN with attention mechanism in experimental scenarios.

Fig. 17(a) provides the target positions in the experimental dataset. Evaluation metrics are not estimated for these reconstructions since there is no ground truth image for the different experimental positions. The mNR-generated reconstructions are shown in Fig. 17(b), which present a not homogeneous background. Figures 17(c) and 17(d) display the reconstructions of pix2pix GAN without and with attention mechanism respectively. Like the numerical cases, the target locations for mNR show noticeable background irregularities in most of the reconstructions. The pix2pix GAN model's reconstructions in Fig. 17(c) and Fig. 17(d) feature a more homogeneous background due to learning from the target location images, and the target shapes are well approximated due to the paired training.

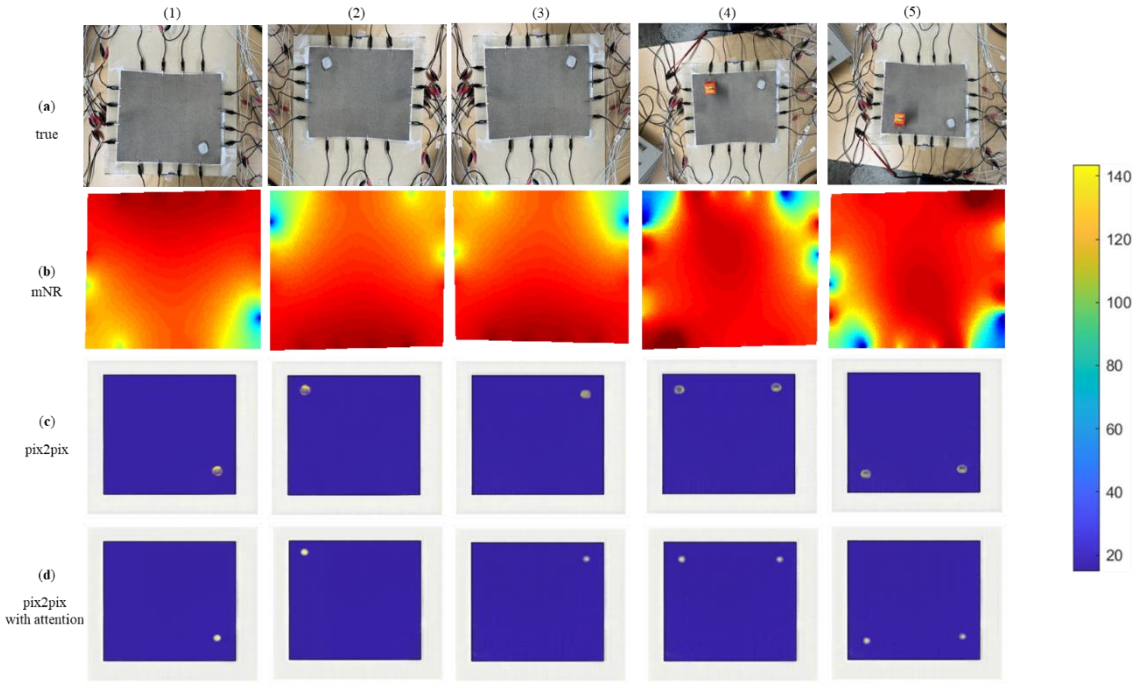


Fig 17. Experimental results for piezoresistive fabric. (a) target location (b) mNR reconstruction (c) pix2pix reconstruction (d) pix2pix with attention mechanism reconstruction, a color reference bar is included for the reconstructions

6. Future Works

The integration of Generative Adversarial Networks (GAN) with Electrical Impedance Tomography (EIT) provides numerous opportunities for reconstruction and image enhancement models. Other applications for data augmentation and generalization techniques can also be developed with the use of GANs.

Regarding the improvement of applied models, besides the attention mechanism, there can be variations in the network's internal structure, such as in the recurrent neural networks (RNN) or the addition of residual modules. Another field can be the integration of EIT with other imaging modalities through GANs for more comprehensive diagnostic capabilities in the medical and materials science fields.

7. Conclusions

In this study, models based on Generative Adversarial Networks were used for EIT reconstruction and image enhancement. Results were verified with both simulated and experimental cases and reviewed by visual and metric analyses. The quality of the generated images has been enhanced in comparison to conventional reconstruction techniques like mNR and other deep learning methods. Generative Adversarial Networks can offer a good potential for their implementation in further EIT related applications.

8. References

[1] Hussein, H. M. A., Ahmed, S. T. and Habeeb, L. J., 2020. Two-phase flow for gas-liquid, gas-solid, liquid-solid, and liquid-liquid in a horizontal smooth and turbulator conduit–a review. Journal of Mechanical Engineering Research and Developments, 43, pp.26-50, doi: 10.1515/cls-2022-0018.

[2] Eda, T. et al. 2013. Experimental study on liquid spread and maldistribution in the trickle bed reactor using electrical resistance tomography. Journal of Power and Energy Systems,7(2), pp.94-105, doi: 10.1299/jpes.7.94.

[3] Silvera-Tawil, D., Rye, D., Soleimani, M. and Velonaki M., 2015. Electrical Impedance Tomography for Artificial Sensitive Robotic Skin: A Review. IEEE Sensors Journal, 15(4), pp.2001-2016, doi: 10.1109/JSEN.2014.2375346.

[4] Xin, W., Zhu, F., Wang, P., Xie, Z., Tang, Z., and Laschi, C., 2023. Electrical Impedance Tomographic Shape Sensing for Soft Robots. IEEE Robotics and Automation Letters, 8(3), pp.1555-1562, doi: 10.1109/LRA.2023.3240368.

[5] Engl, H., Hanke, M. and Neubauer, A., 1996. Regularization of Inverse Problems. Mathematics and Its Applications, 375, Kluwer Academic.

[6] Bayford, R., 2018. Basic electrical impedance tomography. Bioimpedance in Biomedical Applications and Research. Springer Nature, pp.29-44, doi: 10.1007/978-3-319-74388-2_3.

[7] Somersalo, E., Cheney, M. and Isaacson, D., 1992. Existence and uniqueness for electrode models for electric current computed tomography. Siam Journal on Applied Mathematics, 52, pp.1023-1040.

[8] Boverman, G., Kim, B. S., Isaacson, D., and Newell, J. C., 2007. The complete electrode model for imaging and electrode contact compensation in electrical impedance tomography. Proceedings of 29th Annual International Conference of the IEEE EMBS, pp.3462-3465, doi: 10.1109/IEMBS.2007.4353076.

[9] Yang, Z., Li, Z., Yue, S., Ding, M. and Li, J., 2018. An efficient LBP algorithm for ET imaging by reusing measurements. Proceedings of 2018 13th WCICA, pp. 1425-1429.

[10] Cheng, K. S., Isaacson, D., Newell, J. C. and Gisser, D. G., 1989. Electrode models for electric current computed tomography. IEEE Transactions on Biomedical Engineering, 36(9) pp.918-924, doi: 10.1109/10.35300.

[11] Vauhkonen, M., 1997. Electrical impedance tomography and prior information, Ph.D. dissertation, Dept. App. Phys., University of Kuopio, Kuopio, Finland.

[12] Rao, L., et al., 1999. An efficient improvement of modified Newton-Raphson algorithm for electrical impedance tomography. IEEE Transactions on Magnetics, 35(3), pp.1562-1565, doi: 10.1109/20.767269.

[13] Khambampati, A. K., Rahman, S. A., Sharma, S. K., Kim, W. Y. and K. Y. Kim, 2022. Nonlinear difference imaging to image local conductivity of single-layer graphene using electrical impedance tomography. IEEE Transactions on Instrumentation and Measurements, 71(1), pp.1-12, doi: 10.1109/TIM.20 22.3147894.

[14] Nam, I. H., Kang, B. C., Kim, J. H., and Choi, B. Y., 2003. Modified regularized Newton-Raphson algorithm for electrical impedance tomography in region of interest. in Proceedings of ICCAS 2003, pp. 134-137.

[15] Lionheart, W. R. B., 2004. EIT reconstruction algorithms: pitfalls, challenges and recent developments. Physiological Measurement, 25(1), pp. 125, doi: 10.1088/0967-3334/25/1/021.

[16] Zheng, J., and Peng, L., 2018. An autoencoder-based image reconstruction for electrical capacitance tomography. IEEE Sensors Journal, 18(13), pp.5464-5474, doi: 10.1109/JSEN.2018.2836337.

[17] Raschka S., Liu, Y. H., Mirjalili, V., and Dzhulgakov, D., 2022. Generative adversarial networks for synthesizing new data. Machine Learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python, 1st ed. Birmingham, UK: Packt Pub. Ltd., ch. 17, pp. 589-611.

[18] Li, X. et al., 2017. An image reconstruction framework based on deep neural network for electrical impedance tomography. IEEE ICIP, pp.3585-3589, doi: 10.1109/ICIP.2017.8296 950.

[19] Langr, J., and Bok, V., 2019. Your first GAN: Generating handwritten digits. GANs in Action, Shelter Island, NY, USA: Manning Publications, ch. 3, pp.36-50.

[20] Prabhat, Nishant, and Vishwakarma, D. K., 2020. Comparative Analysis of Deep Convolutional Generative Adversarial Network and Conditional Generative Adversarial Network using Hand Written Digits. 4th International Conference on Intelligent Computing and Control Systems (ICICCS), pp. 1072-1075, doi: 10.1109/ICICCS48265.2020.9121178.

[21] Kingma, D. P., and Ba, J., 2014. ADAM: a method for stochastic optimization, arXiv:1412.6980.

[22] Radford, A., Metz, L., and Chintala, S., 2015. Unsupervised representation learning with deep convolutional generative adversarial networks, arXiv:1511.06434.

[23] Popescu D., et al., 2021. Retinal Blood Vessel Segmentation Using Pix2Pix GAN. 2021 29th Mediterranean Conference on Control and Automation (MED), pp.1173-1178. DOI: 10.1109/MED51440.2021.9480169.

[24] Bahdanau, D., Cho, K. and Bengio, Y., 2014. Neural machine translation by jointly learning to align and translate, arXiv preprint arXiv:1409.0473.

[25] Raschka, S., Liu, Y. H., Mirjalili, V., and Dzhulgakov D., 2022. Generative adversarial networks for synthesizing new data, Machine Learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python, 1st ed. Birmingham, UK: Packt Pub. Ltd., ch.17, pp.589-611.

[26] Adler, A., and Lionheart, W., 2005. EIDORS: Towards a community-based extensible software base for EIT.

[27] Nombo, J., Mwambela, A., and Kisangiri, M., 2014. A grey level fitting mechanism based on Gompertz function for two phase flow imaging using electrical capacitance tomography measurement systems. International Journal of Computer Applications, 101(9), pp.7-12, doi: 10.5120/17705-8704.

[28] Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P., 2004. Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing, 13(4) pp.600-612, doi: 10.1109/TIP.2003.819861.

[29] Liu, D., Kolehmainen, V., Siltanen, S., Laukkanen, A. M., and Seppänen, A., 2016. Nonlinear difference imaging approach to three-dimensional electrical impedance tomography in the presence of geometric modeling errors. IEEE Transactions on Biomedical Engineering, 63(9), pp.1956-1965, doi: 10.1109/TBME.2015.2509508.

[30] Google, 2021. Google Colaboratory. [Online]. Available: https://research.google.com/colaboratory.