Deep learning correction for image reconstruction in electrical impedance tomography using UNet model

Abstract. This article was inspired by a similar Deep DBar algorithm, where a modified UNet convolutional model was used to correct the output of the DBar algorithm using the UNet model. However, instead of the DBar algorithm, another deterministic electrical impedance tomography reconstruction algorithm was used in this solution. The modified UNet model was used to successfully correct the initial reconstructions, which were computed using Kotre regularities using pseudo-inversion of the sensitivity matrix.

Streszczenie. Ten artykuł został inspirowany podobnym algorytmem Deep DBar, w którym zmodyfikowany model splotowy UNet został użyty do skorygowania danych wyjściowych algorytmu DBar przy użyciu modelu UNet. Jednak zamiast algorytmu DBar w tym rozwiązaniu zastosowano inny deterministyczny algorytm rekonstrukcji elektrycznej tomografii impedancyjnej. Zmodyfikowany model UNet został wykorzystany do skutecznej korekcji wstępnych rekonstrukcji, które zostały obliczone przy użyciu regularności Kotrego z wykorzystaniem pseudo-inwersji macierzy czułości (Poprawa rekonstrukcji tomografii impedancyjnej oparta o głębokie uczenie przy użyciu modelu UNet).

Keywords: Electrical Impedance Tomography, convolutional neural networks, UNet. Slowa kluczowe: elektryczna tomografia impedancyjna, sieci konwolucyjne, UNet.

Introduction

Electrical Impedance Tomography (EIT) reconstruction is a difficult (undetermined) problem to resolve. It relies on the reconstruction of the image on the scene based on a vector obtained from numerous measurements using EIT sensors. There are a lot of EIT reconstruction algorithms available in the bibliography. On the other hand, the most excellent results are obtained using neural networks (profound learning solutions).

Various numerical methods are available for such tasks [1-25]. The research [2] uses Artificial Neural Networks to reconstruct the image. This article describes using separate neural networks for each output pixel to reconstruct individual pixels. It produces better results than a single NN with multiple outputs, but the amount of neural networks and parameters in each network is huge. The article [17] showed the example application of convolutional neural network in EIT reconstruction.

Nowadays, the multiple ANN EIT reconstruction methods are based on deep and convolutional autoencoders. Paper [16] describes a solution based on EIT reconstruction gained using a deterministic algorithm (DBar) and applies UNet convolutional model to correct these initial reconstructions. The images are the input and output of the UNet model, so we can use them to correct other reconstruction algorithms’ results.

Another method using deep autoencoders described in [20] reconstructs lungs object based on Electrical Impedance Tomography. The method includes three steps:
1) A deep convolutional autoencoder is trained on reference (output) images (the lungs image reconstruction problem is presented in the paper).
2) The images encoded by the encoder part from the convolutional autoencoder trained in the previous step are applied as outputs to train of network with fully connected layers to predict such vectors based on electric potential vectors obtained from EIT sensors.
3) In the last stage, the joint model of both pre-trained networks is prepared. Finally, the output from the pre-trained model in stage 2 is inserted into the encoder part from the autoencoder obtained in the initial stage. This resultant hybrid network can reconstruct EIT images based on electrical potentials from Electrical Impedance Tomography measurements.

The research presented in this article was inspired by the algorithm called Deep DBar [16], which relies on the improvement of deterministic DBar algorithm output [24] using deep learning using the UNet model [26] with modifications. After examination of the DBar algorithm, it turned out that it has a relatively very low speed, so another faster deterministic algorithm for some practical applications should be developed. After examination of various models, there are also available different DNN models useful for EIT reconstruction using autoencoders [22], [23]. This model contains two separately trained parts (SAE and LR). The coder part from pre-trained earlier SAE autoencoder encodes potential vector and logistic regression layers (LR) reconstructing EIT images. The training process of the SAE autoencoder includes a few repeatable stages. All stages are used to train the SAE model to encode EIT potential vectors; thus, the potential vectors are on the input and output in this model. Initially, the SAE autoencoder contains only three layers (input layer, hidden layer and output layer). The hidden layer contains the encoded vector from the first step (which will be used in the next step). After a few iterations of that process, a deep autoencoder with more hidden layers to encode potential vectors is formed. In the end, the final hybrid model is constructed, which contains the encoder part from the SAE autoencoder and LE layers.

Training data generation

The datasets used in this experiment for training are synthetic. The algorithm for data generation produced 150 thousand scenes with different inclusions such as: circle, square or two them - where each subset contains 50 thousand of one type of samples with noise. The conductivity of the circle is less, and the conductivity of the square is greater than the background conductivity to obtain similar conditions as in the laboratory where the actual data comes from. Each scene image from the dataset was used in EIT simulation to generate potential vectors with lengths equal to 192.

The synthetic data generation for EIT reconstruction was also a difficult (inverse to inverse) problem because the
generated data must be close to real EIT data achieved in the laboratory. In order to gain vectors of potentials based on generated scenes with different kinds of inclusions, the simulation using the finite element method using square shapes was performed. The simulation algorithm parameters for dataset generation used in the experiments were adapted to get synthetic data close to real data obtained in the laboratory. Next, the potential differential vectors are computed using received potential vectors:

\[ X_i = x_i - x0_i \]

where: \( x \) – is the potential vector gained based on the scene containing inclusions, \( x0 \) – is the potential vector gained for the empty scene (with background only), \( X \) – is the potential differential vector, \( i \) – indicates the position of elements in potential and differential potential vectors.

Obtained samples of reference images (from now on referred to as \( Y \)) as well as potential differential vectors (after this referred to as \( X \)) were divided into a training dataset (inclusive 120 thousand samples) and test dataset (inclusive 30 thousand samples).

### Preliminary EIT reconstructions using a deterministic algorithm

The preliminary EIT reconstructions using the deterministic algorithm presented below generated 80x80 images based on potential differential vectors from each sample in the training and test datasets. The preliminary reconstructions (based on inputs in the form of potential differential vectors) were done using Kotre’s regularisation [25] by sensitivity matrix pseudo-inversion:

\[ J^{-1} = (J^T J + \lambda R)^{-1} J^T, R = (I \cdot (J^T J)) \frac{1}{2} \]

where: \( J \) – is the matrix of sensitivity, \( \lambda \) – is the regularisation coefficient established using the gradient method, \( (\cdot) \) – is the operator of the multiplication element by element.

Reconstructed conductivity is determined by:

\[ \sigma = J^{-1} V \]

where \( V \) – is a final post-processed measurement (in the form of a differential vector).

### Preprocessing of data

The data in the experiments was normalised as follows. First, the potential differential vectors are used to compute preliminary EIT reconstructions using the deterministic algorithm described earlier. Then, the reference images were normalised to range \([-0.5, 0.5]\). After that, the background pixel values have values of 0.5, while inclusions with conductivity lesser than the background (circles) have values of 0.0 and objects with conductivity greater than the background (squares) have values of 1.0. The second stage of reference image processing is the removal of the outer background (caused by the EIT simulation program – surrounding electrodes designate the proper EIT area). Since convolutional neural networks consider the entire area of the image, the input and output images should have a consistent background.

The preliminary EIT reconstruction images (obtained through a deterministic algorithm) are normalised using the min-max method for each sample (image) separately. This normalisation method was chosen based on our experiments with training EIT reconstruction images autoencoder, and after the data analysis, we noticed that the differences between minimal and maximal values in different images in the entire dataset were too huge. Used min-max normalisation is computed in the preprocessed image in the following way:

\[ v_{\text{norm}} = \frac{v - v_{\text{min}}}{v_{\text{max}} - v_{\text{min}}} \]

where: \( v \) - is the given value from the preliminary EIT reconstruction image with preprocessing, \( v_{\text{min}} \) - is the minimum pixel value in the image, \( v_{\text{max}} \) - is the maximal pixel value in the image, \( v_{\text{norm}} \) - is the output pixel value (normalized).

### Deep learning correction of EIT reconstructions

This section describes research leading to EIT reconstruction correction using a modified UNet model [26]. Because of the problem of training the classical UNet model, which uses convolutions equal to 3x3, these sizes of convolution filters were changed to 5x5, like in the paper [16]. In addition, the soft sign activation function was set in the model’s final layer. The training was done using 120 thousand pairs of images (with preliminary EIT reconstructions designated using the deterministic algorithm as input and reference scene images as output). All kinds of images during training (input and output) have 80x80 sizes. We are using Adam optimiser in the training process with a learning rate equal to \(10^{-3}\), MSE loss function, and batches with sizes equal to 64 and 45 epochs. After training using the training data set, the following numerical results MAE loss and DICE metric [27] on training and test datasets were obtained:

<table>
<thead>
<tr>
<th>DATASET</th>
<th>MAE</th>
<th>DICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>training</td>
<td>0.0017251</td>
<td>97.12</td>
</tr>
<tr>
<td>test</td>
<td>0.0031536</td>
<td>92.52</td>
</tr>
</tbody>
</table>

The visual samples of results for three kinds of inclusions inside (circles, squares, circles and squares) are presented in Figure 1.

### Testing of modified UNet model on the real data

The synthetic data used for modified UNet model training was very similar to real data obtained in the laboratory. We have multiple data sets obtained using a custom Electrical Impedance tomograph with two types of inclusions dipped in water. A part of the set only contains a plastic tube (with a circular cross-section), while other sets contain a plastic tube and a metal cuboid block (with a
square cross-section). The plastic tube has conductivity smaller than water, while the metal cuboid block has conductivity bigger than water.

For each sample in each real dataset, the potential vectors were obtained. In addition, the potential vectors were also obtained for the case without any inclusion, and potential differential vectors were calculated and saved per each sample (as in the case of synthetic data). After obtaining potential differential vectors, the initial EIT reconstructions were performed using a deterministic algorithm (as in the synthetic data case).

Notice that we have only the differential potential vectors obtained for real data. Because we have no reference images, we cannot assess the results numerically. However, all real data sets represent the same scene, and we know the inclusions types and positions to assess the results visually.

$$I_{pp_{i,j}} = I_{k_{i,j}}$$

where: $I_{pp}$ - is the corrected reconstruction image after postprocessing, $I$ - is the reconstruction image - UNet output for one sample, $k$ - is the number of samples, $i, j$ - are the coordinates of the image.

After that, the two thresholds are performed to extract inclusions with conductivity smaller than the background and greater than the background separately. Figure 3 shows samples for each dataset after the first (average) and second (thresholding) steps of postprocessing using all samples in the given real data dataset.

**Conclusions**

This paper used the modified UNet model for successful initial EIT reconstruction correction. The initial EIT reconstructions were computed using Kotre’s regularisation using pseudo-inversion of the sensitivity matrix. The method used for initial EIT reconstruction works very fast. The modified model after the train was tested on synthetic and real data. After postprocessing outputs for real data, the final results seem to be good.
REFERENCES


