Multilayer Sparse LSM=Deep Neural Network

Zhaolun Liu¹ and Gerard Schuster¹

¹King Abdullah University of Science and Technology (KAUST)

SUMMARY

We recast the multilayered sparse inversion problem as a multilayered neural network problem. Unlike standard least squares migration (LSM) which finds the optimal reflectivity image, neural network least squares migration (NNLSM) finds both the optimal reflectivity image and the quasi-migration Green's functions. These quasi-migration Green's functions are also denoted as the convolutional filters in a convolutional neural network and are similar to migration Green's functions. The advantage of NNLSM over standard LSM is that its computational cost is significantly less and it can be used for denoising migration images. Its disadvantage is that the NNLSM reflectivity image is only an approximation to the actual reflectivity distribution.

INTRODUCTION

The biological processing of information in a cat's brain can be mathematically approximated as a weighted summation of input values into a vertical layer of neurons, followed by a thresholding operation (Hubel and Wiesel, 1962). Thresholding simplifies the amount of information to be processed by eliminating unimportant input values that fall below a certain threshold. This pair of operations, weighted summation and thresholding of the input values into the first layer of neurons, leads to new inputs inserted into a second layer of neurons. These new inputs are reweighted, summed, and thresholded to inject into the second column of neurons. This process is repeated again and again to form a multi-layered neural network. Such networks are now used in many areas to automatically classify and make decisions about large data sets (LeCun et al., 2015).

Trial-and-error experimentation with neural networks over several decades generated the architecture known as deep convolutional neural networks (CNNs). The success of the current CNN architectures is evidenced by their practical applications with self-driving cars, image classification and retrieval from digital archives, and medical diagnosis from a combination of body images generated by MRIs, CAT scans, and PET scans.

Until recently, the design of effective CNN architectures was largely based on heuristic experimentation. This shortcoming largely results from the absence of a rigorous mathematical foundation for neural networks in general, and CNN in particular. Elad and his coauthors proposed that CNN problem could be recast as finding the sparsest model $\mathbf{m}$ under the L_1 norm subject to honoring the data misfit constraint $\|\Gamma\mathbf{m} - \mathbf{m}^{mig}\|_2^2 \leq$

β (Papayan et al., 2017; Elad, 2018):

$$\begin{aligned} \text{Given : } & \quad \Gamma, \mathbf{m}^{mig} \quad \text{and} \quad \Gamma\mathbf{m} + \text{noise} = \mathbf{m}^{mig}, \\ \text{Find : } & \quad \mathbf{m}^* = \arg \min_{\mathbf{m}} \|\mathbf{m}\|_1 \\ & \quad \text{subject to } \|\Gamma\mathbf{m} - \mathbf{m}^{mig}\|_2^2 \leq \beta \end{aligned} \quad (1)$$

where $\mathbf{m}$ is an $N \times 1$ input real-valued vector and Γ is an $N \times N$ matrix of real-valued weights. The scalar β is the specified noise tolerance. The iterative solution to this problem is a series of forward-modeling operations of a neural network, where each layer consists of a concatenation of a weighted summation of input values to give the vector $\mathbf{z}$ followed by a two-sided soft thresholding operation denoted as $\sigma(\mathbf{z})$ (Papayan et al., 2017).

We now show that the sparse solution to the least squares migration problem reduces to the forward modeling operations of a multi-layered neural network. Instead of just finding the optimal $\mathbf{m}^*$, we optimize for both the reflectivity $\mathbf{m}$ and the quasi-migration Green's functions Γ . These quasi-migration Green's functions are also denoted as the convolutional filters in a convolutional neural network approximate the role of migration Green's function (Schuster and Hu, 2000). The final image is denoted as the NNLSM estimate of the reflectivity distribution that honors the L_1 sparsity condition. The next section shows the connection between the multilayer neural network and the solution to the multilayer LSM problem. This is followed by the numerical examples with a synthetic salt model and field data from the North Sea.

THEORY OF NEURAL NETWORK LEAST SQUARES MIGRATION

The theory of standard least squares migration is first presented to establish the benchmark solution under the L_2 norm. This is then followed by the derivation of the sparse least squares migration (SLSM) solution for a single-layer network. The final subsection derives the NNLSM solution for a multi-layer network.

Least Squares Migration

The least squares migration problem in the image domain is defined (Schuster and Hu, 2000) as finding the reflectivity coefficients m_i in the $N \times 1$ vector $\mathbf{m}$ that minimize the L_2 objective function ϵ :

$$\mathbf{m}^* = \arg \min_{\mathbf{m}} \left\{ \frac{1}{2} \|\Gamma\mathbf{m} - \mathbf{m}^{mig}\|_2^2 \right\}, \quad (2)$$

where $\Gamma = \mathbf{L}^T \mathbf{L}$ is the symmetric $N \times N$ Hessian matrix, $\mathbf{L}$ is the forward modeling operator, and $\mathbf{L}^T$ is the migration operator. $\mathbf{m}^{mig} = \mathbf{L}^T \mathbf{d}$ is the migration image computed by migrating the recorded data $\mathbf{d}$ with the migration operator $\mathbf{L}^T$. The

Multilayer SLSM

kernel associated with the Hessian matrix $\mathbf{L}^T \mathbf{L}$ is also known as the point scatterer response of the migration operator or the migration Green's function (Schuster and Hu, 2000).

The formal solution to equation 2 is

$$\mathbf{m}^* = \mathbf{\Gamma}^{-1} \mathbf{m}^{mig}, \quad (3)$$

where it is too expensive to directly compute the inverse Hessian $\mathbf{\Gamma}^{-1}$. Instead, a gradient method gives the iterative solution

$$\mathbf{m}^{(k+1)} = \mathbf{m}^{(k)} - \alpha \mathbf{\Gamma}^T (\mathbf{\Gamma} \mathbf{m}^{(k)} - \mathbf{m}^{mig}), \quad (4)$$

where α is the step length, $\mathbf{\Gamma}$ is symmetric, and $\mathbf{m}^{(k)}$ is the solution at the k^{th} iteration.

Sparse Least Squares Migration

The sparse least squares problem (SLSM) is defined as finding the reflectivity coefficients m_i in the $N \times 1$ vector $\mathbf{m}$ that minimize the objective function ε (Perez et al., 2013):

$$\varepsilon = \frac{1}{2} \|\mathbf{\Gamma} \mathbf{m} - \mathbf{m}^{mig}\|_2^2 + \lambda S(\mathbf{m}), \quad (5)$$

where $\mathbf{\Gamma} = \mathbf{L}^T \mathbf{L}$ represents the migration Green's function (Schuster and Hu, 2000), $\mathbf{m}^{mig} = \mathbf{L}^T \mathbf{d}$ is the migration image, $\lambda > 0$ is a positive scalar, and $S(\mathbf{m})$ is a sparseness function. For example, the sparseness function might be $S(\mathbf{m}) = \|\mathbf{m}\|_1$ or $S(\mathbf{m}) = \log(1 + \|\mathbf{m}\|_2^2)$.

Single-Layer Sparse LSM

The solution to equation 5 is

$$\mathbf{m}^* = \arg \min_{\mathbf{m}} \left[\frac{1}{2} \|\mathbf{\Gamma} \mathbf{m} - \mathbf{m}^{mig}\|_2^2 + \lambda S(\mathbf{m}) \right], \quad (6)$$

which can be approximated by an iterative gradient descent method:

$$\begin{aligned} m_i^{(k+1)} &= m_i^{(k)} - \alpha \left[\overbrace{\mathbf{\Gamma}^T (\mathbf{\Gamma} \mathbf{m} - \mathbf{m}^{mig})}^{\mathbf{r}=\text{residual}} + \lambda S(\mathbf{m})' \right]_i, \\ &= m_i^{(k)} - \alpha [\mathbf{\Gamma}^T \mathbf{r} + \lambda S(\mathbf{m})']_i. \end{aligned} \quad (7)$$

Here, $S(\mathbf{m})'_i$ is the derivative of the sparseness function with respect to the model parameter m_i and the step length is α . Vectors and matrices are, respectively, denoted by boldface lowercase and uppercase letters. When $S(\mathbf{m}) = \|\mathbf{m}\|_1$, the solution in equation 7 can be recast as

$$\begin{aligned} m_i^{(k+1)} &= \\ \text{soft} \left(\left[\mathbf{m}^{(k)} - \frac{1}{\alpha} \mathbf{\Gamma}^T (\mathbf{\Gamma} \mathbf{m}^{(k)} - \mathbf{m}^{mig}) \right]_i, \frac{\lambda}{\alpha} \right), \end{aligned} \quad (8)$$

where, *soft* is the 2-sided soft thresholding function (Elad, 2010). Equation 8 is similar to the forward modeling procedure of the one-layer neural network. That is, set $k = 0$, $\mathbf{m}^{(0)} = 0$, and let the input vector be the residual vector $\mathbf{r} = -(\mathbf{\Gamma} \mathbf{m}^{(0)} - \mathbf{m}^{mig}) = \mathbf{m}^{mig}$ so that the first-iterate solution can be compactly represented by

$$\mathbf{m}^{(1)} = \text{soft}(\mathbf{\Gamma}^T \mathbf{m}^{mig}, \lambda), \quad (9)$$

where $\alpha = 1$. Here, the input vector $\mathbf{r} = \mathbf{m}^{mig}$ is multiplied by the matrix $\mathbf{\Gamma}^T$ to give $\mathbf{z} = \mathbf{\Gamma}^T \mathbf{r}$, and $\mathbf{z}$ is then thresholded

and shrunk to give the output $\mathbf{m} = \text{soft}(\mathbf{z}, \lambda)$. If we impose a positivity constraint for $\mathbf{z}$ and a shrinkage constraint so λ is small, then the soft thresholding function becomes that of a one-sided threshold function, also known as the Rectified Linear Unit or ReLU function. To simplify the notation, the $\text{soft}(\mathbf{z}, \lambda)$ function or ReLU($\mathbf{z}$) function is replaced by $\sigma_\lambda(\mathbf{z})$ so that equation 9 is given by

$$\mathbf{m}^{(1)} = \sigma_\lambda(\mathbf{\Gamma}^T \mathbf{m}^{mig}). \quad (10)$$

For the ReLU function there is no shrinkage so $\lambda = 0$.

We propose neural network LSM that finds both $\mathbf{\Gamma}^*$ and $\mathbf{m}^*$ which minimizes equation 5 (Liu and Schuster, 2018). To find the solution we use the alternating steepest descent method that alternates between finding $\mathbf{\Gamma}^*$ after, say 15, iterations and then finding $\mathbf{m}^*$ after the same number of iterations. $\mathbf{\Gamma}^*$ is the approximate migration Green's functions, also known as basis functions and convolutional filters (Liu and Schuster, 2018). Each filter is used to compute the feature map which corresponds to sub-images of reflection coefficients in the context of LSM. As will be seen later, convolutional filters that appear to be coherent noise can be excluded for denoising the migration image. The advantage of this approach is that only inexpensive matrix-vector multiplications are used and no expensive solutions to the wave equation are needed for backward and forward propagation of the wavefield.

Multilayer Neural Network LSM

Similar to the derivation by Elad (2018) and Pappayan et al. (2017), the multilayer neural network LSM problem is defined as the following.

Find: $\mathbf{m}_i, \mathbf{\Gamma}_i$ such that

$$\begin{aligned} \mathbf{m}_1^* &= \arg \min_{\mathbf{m}_1, \mathbf{\Gamma}_1} \left[\frac{1}{2} \|\mathbf{\Gamma}_1 \mathbf{m}_1 - \mathbf{m}^{mig}\|_2^2 + \lambda S(\mathbf{m}_1) \right], \\ \mathbf{m}_2^* &= \arg \min_{\mathbf{m}_2, \mathbf{\Gamma}_2} \left[\frac{1}{2} \|\mathbf{\Gamma}_2 \mathbf{m}_2 - \mathbf{m}_1^*\|_2^2 + \lambda S(\mathbf{m}_2) \right], \\ &\vdots \\ \mathbf{m}_N^* &= \arg \min_{\mathbf{m}_N, \mathbf{\Gamma}_N} \left[\frac{1}{2} \|\mathbf{\Gamma}_N \mathbf{m}_N - \mathbf{m}_{N-1}^*\|_2^2 + \lambda S(\mathbf{m}_N) \right], \end{aligned} \quad (11)$$

where $\mathbf{\Gamma}_i$ is the i^{th} Hessian matrix in the i^{th} layer. The first iterate solution to the above system of equations can be cast in a form similar to equation 10, except we have

$$\mathbf{m}_N^* \approx \sigma_\lambda(\mathbf{\Gamma}_N^T \sigma_\lambda(\mathbf{\Gamma}_{N-1}^T (\dots \sigma_\lambda(\mathbf{\Gamma}_1^T \mathbf{m}^{mig}) \dots)), \quad (12)$$

which is a repeated concatenation of the two operations of a multilayered neural network: matrix-vector multiplication followed by a thresholding operation. In all cases we use a convolutional neural network where different filters are applied to the input from the previous layer to give feature maps associated with the next layer.

For a perfect prediction of the migration image, $\mathbf{m}^{mig}$ can also be approximated as $\mathbf{m}^{mig} = \mathbf{\Gamma}_1 \mathbf{\Gamma}_2 \dots \mathbf{\Gamma}_N \mathbf{m}_N$. We refer to $\mathbf{\Gamma}^{(i)}$ as the effective basis function at the i^{th} level, i.e., $\mathbf{\Gamma}^{(i)} = \mathbf{\Gamma}_1 \mathbf{\Gamma}_2 \dots \mathbf{\Gamma}_i$ so that

$$\mathbf{m}^{mig} = \mathbf{\Gamma}^{(i)} \mathbf{m}_i. \quad (13)$$

NUMERICAL RESULTS

We provide numerical examples of multilayer NNLSM defined in equation 11 by using a migration image associated with the 2D SEG/EAGE salt velocity model shown in Figure 1a. The grid size of the model is 101 grid points in the z-direction and 101 grid points in the x-direction. The grid interval is 40 m in the x direction and 20 m in the z direction. Figure 1b shows the reverse time migration (RTM) image. The multilayer NNLSM consists of 3 convolutional layers: the first one contains 15 basis functions, i.e. filters, of size 11×11 grid points, the second one consists 15 basis functions of dimensions $11 \times 11 \times 15$, and the last one contains contains 15 basis function of dimensions $11 \times 11 \times 15$. Equation 11 is solved for both $\mathbf{m}_i$ and $\mathbf{\Gamma}_i$ ($i \in 1, 2, 3$) by the two-step iterative procedure denoted as the alternating descent method. The computed effective basis functions for these layers are shown in Figures 1c-1e, where the yellow, red and green boxes indicate the size of the effective basis functions, also known as quasi-migration Green's functions. It indicates that the basis functions of the first layer $\mathbf{\Gamma}_1$ contains very simple small-dimensional edges, which are called "atoms" by Elad (2018). The non-zeros of the second basis functions $\mathbf{\Gamma}_2$ combine a few atoms from $\mathbf{\Gamma}_1$ to create slightly more complex edges, junctions and corners in the effective basis function $\mathbf{\Gamma}^{(2)}$. Lastly, $\mathbf{\Gamma}_3$ combines atoms from $\mathbf{\Gamma}^{(2)}$ in order to create more complex structure of the migration image. The corresponding stacked coefficient images, also known as feature maps, are shown in Figures 1f-1h, which give the reflectivity distributions. The reconstructed migration images are shown in Figures 1i-1k.

We apply the one-layer NNLSM method to field data collected in the North Sea (Schroot and Scüttenhelm, 2003). The time migration image is shown in Figure 2a. The time axis is gridded with 213 evenly-spaced points and there are 301 grid points along the x-axis. Twenty-one 13-by-5 (grid point) convolutional basis functions are estimated by the alternating NNLSM procedure (see Figure 2b). These filters approximate dip-filtered migration Green's functions, and the basis function is marked as the yellow boxes in Figure 2a and 2b. The stacked coefficients (reflectivity distribution) are displayed in Figure 2c. It is evident that the stacked coefficients can provide a high-resolution migration image. After reconstruction from the learned basis functions and coefficients, the migration image is shown in Figure 2d with less noise.

SUMMARY

Neural network least squares migration finds the optimal reflectivity distribution $m(\mathbf{x})$ and quasi-migration Green's functions that minimize a sum of migration misfit and sparsity functions. The advantages of NNLSM over standard LSM are that its computational cost is significantly less than that for LSM and it can be used for filtering both coherent and incoherent noise in migration images. Its disadvantage is that the NNLSM reflectivity image is only an approximation to the actual reflectivity distribution.

To our knowledge, this paper is the first example of connect-

ing the physics of seismic imaging with the mathematics of sparse data representation and deep learning. For example, 1) quasi-migration Green's functions=CNN filters=basis functions, 2) reflectivity maps=CNN feature maps=weights for the basis functions. This connection to LSM might be extended to using CNN to inexpensively approximate the operations of full waveform inversion.

Multilayer SLSM

Figure 1: (a) 2D SEG/EAGE salt model, (b) RTM image, (c)-(e) learned effective basis functions $\Gamma^{(1)}$, $\Gamma^{(2)}$ and $\Gamma^{(3)}$, (f)-(h) stacked reflectivity coefficients for m_1 , m_2 and m_3 , (i)-(k) reconstructed migration images $\Gamma^{(1)}m_1$, $\Gamma^{(2)}m_2$ and $\Gamma^{(3)}m_3$.

Figure 2: a) Migration image computed from the F3 offshore block data, b) learned basis functions, c) stacked coefficients and d) migration image after filtering.

REFERENCES

- Elad, M., 2010, Sparse and redundant representations: from theory to applications in signal and image processing: Springer Science & Business Media.
- , 2018, Sparse modeling in image processing and deep learning: <https://www.youtube.com/playlist?list=PL0H3pMD88m8W39EbuArGLa9yXk8q6QGip>. (Accessed: 2019-03-27).
- Hubel, D. H., and T. N. Wiesel, 1962, Receptive fields, binocular interaction and functional architecture in the cat's visual cortex: *The Journal of physiology*, **160**, 106–154.
- LeCun, Y., Y. Bengio, and G. Hinton, 2015, Deep learning: *nature*, **521**, 436.
- Liu, Z., and G. Schuster, 2018, Neural network least squares migration: Presented at the First EAGE/SBGf Workshop on Least-Squares Migration.
- Papayan, V., Y. Romano, and M. Elad, 2017, Convolutional neural networks analyzed via convolutional sparse coding: *The Journal of Machine Learning Research*, **18**, 2887–2938.
- Perez, D. O., D. R. Velis, and M. D. Sacchi, 2013, Estimating sparse-spike attributes from ava data using a fast iterative shrinkage-thresholding algorithm and least squares, *in* SEG Technical Program Expanded Abstracts 2013: Society of Exploration Geophysicists, 3062–3067.
- Schroot, B., and R. Scüttenhelm, 2003, Expressions of shallow gas in the Netherlands North Sea: *Netherlands Journal of Geosciences - Geologie en Mijnbouw*, **82**, 91–105.
- Schuster, G. T., and J. Hu, 2000, Green's function for migration: Continuous recording geometry: *Geophysics*, **65**, 167–175.