To Be Decided

André Costa Batista, Ricardo Adriano, and Lucas S. Batista

***Abstract*—To be written.**

***Index Terms*—Keyword 1, keyword 2, keyword 3. Remember: alphabetical order.**

I. Introduction

HERE we go.

II. Problem Statement

Let $D \in \mathbb{R}^2$ denote the Domain of Interest (DOI) embedded within a homogeneous, isotropic, nonmagnetic ($\mu = \mu_0 = 4\pi \times 10^{-7}$ H/m), and lossless ($\sigma = \sigma_0 = 0$ S/m) background medium with permittivity $\epsilon_b = \epsilon_{rb}\epsilon_0$, where $\epsilon_0 \approx 8.85 \times 10^{-12}$ F/m. We consider a 2D Transverse Magnetic (TMz) polarization where the DOI is illuminated by incident plane waves and assume the time convention $e^{j\omega t}$. All scatterers are located within the DOI, and the scattered field E_z^s at point $\boldsymbol{\rho} \in S$ outside D is evaluated according to the following integral equation [1]:

$$E_z^s(\boldsymbol{\rho}) = -\frac{k_b^2}{4}\int_D H_0^{(2)}(k_b|\boldsymbol{\rho}-\boldsymbol{\rho'}|)\chi(\boldsymbol{\rho'})E_z(\boldsymbol{\rho'})dS' \quad (1)$$

where $k_b = \omega\sqrt{\epsilon_b\mu_0} = 2\pi/\lambda_b$ is the background wave number, λ_b is the wavelength of the incident wave, $H_0^{(2)}$ is the zero-order Hankel function of the second kind, E_z is the total electric field in the DOI, and χ is the contrast function given by

$$\chi(\boldsymbol{\rho}) = \frac{\epsilon_r(\boldsymbol{\rho})}{\epsilon_{rb}} - 1 - j\frac{\sigma(\boldsymbol{\rho})}{\omega\epsilon_b}, \quad (2)$$

which maps the relative permittivity $\epsilon_r(\boldsymbol{\rho})$ and conductivity $\sigma(\boldsymbol{\rho})$ distributions within D. It should be noted that the term $(-jk_b^2/4)H_0^{(2)}(k_b|\boldsymbol{\rho}-\boldsymbol{\rho'}|)$ corresponds to the Green's function in free space.

In electromagnetic inverse scattering problems, we aim to recover the contrast function χ using a set of N_M measurements of the scattered field collected for each of the N_S sources. Additionally, E_z is also unknown in D and must be solved. For our formulation, the N_S sources correspond to N_S incidence angles of the plane wave, and the N_M measurements are taken at N_M points in S, arranged in a circular, equidistant array situated far from the center of the DOI by a radius R_O. By discretizing the DOI into $N_X \times N_Y = N$ pixels, the inverse problem is solved numerically according to the following equation:

$$E_{ms}^s = -\sum_{n=1}^{N} G_{mn}^S \chi_n E_{ns} \quad (3)$$

where G^S is given according to the Richmond discretization [2], [3]. Equation (3) can also be expressed in matrix form:

$$\mathbf{E^S} = \mathbf{G^S}\boldsymbol{\chi}\mathbf{E} \quad (4)$$

where $\mathbf{E^S}$ is the $N_M \times N_S$ scattered field matrix, $\mathbf{G^S}$ is the $N_M \times N$ Green's function matrix, $\boldsymbol{\chi}$ is the $N \times N$ contrast diagonal matrix, and $\mathbf{E}$ is the $N \times N_S$ total electric field matrix.

It is important to highlight that, although assumptions are made regarding the scattered field domain and the incident field, the proposed indicators do not depend on these assumptions. They can be applied in any scenario where these entities are defined, and the assumptions in this paper serve only to clearly establish the context in which the indicators are tested.

III. Indicators

Once the result of the contrast function reconstruction is obtained by a given algorithm, the error in recovering the shape and location of the objects can be measured by comparing it with the original images. This means that the application of these metrics is relevant in studies where the exact response is known. Although in many real-world scenarios the imaged scatterer is unknown, applying these metrics to known cases can be valuable for comparing the performance of different algorithms and for estimating a confidence interval for the average performance of an algorithm in a given configuration. Therefore, these metrics are relevant tools that can contribute to the investigation and evaluation of algorithms for the problem.

For the two indicators that will be explained below, the image of the original contrast function of the problem and the image reconstructed by any algorithm will be denoted by the matrices $\bar{\boldsymbol{\chi}}_{\boldsymbol{o}}$ and $\bar{\boldsymbol{\chi}}_{\boldsymbol{r}}$, both of size $N_X \times N_Y$. In other words, the elements of these matrices represent the pixels of the images.

A. Shape Error

The contrast function has complex values when the medium or the scatterers have losses. Therefore, the first step is to account for the possibility that the elements of the matrices $\bar{\boldsymbol{\chi}}_{\boldsymbol{o}}$ and $\bar{\boldsymbol{\chi}}_{\boldsymbol{r}}$ have real and imaginary parts. The simplest approach is to use the magnitude of the complex variables as the value of each pixel. This choice does not eliminate the possibility that objects with different contrasts might have

This work has been supported by the Brazilian agencies Coordination for the Improvement of Higher Education Personnel - Brazil (CAPES) through the Academic Excellence Program (PROEX) under Grant 88887.815891/2023-00, FAPEMIG (Research Support Foundation of the State of Minas Gerais), and CNPq (The National Council for Scientific and Technological Development).

A. C. Batista, R. Adriano, and L. S. Batista are with the Department of Electrical Engineering, Universidade Federal de Minas Gerais, Belo Horizonte, MG 31270-901, Brazil (e-mail: andre-costa@ufmg.br; rluiz@ufmg.br; lusoba@ufmg.br). A. C. Batista and L. S. Batista are also with Operations Research and Complex Systems Laboratory (ORCS Lab), Belo Horizonte, MG, Brazil.

the same magnitude. However, in cases involving only lossless materials, this approach is not problematic because the contrast is purely real. Since the primary objective is to verify the shape of the scatterers, the critical aspect is distinguishing between the background medium and the scatterer, with the background medium having zero contrast. Thus, this approach to simplifying the methodology for measuring shape error should not have significant consequences in performance verification scenarios for the algorithm.

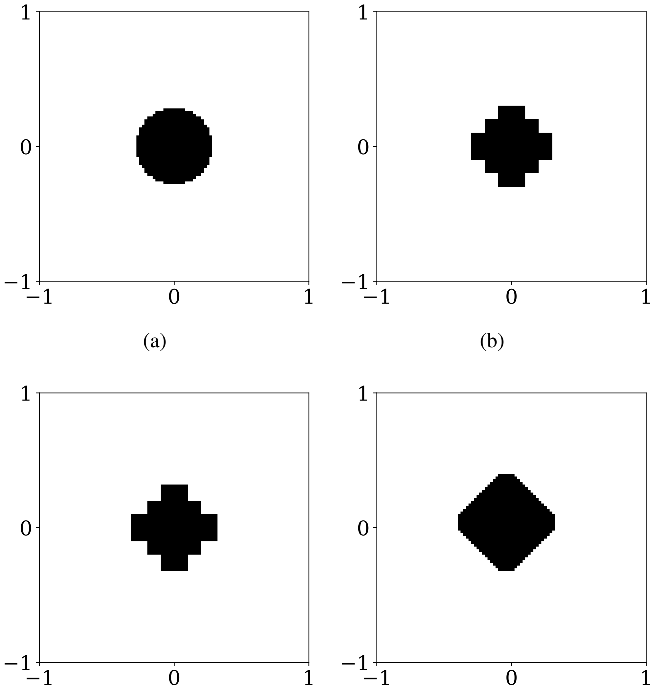
The contrast function reconstructed by many algorithms – such as BIM, DBIM, CSI, SOM, among others – is commonly a discretized approximation of a continuous surface. Therefore, even if the original image contains only objects with well-defined boundaries and homogeneous contrast, the image of these objects obtained by the algorithms will have frequently smooth edges, i.e., the contrast value gradually varies between the background medium and the object's value. Consequently, identifying a contour in the resulting image is heuristic because it requires selecting a criterion to define the contrast value at which a pixel can be considered as belonging to the object. For simplicity, the threshold T for considering a pixel in the recovered image to be within an object's contour will be defined as half of the total contrast variation in the image, i.e.:

$$T = \min(|\bar{\boldsymbol{\chi}}_{\boldsymbol{r}}|) + \frac{1}{2}\left[\max(|\bar{\boldsymbol{\chi}}_{\boldsymbol{r}}|) - \min(|\bar{\boldsymbol{\chi}}_{\boldsymbol{r}}|)\right] \qquad (5)$$

This approach has advantages and disadvantages. If the reconstructed image contains multiple scatterers, and one or more of them has a contrast below this threshold, the methodology will fail to detect the contour of these objects. However, in scenarios where there is only one scatterer in the image, this approach is simple and effective. In fact, a single scatterer is the most objective way to test an algorithm's potential in recovering geometry, since the presence of multiple scatterers can degrade the algorithm's performance due to the influence of the induced currents in each scatterer on the scattering of the other. Furthermore, this approach poses no issues for algorithms capable of accurately capturing contours, such as machine learning-based methods. Moreover, instead of using the values from the original image, it is more beneficial to use only the values from the reconstructed image, both to isolate the influence of error in the contrast estimate and to apply the metric to qualitative methods that do not estimate contrast. If, for some reason, the original contrast function involves soft boundaries of the objects, the threshold might also be applied to the original image.

Once the thresholding step is applied, the contours of the scatterers can be determined. One of the techniques suitable for this task is the Marching Squares algorithm [4]. This algorithm efficiently returns a set of points that delineate each contour. It is important to note that hollow objects may have two or more contours. Additionally, if the imaging algorithm fails to detect an object, a discrepancy will arise in the number of contours between the original and reconstructed images.

For these reasons, instead of comparing the geometries through calculations based on the points of each contour, we prefer to perform the comparison based on the number of pixels that were misclassified. Specifically, after identifying which pixels fall within each contour in each image, we can determine which pixels belonged to the scatterers in the original image but were not captured in the reconstructed image (false negatives) and which pixels did not belong to the scatterers in the original image but were classified as such in the reconstructed image (false positives).



(a)

(b)

(c)

(d)

Fig. 1. An example illustrating the differences between nearest neighbor interpolation and the adopted contour detection methodology. (a) Original image of the scatterer (resolution 100x100); (b) Image reconstructed by an imaging algorithm after thresholding (20x20); (c) Image reconstructed to the original resolution using nearest neighbor interpolation; (d) Image reconstructed to the original resolution using contour identification.

A potential question that may arise from this choice is why not directly calculate false positives and false negatives immediately after thresholding. One reason is that original and reconstructed images typically have different resolutions. While the nearest neighbor interpolation method is an alternative for addressing this issue, the chosen approach may be more effective for capturing and preserving object contours, particularly when the resolution of the reconstructed image is significantly lower. This is illustrated in Fig. 1.

The sum of false negatives and false positives can be easily obtained by applying the XOR operation between the two images. Then, we can compute the shape error as the ratio between the number of incorrectly classified pixels and the total number of pixels that are part of the scatterers in the original image. Additionally, if we consider the shape error in percentage values, we can then define the indicator ζ_S as

$$\zeta_S = \frac{\text{FP} + \text{FN}}{\text{TP}} \times 100\%, \qquad (6)$$

where FP and FN are the numbers of false-positive and false-negative pixels, respectively, and TP is the total number of pixels that are part of the scatterers in the original image. Therefore, the indicator assumes values greater than or equal to zero. Note that values greater than 100% are also possible, indicating that the reconstructed image has more incorrectly classified pixels than the total number of pixels that make up the scatterers in the original image. This can occur more frequently when the number of pixels of the scatterer in the original image is small. The Algorithm 1 summarizes the steps for calculating the shape error indicator.

Algorithm 1 Compute Shape Error Indicator ζ_S

```
Require: Original contrast distribution $\chi_o$
Require: Reconstructed contrast distribution $\chi_r$
Require: Threshold factor $threshold = 0.5$ (default)
Ensure: Shape error indicator $\zeta_S$ (%)
Extract magnitude information from both contrast distributions
Determine adaptive threshold based on reconstructed image dynamic range
Apply binary classification to identify object and background regions
if no object regions detected in reconstruction then
    return complete failure indicator (100%)
end if
Extract object boundaries using contour detection methodology
Normalize spatial coordinates to handle resolution differences between images
Generate binary masks from contour information using polygon filling:
    - Apply odd-parity rule to determine interior points
    - Handle multiple contours for complex geometries
Compare geometric regions through symmetric difference operation
Quantify classification errors relative to original object size
return normalized shape error as percentage
```

B. Position Error

The methodology for calculating the position detection error shares similarities with that of shape error calculation. As it is done when computing the shape error, the position error calculation also uses the absolute value of the image contrasts and applies the same thresholding operation. The reasons for this approach are the same.

The key difference lies in the method of position error calculation, which is based on comparing the centroid of pixels classified as objects by the thresholding operation in each image. Specifically, the centroid for each image is determined by averaging the coordinates of the pixels that constitute the objects. This method does not require contour detection, as the coordinate points are within the same boundaries for both images, even when resolutions differ.

The position error indicator is then defined as the Euclidean distance between the centroids in the two images. It is important to note that these centroids are normalized within a range of 0 to 1, where 0 represents the image's origin and 1 represents its end. By multiplying this distance by 100, the error can be interpreted as a percentage relative to the image size. Thus, the indicator quantifies the error in object localization relative to the image size, allowing its application across test sets with varying domain sizes. Therefore, the position error indicator ζ_P is defined as:

$$\zeta_P = \sqrt{(x_c^R - x_c^O)^2 + (y_c^R - y_c^O)^2} \times 100\%, \tag{7}$$

where (x_c^R, y_c^R) and (x_c^O, y_c^O) are the centroids of the reconstructed and original images, respectively.

It should be noted that inaccuracies in the reconstructed geometry of objects can significantly influence the position error calculation. For example, if a star-shaped object with five points is reconstructed with one point distorted, this distortion will affect the object's centroid, thereby impacting the position error. However, this effect is expected, as any criterion for determining an object's position inherently considers the pixels defining it, as it is for its geometry. Furthermore, the proposed indicator is straightforward and versatile, applicable even in scenarios involving images with multiple objects. The Algorithm 2 summarizes the steps for calculating the position error indicator.

Algorithm 2 Compute Position Error Indicator ζ_P

```
Require: Original contrast distribution $\chi_o$
Require: Reconstructed contrast distribution $\chi_r$
Ensure: Position error indicator $\zeta_P$ (%)
Extract magnitude information from both contrast distributions
Determine adaptive threshold based on reconstructed image dynamic range
Generate binary object masks using threshold-based classification
    - Original mask: pixels with non-zero contrast values
    - Reconstructed mask: pixels above computed threshold
if no object regions detected in reconstruction then
    return complete failure indicator (100%)
end if
Create normalized coordinate systems for both images
    - Map pixel coordinates to unit interval [0,1]
    - Handle potential resolution differences
Compute geometric centroids using weighted coordinate averaging:
    - Calculate center of mass for original object mask
    - Calculate center of mass for reconstructed object mask
Measure centroid displacement using Euclidean distance metric
Normalize distance to percentage scale relative to image dimensions
return position error as percentage of maximum possible displacement
```

IV. Computational Experiments

The experiments presented in this article are intended to illustrate the application of the proposed indicators. The design of these experiments focuses on demonstrating the potential utility of these indicators, rather than achieving pioneering results in the field. Nevertheless, the findings from these experiments can still provide a foundation for further research by the scientific community.

The experiments were conducted using the open-source library EISPY2D [5], specifically developed to design and evaluate algorithms for the inverse electromagnetic scattering problem. Both indicators will be analyzed, with each approach featuring two case studies (one fundamental and the other exploring a particular aspect) as well as a benchmarking study. The case studies aim to assess performance in specific scenarios, while the benchmarking study facilitates broader generalization of the results.

For all experiments, the following common parameters were used: background wavelength $\lambda_b = 1$ m, noise level of 5%, relative permittivity of the background medium $\epsilon_{rb} = 1$, and incident wave amplitude of 1 V/m.

Data synthesis was performed using the MoM-CG-FFT algorithm [6] with 5000 iterations. The measurement configuration consisted of 80 measurement points (N_M) and 80 incidence angles (N_S), with measurement points arranged in a circular array at distance $R_O = 4\lambda_b$ from the DOI center.

The DOI dimensions were set to $2\lambda_b \times 2\lambda_b$, and all reconstructions were performed using a 40×40 pixel discretization.

A. Shape Recovering Study

To evaluate shape error, two case studies were conducted. The first case study involves a single scatterer with homogeneous contrast, while the second case study involves a scatterer with variable contrast. These case studies were selected to demonstrate the application of the proposed indicator in typical scenarios encountered in electromagnetic scattering problems. Additionally, a benchmarking study was performed to assess the average performance of algorithms in more general situations.

1) Single scatterer: The first case study was designed to evaluate the applicability of the proposed indicator in a straightforward scenario where various algorithms could be applied. This case study features a scatterer shaped like a five-pointed star, centrally positioned within the image. This choice was made to select a geometry that, while symmetric and well-defined, has more vertices compared to more common shapes such as squares or triangles. The contrast was set to 0.25, and the radius from the center of the scatterer to the farthest vertices was set to $0.9\lambda_b$. Under these conditions, the Degree of Nonlinearity (DNL) for the test is 0.12, which is significantly lower than the threshold beyond which the Born Approximation becomes inapplicable, i.e., 1 [7]. The scattered field was synthesized using a discretization of 120×120 pixels for the original image.

For this experiment, the following algorithms were considered: LSM, OSM, BIM, CSI, and SOM. Their parameters were chosen after empirical tests that aimed to achieve their best performance. The configurations for each algorithm are as follows:

- Linear Sampling Method (LSM):
  - Threshold: 0.7;
  - Regularization Method: Conjugate Gradient with 300 iterations.
- Orthogonality Sampling Method (OSM):
  - Threshold: 0.35;
- Born Iterative Method (BIM):
  - Regularization Method: Conjugate Gradient with 300 iterations.
  - Stopping Criterion: 30 iterations.
- Contrast Source Inversion (CSI):
  - Stopping Criterion: 300 iterations.
- Subspace Optimization Method (SOM):
  - Stopping Criterion: 30 iterations.
  - Eigenvalue Cutoff Index: 5.

The images reconstructed by each algorithm are shown in Fig. 2. For the qualitative methods, OSM exhibited contrast variation within the object, which is consistent with the inherent characteristics of the method. Additionally, OSM recovered a larger area with more defined vertices compared to LSM. Regarding the quantitative methods, CSI demonstrated the lowest performance in recovering the object's contrast value close to the ground truth. Moreover, the quantitative methods produced more accurate contours than their qualitative counterparts.

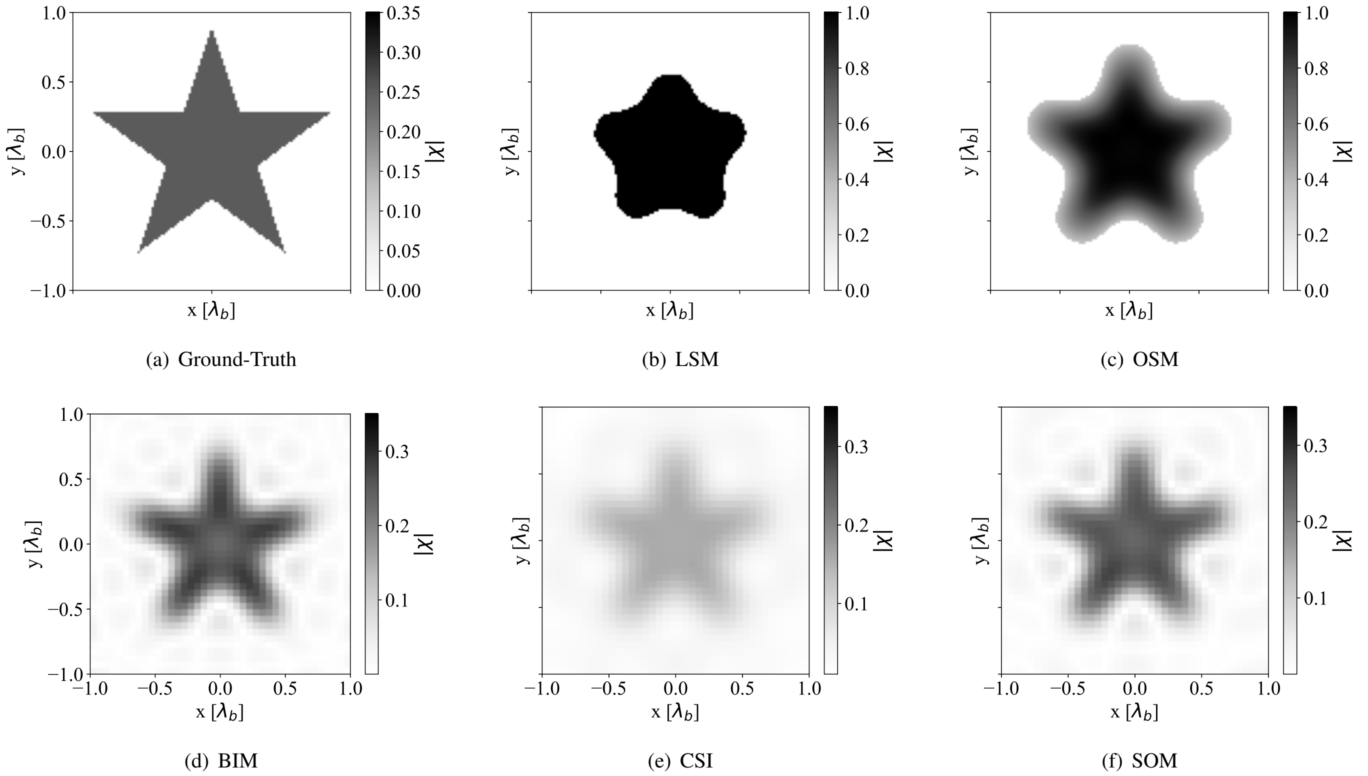
Table I summarizes the results of the shape error indicator for each algorithm. The results indicate that BIM and SOM achieved the best performance, while CSI, LSM and OSM performed similarly. This is consistent with the observations from Fig. 2, since the quantitative methods had a greater ability to capture the object's contours compared to the qualitative methods. This was not true for CSI, which, due to its greater difficulty in quantifying the object's contrast, had more difficulty in capturing the object's contours.

TABLE I
SHAPE RECOVERING STUDY, SINGLE SCATTERER: SHAPE ERROR INDICATOR ζ_S FOR EACH ALGORITHM.

Method	LSM	OSM	BIM	CSI	SOM
ζ_S (%)	29.2	24.5	14.9	26.3	15.4

2) Varying contrast: Another type of study that can complement the analysis of algorithm performance in object geometry reconstruction is the observation of indicator performance with varying object contrast. Specifically, for a given scatterer, as its contrast value increases, the DNL increases, and consequently, algorithms may lose performance since the problem becomes more challenging. To demonstrate this relationship, the same scatterer from the previous study was considered with identical scenario configurations. The only modification was that the original image resolution was increased to 160×160 pixels to ensure precision in field calculations for high contrast scenarios.

The scatterer was reconstructed using five contrast values: 0.25, 0.5, 0.75, 1.0, and 1.5. Fig. 4(b) shows the corresponding DNL for each contrast value. For this study, the OSM and SOM methods were employed with the same configurations as



(a) Ground-Truth

(b) LSM

(c) OSM

(d) BIM

(e) CSI

(f) SOM

Fig. 2. Shape recovering study, single scatterer: ground-truth and reconstructed images by each algorithm.

the previous study. While the method parameters could be optimized for each contrast value, they were kept constant across all contrast values to demonstrate the indicator's applicability in a controlled manner.

Fig. 3 shows the images reconstructed by OSM and SOM for each contrast value. As expected, reconstruction precision decreased with increasing contrast, reaching a point where the geometry was no longer correctly recovered for the highest contrast value (1.5), particularly in the case of SOM. Notably, SOM was able to estimate the scatterer's contrast with reasonable precision up to $\chi = 1$, where the DNL is approximately 2.

Fig. 4(a) presents the shape error indicator ζ_S for both OSM and SOM methods. The results demonstrate that the shape error increases proportionally with contrast value, which is consistent with the expected algorithmic behavior. Furthermore, for $\chi = 1.5$, the indicator ζ_S exceeds 100%, indicating that the number of incorrectly classified pixels surpasses the number of pixels composing the scatterer in the original image. This observation aligns with the finding that geometry recovery fails at this contrast level. Additionally, the difference between the OSM and SOM indicators remains relatively constant up to $\chi = 1$.

3) Average performance: When comparing multiple algorithms, it is advantageous to conduct case studies using test sets that represent specific problem classes. This approach enables the measurement and statistical comparison of average algorithm performance, thereby supporting more robust and reliable conclusions regarding their relative effectiveness.

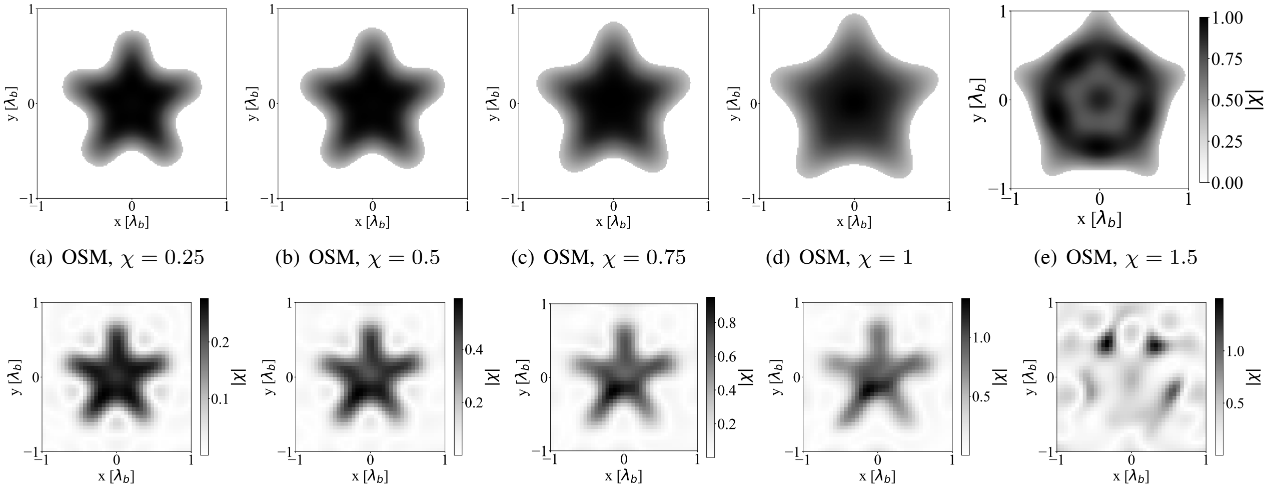
Following such an approach, an experiment was conducted to compare the LSM, OSM, BIM, CSI, and SOM algorithms. This experiment utilized 30 scatterer images with varying geometries to provide a more comprehensive assessment of algorithm performance across different shape configurations.

The scatterers were randomly generated as polygons with 4 to 15 vertices, where the radius from the center to the farthest vertices ranged between $0.45\lambda_b$ and $0.9\lambda_b$. The contrast was fixed at 0.25 for all scatterers, and the data synthesis parameters from Section IV-A1 were maintained for consistency.

Due to the geometric variations, the DNL for each instance varies accordingly, as illustrated in Fig. 5(a). The median DNL exceeds the value from the experiment in Section IV-A1, though both remain significantly below the threshold of 1, above which the Born Approximation becomes inapplicable [7]. This experimental design enables the assessment of statistical evidence supporting the superiority of any algorithm within this problem class.

Fig. 5(b) presents the shape error indicator ζ_S for each algorithm across all test instances. The results reveal a clear performance distinction when considering LSM, which demonstrates significantly worse performance compared to the other methods. Among the remaining algorithms, while overall performance levels appear similar, some patterns emerge: CSI exhibits higher error rates across a significant number of instances, whereas BIM demonstrates superior performance in a significant portion of the test cases.

While Fig. 5(b) provides initial insights into the results, statistical testing is important to validate the conclusions rigorously. To this end, Randomized Complete Block Design (RCBD) was applied to assess whether significant paired



(a) OSM, $\chi = 0.25$ (b) OSM, $\chi = 0.5$ (c) OSM, $\chi = 0.75$ (d) OSM, $\chi = 1$ (e) OSM, $\chi = 1.5$

(f) SOM, $\chi = 0.25$ (g) SOM, $\chi = 0.5$ (h) SOM, $\chi = 0.75$ (i) SOM, $\chi = 1$ (j) SOM, $\chi = 1.5$

Fig. 3. Shape recovering study, varying contrast: reconstructed images by OSM and SOM for different contrast values.

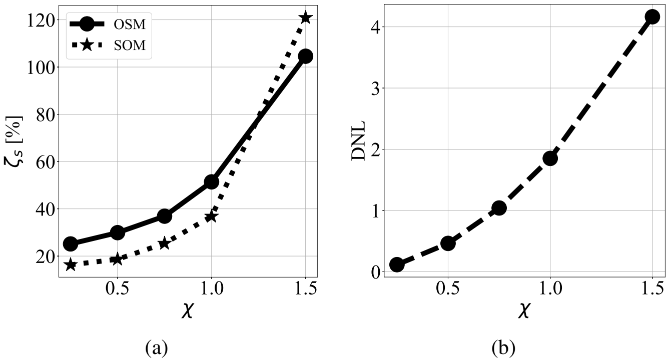


Fig. 4. Shape recovering study, varying contrast: (a) shape error indicator ζ_S for OSM and SOM; (b) Degree of Nonlinearity (DNL) for each contrast value.

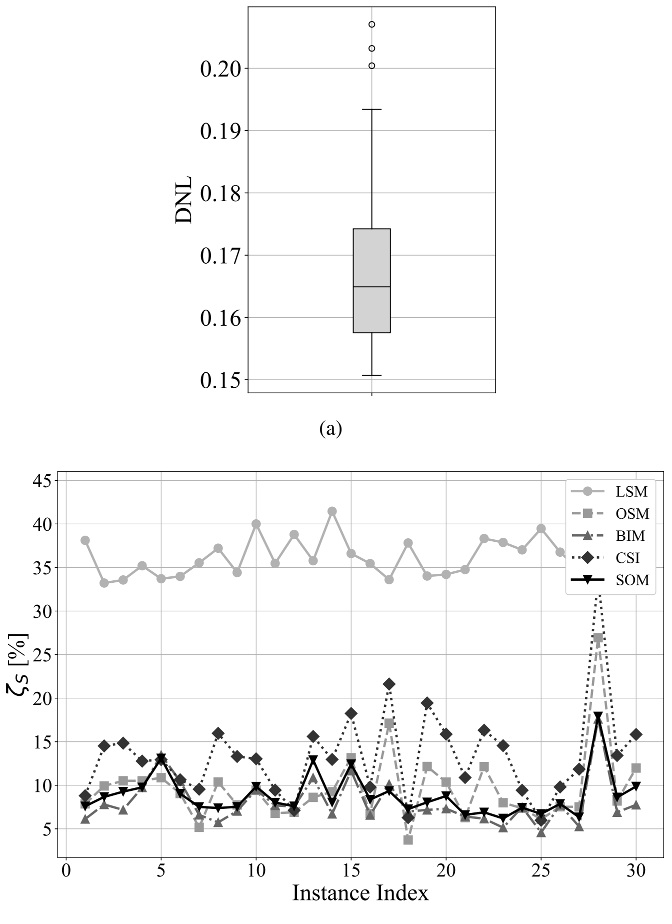


Fig. 5. Shape recovering study, average performance: (a) Degree of Nonlinearity (DNL) for each scatterer; (b) shape error indicator ζ_S for each algorithm in each instance.

differences exist among the algorithms. RCBD is a statistical test which is used to evaluate significant paired differences between three or more related groups when the data meet the assumptions for parametric tests. In this context, the groups correspond to the algorithms, and the measurements are the ζ_S indicator values for each algorithm across all instances. Given that LSM demonstrated substantially inferior performance compared to the other algorithms, it was excluded from the statistical analysis to focus on the more competitive methods.

After validating normality assumption, the p-value computed by RCBD result was less than 10^{-15}, indicating that at least one pair of algorithms shows a significant mean paired difference. Subsequently, multiple paired T-Test with Bonferroni correction were applied to identify which algorithms have significant mean paired differences between them. The p-values obtained from these tests are presented in Table II. As shown in the table, the null hypothesis that algorithms have equal performance is rejected for all algorithm pairs, except for OSM-SOM, indicating that OSM and SOM do not exhibit a significant difference in their average performance.

To complement the analysis, Fig. 6 presents the 95% confidence intervals for the mean paired differences of the shape error indicator ζ_S for each algorithm pair. As observed, the differences consistently favor BIM, providing evidence for BIM's superiority over the other algorithms. Additionally, CSI

TABLE II
SHAPE RECOVERING STUDY, AVERAGE PERFORMANCE: p-VALUES FOR THE PAIRED T-TEST WITH BONFERRONI CORRECTION.

Pairs	OSM-BIM	OSM-CSI	OSM-SOM	BIM-CSI	BIM-SOM	CSI-SOM
p-value	$< 10^{-2}$	$< 10^{-11}$	0.15	$< 10^{-7}$	$< 10^{-4}$	$< 10^{-6}$

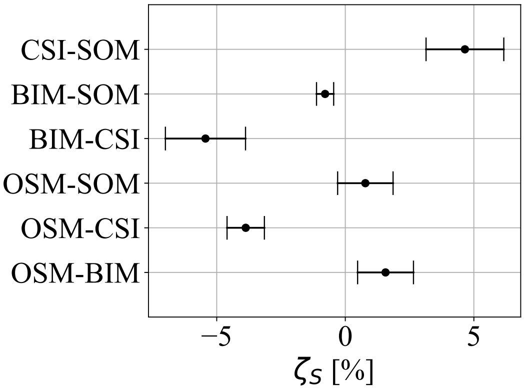


Fig. 6. Shape recovering study, average performance: Confidence intervals (95%) for the shape error indicator ζ_S for each pair of algorithms.

performs significantly worse than both BIM, OSM and SOM. However, all differences in average performance are less than 7%.

Furthermore, it is noteworthy that BIM also achieved the best performance in the study from Section IV-A1. However, while the difference between SOM and OSM was significant in the experiment from Section IV-A1, this benchmarking study provided evidence for performance equivalence between these two methods. In other words, if conclusions were drawn based solely on the case study, this could lead to a misconception about the performance of these two methods in this problem configuration. Therefore, this reinforces the importance of conducting benchmarking studies combined with statistical analyses to obtain more robust conclusions about algorithm performance.

B. Position Detection Study

In many scenarios where algorithms can adequately reconstruct the geometry of scatterers, the error in detecting scatterer positions tends to be low. For example, in the study described in Section IV-A1, the ζ_P indicator values obtained by LSM, OSM, BIM, CSI, and SOM were 0.049%, 0.057%, 0.25%, 0.31%, and 0.14%, respectively. This occurs because geometry and position are scatterer characteristics that algorithms typically do not handle separately.

An exception would be algorithms that solve the imaging problem as an optimization problem, where the decision variables are the position, vertices (or other characteristics of a predefined geometry), and contrast of the scatterers [8]–[10]. Additionally, in scenarios involving strong scatterers with small size compared to the wavelength, traditional methods generally struggle to recover scatterer geometry effectively due to the increased problem hardness. In such scenarios, studying position detection becomes more relevant.

Considering this, the experiments in this section were designed for scenarios involving strong scatterers with small size, where traditional methods cannot adequately recover scatterer geometry. Therefore, the position error indicator ζ_P serves as a relevant metric for comparing algorithm performance in such challenging scenarios. In addition, the DOI discretization for data synthesis was performed using a finer grid with 160×160 pixels.

Considering this, the experiments in this section were designed for scenarios involving strong scatterers with small dimensions, where traditional methods cannot adequately recover scatterer geometry. Therefore, the position error indicator ζ_P serves as a relevant metric for comparing algorithm performance in such challenging scenarios.

For all subsequent experiments, LSM was removed and another method was introduced. Since the analytical solution for scattering from an infinite dielectric cylinder is well-established [1], it becomes possible to determine the optimal circular approximation (defined by radius, position, and contrast) that best fits the scattered field of any single scatterer.

To optimize these four parameters, we employ a two-stage approach [11]:

1) Global optimization: The Differential Evolution (DE) algorithm [12] with 50 generations, 15 individuals, crossover rate of 0.7, and mutation factor ranging from 0.5 to 1.
2) Local refinement: The L-BFGS-B algorithm [13] for improved accuracy.

While DE is inherently stochastic and would typically require multiple executions for robust analysis, the problem structure and subsequent refinement step allow for single execution per instance based on previous experimental validation. This methodology is referred to as Circle Approximation (CA) throughout this work.

It should be noted that [8] employed a similar DE-based approach for circular fitting, though their work focused specifically on perfect electric conductor cylinders under cylindrical scatterer assumptions.

1) Single scatter: The first scenario for studying the position indicator utilized a star-shaped scatterer, similar to the experiments described in Sections IV-A1 and IV-A2. However, this configuration featured the scatterer positioned in the upper-right corner and rotated by 30 degrees, as illustrated in Fig. 7(a). The scatterer contrast was set to 7, while the radius from the center to the farthest vertices was reduced to $0.15\lambda_b$. Under these configurations, the DNL for this experiment reaches 1.7, which is slightly above the threshold of 1.

Fig. 7 shows the images reconstructed by each algorithm. As expected, given the high contrast and small scatterer size, the methods reconstruct small blobs in the scatterer region. The way these blobs are reconstructed is relevant, as this information directly impacts the measurement of scatterer position.

In this regard, CSI experienced the greatest reconstruction difficulty, with contrast concentrated in a very small region and significantly lower contrast levels compared to the other quantitative methods. It is also notable that CA most closely approximated the actual scatterer contrast.

Table III summarizes the results of the position error indicator for each algorithm. The results indicate that OSM, BIM, and CA achieved the best performance, while SOM performed similarly. The largest error was observed for CSI, which, due to its greater difficulty in quantifying the object's contrast, had more difficulty in capturing the object's position. Generally, the position detection errors for most algorithms were low. However, there may be applications where precision in position detection is critical, such as in tumor detection, for example.

TABLE III
POSITION DETECTION STUDY, SINGLE SCATTERER: POSITION ERROR INDICATOR ζ_P FOR EACH ALGORITHM.

Method	OSM	BIM	CSI	SOM	CA
ζ_P (%)	0.08	0.04	4.68	0.53	0.04

2) Multiple scatterers: Although the proposed indicator for measuring position error was designed for scenarios with a single scatterer, it can also be applied in scenarios with multiple scatterers. In these cases, the centroid of the arrangement as a whole serves as the reference point for measuring position error. This type of investigation can be useful for applications such as through-wall imaging, for example.

Based on this approach, this experiment considers three scatterers: the star from the previous experiment, a square, and a cross. The contrasts of the three scatterers were set to 4, 3.5, and 2.5, respectively. These values were chosen to be lower than in the previous experiment, since the presence of multiple scatterers inherently increases problem hardness due to mutual current induction between the scatterers.

The square has a side length of $0.15\lambda_b$, while the cross has dimensions of $0.3\lambda_b \times 0.3\lambda_b$ with a thickness of $0.15\lambda_b$. The scatterers were positioned such that their collective arrangement is located in the upper corner of the DOI, as illustrated in Fig. 8(a). Under this configuration, the DNL for this experiment is 0.3.

Fig. 8 presents the reconstructions obtained by each algorithm for the multiple scatterer scenario. As expected in this small-size configuration, the algorithms produce blob-like structures that are generally larger than the actual object geometries. The CA method positioned its circular approximation close to the square's location. Although the quantitative methods could not accurately estimate the individual scatterer contrasts, they consistently assigned higher contrast values to the square region rather than the star region. This behavior suggests that the square's larger cross-sectional area compared to the star may have influenced the algorithms' reconstruction process.

Table IV presents the position error results for each algorithm. Interestingly, CSI achieved the best performance in this multi-scatterer scenario, despite its poor performance in the single scatterer case. This suggests that the algorithm's behavior may be different when dealing with multiple targets. BIM and SOM showed similar intermediate performance, while OSM and CA exhibited higher position errors, indicating greater difficulty in accurately localizing the overall scatterer arrangement.

TABLE IV
POSITION DETECTION STUDY, MULTIPLE SCATTERERS: POSITION ERROR INDICATOR ζ_P FOR EACH ALGORITHM.

Method	OSM	BIM	CSI	SOM	CA
ζ_P (%)	10.24	8.30	4.86	8.27	12.85

3) Average performance: Similar to the shape recovery study, an average performance analysis was conducted to compare the position detection capabilities of OSM, BIM, CSI, SOM, and CA algorithms. This experiment employed 30 scatterer instances with configurations similar to those described in Section IV-B1. The scatterer contrast was fixed at 7, and the radius from the center to the farthest vertices was set to $0.15\lambda_b$. Following the approach from Section IV-A3, the scatterers were randomly generated as polygons with 4 to 15 vertices.

The DNL distribution is shown in Fig. 9(a). The values range from below 1 to above 4, with a median close to 2. The high contrast level and geometric variations contributed to the wide distribution of DNL values. However, as shown in Fig. 9(b), the algorithm performance within each instance remained within a similar range. CSI clearly exhibited the worst performance, followed by SOM. The best performance in each instance varied between OSM and CA. These results are consistent with the findings from Section IV-B1.

C. Breast Phantom Study

V. CONCLUSION

A conclusion section is not required. Although a conclusion may review the main points of the paper, do not replicate the abstract as the conclusion. A conclusion might elaborate on the importance of the work or suggest applications and extensions. [14]

ACKNOWLEDGMENT

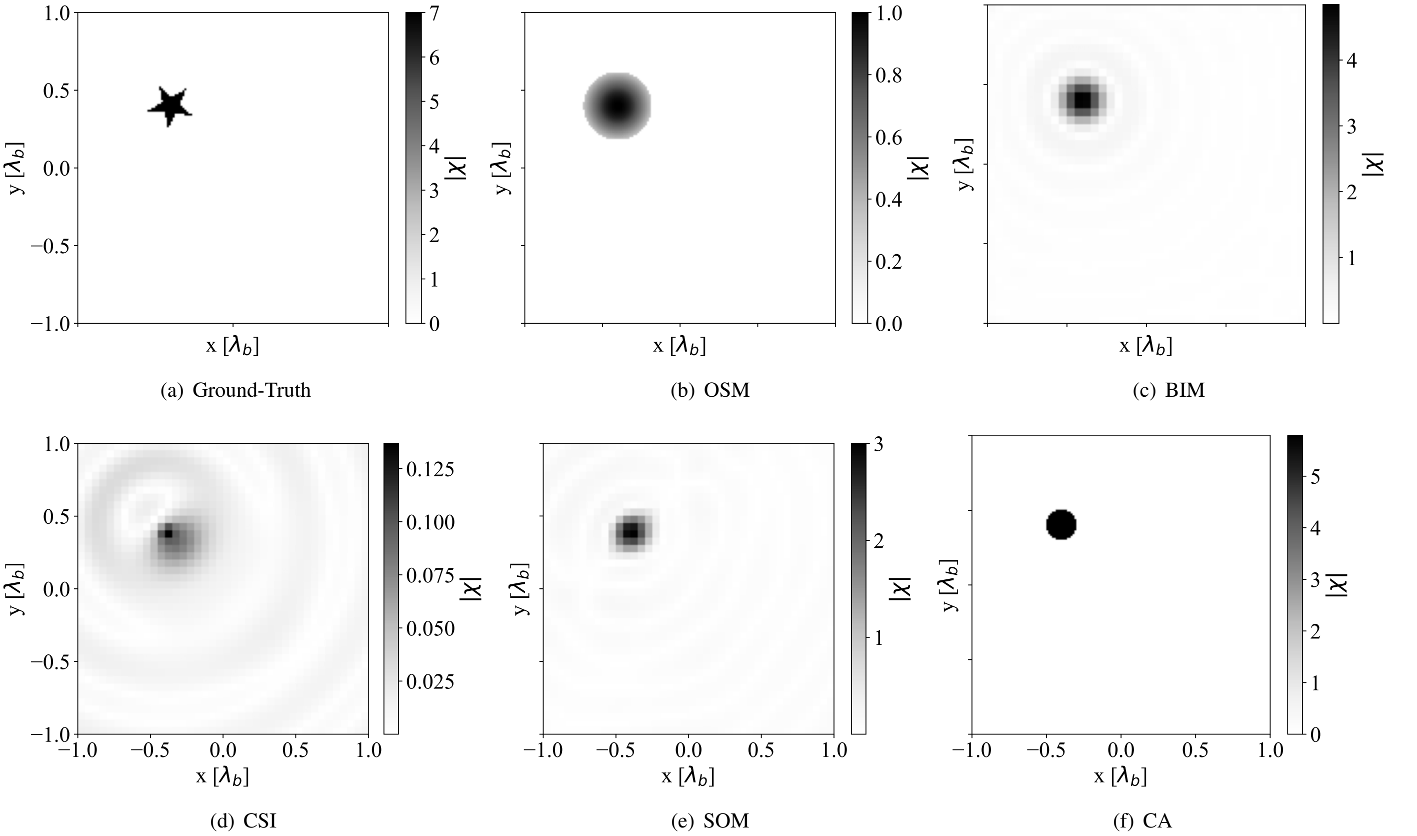
Use the singular heading even if you have many acknowledgments. In most cases, sponsor and financial support acknowledgments are placed in the unnumbered footnote on the first page, not here.

REFERENCES

[1] R. F. Harrington, *Time-Harmonic Electromagnetic Fields*. Wiley-IEEE Press, 2001.

[2] J. Richmond, "Scattering by a dielectric cylinder of arbitrary cross section shape," *IEEE Transactions on Antennas and Propagation*, vol. 13, no. 3, pp. 334–341, 1965.

[3] *The Electromagnetic Inverse Scattering Problem*. John Wiley & Sons, Ltd, 2010, ch. Three, pp. 20–56. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470602492.ch3



(a) Ground-Truth

(b) OSM

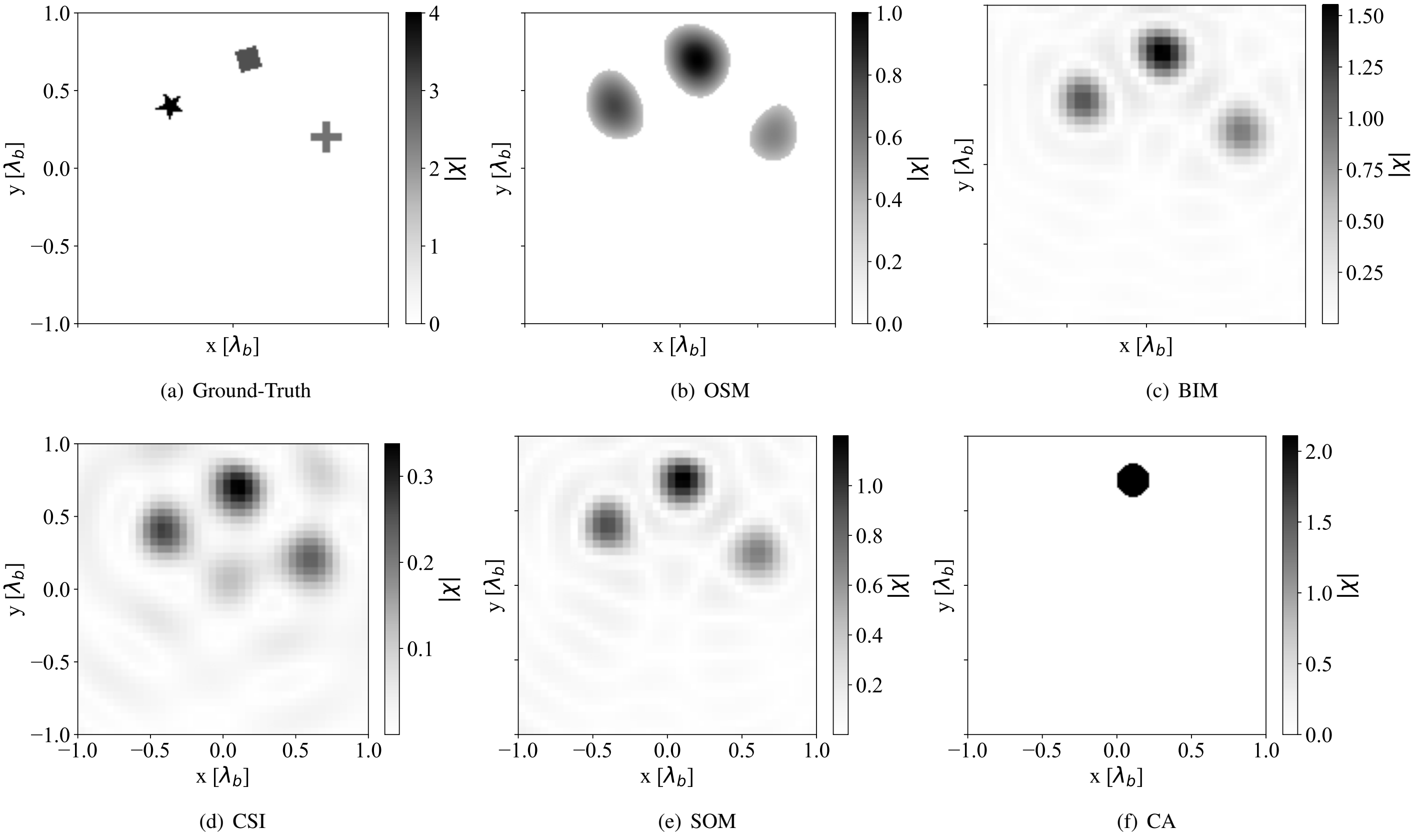
(c) BIM

(d) CSI

(e) SOM

(f) CA

Fig. 7. Position detection study, single scatterer: ground-truth and reconstructed images by each algorithm.



(a) Ground-Truth

(b) OSM

(c) BIM

(d) CSI

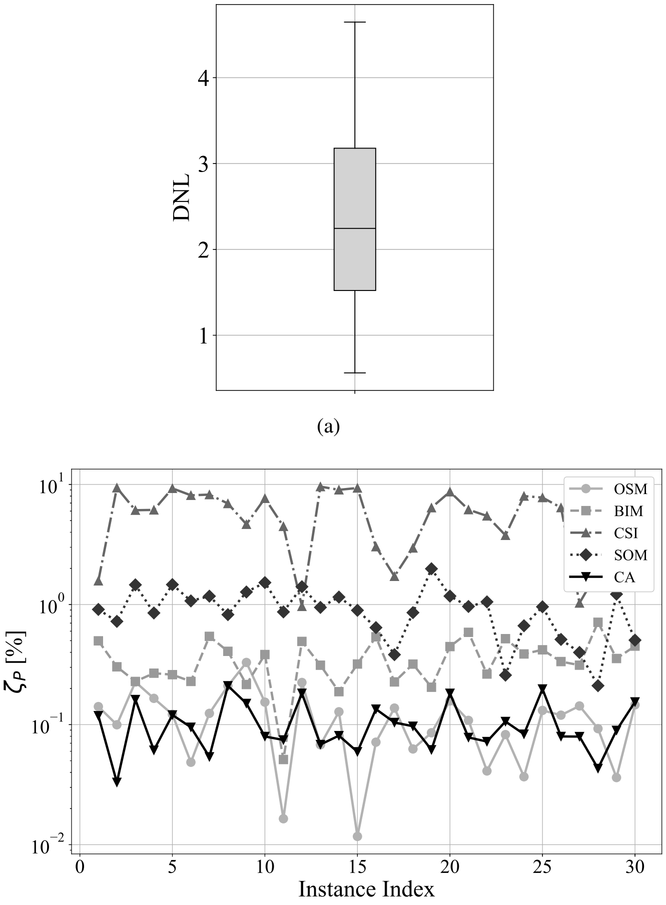
(e) SOM

(f) CA

Fig. 8. Position detection study, multiple scatterer: ground-truth and reconstructed images by each algorithm.

TABLE V
POSITION DETECTION STUDY, AVERAGE PERFORMANCE: p-VALUES FOR THE PAIRED T-TEST WITH BONFERRONI CORRECTION.

Pairs	OSM-BIM	OSM-SOM	OSM-CA	BIM-SOM	BIM-CA	SOM-CA
p-value	$< 10^{-9}$	$< 10^{-13}$	0.95	$< 10^{-6}$	$< 10^{-10}$	$< 10^{-17}$



(a)

(b)

Fig. 9. Position detection study, average performance: (a) Degree of Nonlinearity (DNL) for each scatterer; (b) position error indicator ζ_P for each algorithm in each instance.

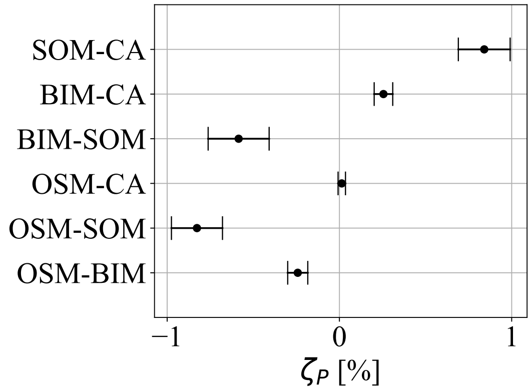


Fig. 10. Position detection study, average performance: Confidence intervals (95%) for the position error indicator ζ_P for each pair of algorithms.

[4] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," in *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '87. New York, NY, USA: Association for Computing Machinery, 1987, p. 163–169. [Online]. Available: https://doi.org/10.1145/37401.37422

[5] A. Costa Batista, R. Adriano, and L. S. Batista, "Eispy2d: An open-source python library for the development and comparison of algorithms in two-dimensional electromagnetic inverse scattering problems," *IEEE Access*, vol. 13, pp. 92 134–92 154, 2025.

[6] C.-C. Su, "Calculation of electromagnetic scattering from a dielectric cylinder using the conjugate gradient method and fft," *IEEE transactions on antennas and propagation*, vol. 35, no. 12, pp. 1418–1425, 1987.

[7] O. M. Bucci, N. Cardace, L. Crocco, and T. Isernia, "Degree of nonlinearity and a new solution procedure in scalar two-dimensional inverse scattering problems," *JOSA A*, vol. 18, no. 8, pp. 1832–1843, 2001.

[8] K. A. Michalski, "Electromagnetic imaging of circular–cylindrical conductors and tunnels using a differential evolution algorithm," *Microwave and Optical Technology Letters*, vol. 27, no. 5, pp. 330–334, 2000.

[9] M. Salucci, L. Poli, P. Rocca, and A. Massa, "Learned global optimization for inverse scattering problems: Matching global search with computational efficiency," *IEEE Transactions on Antennas and Propagation*, vol. 70, no. 8, pp. 6240–6255, 2022.

[10] F. Zardi, L. Tosi, M. Salucci, and A. Massa, "A physics-driven ai approach for microwave imaging of breast tumors," *IEEE Transactions on Antennas and Propagation*, vol. 73, no. 7, pp. 4661–4676, 2025.

[11] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020. [Online]. Available: https://doi.org/10.1038/s41592-019-0686-2

[12] R. Storn and K. Price, "Differential evolution –a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997. [Online]. Available: https://doi.org/10.1023/A:1008202821328

[13] C. Zhu, R. H. Byrd, P. Lu, and J. Nocedal, "Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization," *ACM Trans. Math. Softw.*, vol. 23, no. 4, p. 550–560, Dec. 1997. [Online]. Available: https://doi.org/10.1145/279232.279236

[14] X. Chen, *Computational methods for electromagnetic inverse scattering*. Wiley Online Library, 2018, vol. 244.