RGNZ/MGPB

Shear wave modelling from conventional well logs using integrated deep learning Integrated Convolutional Neural Network (I-CNN)

Rudarsko-geološko-naftni zbornik
(The Mining-Geology-Petroleum Engineering Bulletin)
DOI: 10.17794/rgn.2026.4.4

Original scientific paper

Rahmat Catur Wibowo[1]* ✉, Ida Bagus Suananda Yogi[2] ✉, Indra Arifianto[3] ✉, Muh Sarkowi[4] ✉

[1] *Geological Engineering, Universitas Lampung, Sumantri Brojonegoro No.1, Lampung, 35145, Indonesia.*
[2] *Centre for subsurface Imaging, Universiti Teknologi Petronas, Seri Iskandar 32610, Perak, Malaysia.*
[3] *Earth Resources Engineering Department, Faculty of Engineering, Kyushu University, Fukuoka, 819-0395, Japan.*
[4] *Geophysical Engineering, Universitas Lampung, Sumantri Brojonegoro No.1, Lampung, 35145, Indonesia.*

Abstract

Shear-wave velocity (*Vs*), together with compressional wave velocity, provides a crucial source of information for both geomechanical and geophysical studies. *Vs* data are often unavailable. Moreover, direct measurement of *Vs* remains relatively costly. Four machine learning algorithms were created to predict *Vs* from traditional well logs in order to get around these restrictions: Probability Neural Network (PNN), Multilayer Feed-Forward Neural Network (MLFFNN), Deep Feed-Forward Neural Network (DFFNN), one-dimensional Convolutional Neural Network (1D-CNN), and an Integrated Convolutional Neural Network (I-CNN). The dataset consists of two wells (19,121 data points were gathered) of authentic industrial wireline logs from two anonymized wells, provided with formal authorization from the data owner exclusively for academic research purposes, including model training, testing, and validation. There were three primary parts in the methodology: (1) pre-processing the data to get rid of noise and change it into the right format; (2) using domain knowledge to drive feature engineering and selection; and (3) training, testing, and optimizing the model. The results demonstrated that the I-CNN model in RCW-1 well achieved the best performance, with an R^2 value of 0.971. When applied to the blind well (RCW-2), the I-CNN model maintained strong generalization capability, achieving an average R^2 value of 0.956. These findings indicate that the I-CNN outperforms other methods in handling complex, nonlinear relationships in *Vs* prediction. Overall, this study contributes to the growing body of literature on machine learning applications in petrophysical analysis by introducing an integrated deep learning framework that surpasses traditional approaches.

Keywords:
shear wave; machine learning; neural network; integrated CNN

1. Introduction

Shear-wave velocity (V_s) plays a vital role in subsurface characterization, reservoir evaluation, and seismic interpretation because it controls rock elastic moduli, Poisson's ratio, and lithological differentiation (**Pickett, 1963; Castagna et al., 1985; Tonni and Simonini, 2013**). Accurate V_s estimation is fundamental for applications such as seismic inversion, geomechanical modelling, and dynamic stability assessment of reservoirs. However, obtaining complete V_s logs through dipole sonic measurements is often infeasible due to borehole instability, tool malfunction, or economic constraints (**Lin et al., 2023**). Consequently, researchers have long sought to infer Vs indirectly from other well logs. Classical empirical relationships such as Gardner's velocity–density relation (**Gardner et al., 1974**), Greenberg–Castagna's mudrock line (**Greenberg and Castagna, 1992**), and Faust's resistivity–sonic transform (**Faust, 1953**) provided valuable initial frameworks but are restricted by their dependence on local lithological conditions and inability to capture the nonlinear relationships among petrophysical parameters (**Mahmoudian et al., 2022; Olutoki et al., 2024**).

With the advancement of data-driven techniques, machine learning (ML) and deep learning (DL) approaches have revolutionized the estimation of missing logs and elastic parameters from routine well logs (**Rajabi et al., 2023; Ghorbani et al., 2025**). These methods have been employed to model the highly nonlinear and heterogeneous relationships between petrophysical features and V_s. Among classical ML methods, algorithms such as support vector machines, random forests, and gradient boosting have shown remarkable accuracy and stability when trained on large, curated datasets (**Yang et al., 2023**). Nevertheless, these techniques heavily rely on manually

* Corresponding author: Rahmat Catur Wibowo
e-mail address: rahmat.caturwibowo@eng.unila.ac.id
Received: 5 November 2025. Accepted: 17 January 2026.
Available online: 21 July 2026

engineered features and cannot learn hierarchical representations directly from the data, which limits their scalability to complex stratigraphic intervals (**Olutoki et al., 2024**).

In this context, neural network-based approaches have emerged as superior alternatives owing to their capability to approximate nonlinear functions with high fidelity (**Dolatshahi and Molladavoodi, 2024**). The Probabilistic Neural Network (PNN), a model derived from Bayesian decision theory and Parzen window density estimation, is among the earliest neural frameworks applied to geophysical data (**Specht, 1990**). PNN models estimate the probability density of each class or regression target and compute the conditional expectation, resulting in smooth and uncertainty-aware predictions (**Earp et al., 2020**). The probabilistic nature of PNNs makes them highly robust for noisy and limited datasets. Their kernel-based structure, however, leads to high computational costs and poor scalability to large datasets. Furthermore, the smoothing parameter (σ) critically affects model bias and variance, making parameter tuning essential (**Tang et al., 2011; Gu et al., 2018**). PNNs perform well in homogeneous formations, but tend to struggle in heterogeneous lithologies where spatial and sequential dependencies are dominant (**Al-Mudhafer, 2017**).

To overcome some of these limitations, the Multilayer Feedforward Neural Network (MLFFNN), also known as the classical backpropagation network, has been widely adopted for V_s prediction (**Silva et al., 2013**). MLFFNNs are capable of learning complex nonlinear relationship between input logs and V_s through multiple layers of neurons and nonlinear activation functions. The simplicity and flexibility of MLFFNN architecture make it an attractive choice for various geological applications. However, this model treats each depth sample independently, ignoring the sequential and spatial continuity inherent in well-log data. As a result, while it may achieve high accuracy during training, its performance can degrade when predicting V_s at depth intervals not included in the training dataset or in entirely different wells, particularly if appropriate regularization techniques are not applied (**Khan and Kirmani, 2024; Al-Mudhafar et al., 2022**). Overfitting remains a common challenge, particularly when the training dataset is small or unbalanced.

Advancements in computing and data availability have paved the way for the Deep Feedforward Neural Network (DFFNN), an extension of MLFFNN with greater depth and representation power (**Tolstaya et al., 2023; Zeng et al., 2022**). By stacking multiple hidden layers, DFFNNs are capable for hierarchical feature extraction, enabling the network to learn both low-level and abstract representations of lithological and petrophysical interactions (**Shi et al., 2025; Vázquez-Ayala et al., 2025**). DFFNNs outperform shallow architectures in capturing the complex nonlinearities inherent in well-log data, particularly when predicting V_s across different stratigraphic units. However, the enhanced representational capability of the model comes with trade-offs, including greater computational demands, a higher susceptibility to overfitting, and the requirement for more extensive hyperparameter tuning and optimization (**Khan and Kirmani, 2024**). Moreover, DFFNNs still lack explicit mechanisms to handle depth continuity, as they operate on static samples rather than sequential depth information.

A further evolution of neural networks for geophysical applications is the Convolutional Neural Networks (CNNs), which processes well logs as one-dimensional sequences along depth (**Li et al., 2025; Chen et al., 2022**). CNNs use convolutional filters to capture local spatial dependencies between log measurements, effectively learning depth-invariant features. This ability allows CNNs to recognize stratigraphic transitions and subtle textural variations that conventional neural networks overlook (**Rajabi et al., 2023**). CNNs are also computationally efficient due to parameter sharing and local connectivity. However, conventional CNN architectures are typically designed as single-stack models with fixed kernel sizes, limiting their receptive field and ability to represent multi-scale geological phenomena. As a result, while CNNs excel at detecting fine-scale patterns such as thin beds or local facies transitions, they often fail to capture broader depositional trends or compaction effects that occur over larger depth intervals (**Olutoki et al., 2024; Wang et al., 2022**).

Despite these advancements, several persistent research gaps remain in the literature on *Vs* prediction. First, existing DL models, particularly single-stack CNNs and DFFNNs, inadequately integrate multi-scale spatial features, making it difficult to simultaneously model local and global geological trends (**Olutoki et al., 2024**). Second, generalization across lithologies and basins remains limited; models trained on a single field often exhibit performance degradation when applied to other regions due to differences in depositional environments, compaction history, and mineralogy. Third, while probabilistic models like PNN inherently quantify uncertainty, deep CNN-based approaches typically lack uncertainty estimation, hindering their applicability for risk-sensitive reservoir characterization (**Earp et al., 2020; Alqahtani et al., 2018**). Lastly, many field datasets contain missing or incomplete well logs, whereas most neural network architectures assume the availability of all input features. Consequently, limited research has addressed the development of robust multi-input or modality-specific CNN architectures capable of handling incomplete data through feature-fusion or data-integration strategies (**Wang et al., 2022**).

To address these gaps, this study introduces an Integrated CNN (I-CNN) architecture designed to integrate multi-scale spatial feature extraction, cross-log fusion, and uncertainty awareness for Vs prediction. The proposed architecture consists of parallel convolutional

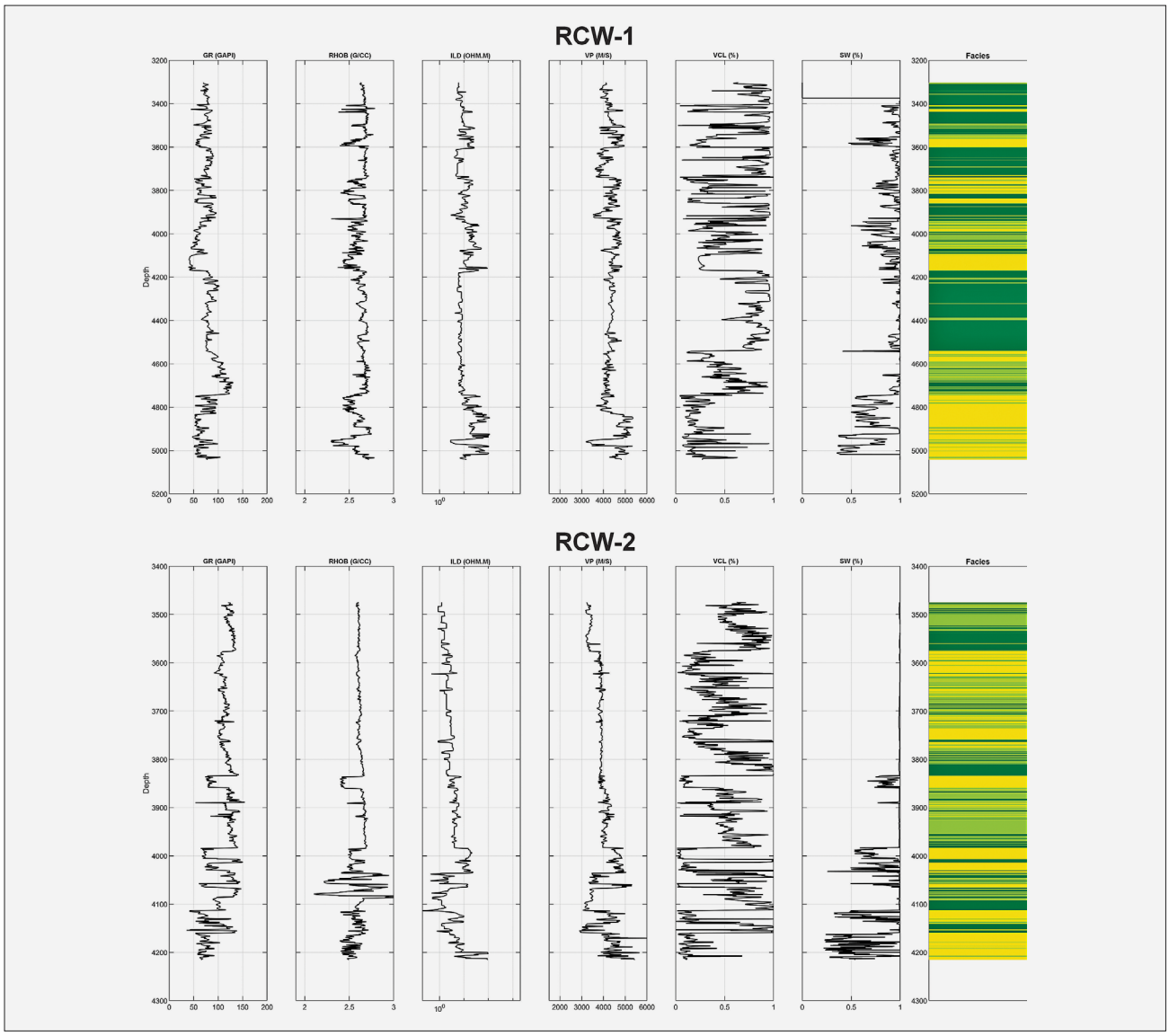


Figure 1: Composite well-log panels and facies interpretation for the RCW-1 (upper part) and RCW-2 (lower part) wells. Displayed from left to right are Gamma Ray (GR), Bulk Density (RHOB), Deep Resistivity (ILD), Compressional-wave velocity (Vp), Clay Volume (VCL), Water Saturation (SW), and the interpreted lithological facies (yellow indicates sandstone, light green indicates shaly sand, dark green indicates shale) plotted against depth.

branches with varying kernel sizes that capture fine to coarse depth patterns, followed by batch normalization, dropout layers for regularization, and feature-fusion dense layers for regression. By leveraging this structure, the model combines the local sensitivity of conventional CNNs with the broader context recognition of deep hierarchical networks. Through a comparative framework including PNN, MLFFNN, DFFNN, and single-stack CNNs, this study aims to demonstrate the superiority of the I-CNN in terms of accuracy, generalization, and robustness when predicting V_s from conventional well logs.

2. Geological condition in wells

The study is based on two anonymized wells, RCW-1 and RCW-2 (see **Figure 1**), which penetrate comparable stratigraphic intervals within a rapidly subsiding siliciclastic sedimentary system typical of large Cenozoic marginal-sea basins. The stratigraphic succession is dominated by thick alternations of sandstone, siltstone, and shale, reflecting deposition in deltaic to shallow-marine environments under high sedimentation rates. Such depositional conditions commonly result in laterally variable facies distributions and pronounced vertical lithological heterogeneity. Sandstone units are generally characterized by lower gamma-ray responses, reduced clay volume (VCL), and higher elastic velocities, whereas shale-rich intervals exhibit elevated gamma-ray values, higher clay content, and reduced *Vs*.

Progressive burial and rapid subsidence lead to strong compaction effects and significant contrasts in elastic properties across the stratigraphic section. In shale-dom-

inated intervals, incomplete compaction and fluid-related effects may further reduce *Vs*, while sandstone intervals typically develop higher stiffness due to stronger grain frameworks and diagenetic modification. These lithological and stratigraphic characteristics provide a physically meaningful framework for interpreting variations in *Vs* predicted by machine-learning models.

Although the wells originate from the same anonymized dataset, differences in shale proportion, resistivity response, and depth coverage introduce sufficient stratigraphic variability to enable a robust evaluation of cross-well model generalization within this geological framework.

3. Methods

Deep neural networks (DNN) are being used in all engineering specialties (**Blake et al., 2021; Spandonidis et al., 2022**). Because of its comprehensive data, DNN is widely used in petroleum exploration and production, according to a review by **Alkinani et al. (2019)**. The applications encompass the removal of wellbore damage (**Kellogg et al., 2018**), the integration of core and log data for permeability prediction (**Babadagli and Al-Salmi, 2004; Eriavbe and Okene, 2019**), forecasting hydrocarbon production in reservoirs subjected to water injection (**Negash and Yaw, 2020**), flow regime identification (**Van Der Spek and Thomas, 1999**), and the estimation of fluid rates across various phases and production simulation utilizing deep neural networks (**Alakeely and Horne, 2020a**). To identify the lithology from traditional well logs, a variety of models have been developed (**Saikia et al., 2020**). The following provides a detailed explanation and summary of the neural network-based V_s prediction approach using well logs.

3.1. Probability Neural Network (PNN)

As a non-linear method, the PNN approximation may successfully manage a variety of lithologies and differentiate between producing sands and shale intercalation (**Khan et al., 2022; Durrani et al., 2020**). A thorough description of PNN, a mathematical method for nonlinear interpolation that trains input petrophysical logs, was given by **Durrani et al. (2020)**.

The PNN is a theory of Bayes-Parzen classification, which is based on neural network principle (**Mohebali et al., 2020**). It employs Gaussian weighting functions in fitting the input vector (x) to the chosen training samples (x_i). The mean of a normal Gaussian distribution is a representation of the training sample in which the algorithm can finds the best control the width of the Gaussian parameter (σ_j) or standard deviation. The basic equation for the general PNN regression according to Todorov (2000) is given by the **Equation 1**:

$$y\,(x) = \frac{\sum_{i-1}^{n} y_i \exp(-D(x, x_i))}{\sum_{i-1}^{n} \exp(-D(x, x_i))} \tag{1}$$

where y_i represents the measured target log values, n represents total number of training samples, y is the new log value while $D(x, x_i)$ is expressed as **Equation 2**:

$$D(x, x_i) = \sum_{j=1}^{p} \left(\frac{x_i - x_{ij}}{\sigma_j} \right)^2 \tag{2}$$

The term $D(x, x_i)$ denotes the separation between each training point (x_i) and the input (x), and the total number of input is (p). The the width of the Gaussian parameter (σ_j) differs in every input. After identifying the ideal set of smoothing parameters by cross-validation, which excludes some of the training data (**Hampson et al., 2001**), the conjugate gradient method is applied to reduce the validation error (**Todorov, 2000**).

3.2. Multi-layered Feed Forward Neural Network (MLFFNN)

Artificial Neural Networks (ANNs) employ a complex computational framework to predict target responses from input data. They consist of numerous parallel connections between nonlinear, parameterized, and bounded functions known as neurons. These neurons are organized according to the network architecture, typically following a multilayer perceptron (MLP) design in which neurons are arranged in successive layers. Within each layer of an MLP, all neurons receive the same set of input signals but operate independently, without direct interconnections among them.

Although the number of hidden layers and neurons in each layer can be chosen arbitrarily, an excessive number of neurons may lead to overfitting, where the model learns noise rather than meaningful patterns. Conversely, using too few neurons can result in underfitting and poor network performance. One of the key advantages of Artificial Neural Networks (ANNs) compared to traditional methods is their ability to efficiently process large and complex datasets, capturing nonlinear relationships that conventional approaches may overlook.

The output function of an Artificial Neural Network (ANN) denoted as $h_{\theta(x)}$, is calculated as **Equation 3**:

$$h_{\theta(x)} = \mathrm{a}_1^3 = \mathrm{g}\left(\theta_{10}^{(2)} a_0^{(2)} + \theta_{11}^{(2)} a_1^{(2)} + \theta_{12}^{(2)} a_2^{(2)} + \theta_{13}^{(2)} a_3^{(2)}\right) \tag{3}$$

where (g) represents the activation function, in this case a sigmoid function defined by **Equation 4**:

$$\mathrm{g}(z) = \frac{1}{(1 + e^{-z})} \tag{4}$$

The activation function introduces nonlinearity into the model, allowing the network to capture complex relationships between input and output variables. For each neuron, the activation input z an be vectorized in matrix form, as follows:

$$\begin{Bmatrix} z_1^{(2)} = \theta_{10}^{(1)} I_0 + \theta_{11}^{(1)} I_1 + \theta_{12}^{(1)} I_2 + \theta_{13}^{(1)} I_3 \\ z_2^{(2)} = \theta_{20}^{(1)} I_0 + \theta_{21}^{(1)} I_1 + \theta_{22}^{(1)} I_2 + \theta_{23}^{(1)} I_3 \\ z_3^{(2)} = \theta_{30}^{(1)} I_0 + \theta_{31}^{(1)} I_1 + \theta_{32}^{(1)} I_2 + \theta_{33}^{(1)} I_3 \end{Bmatrix} \tag{5}$$

Here $I_0, I_1, I_2,$ and I_3 denote the input features (e.g. well-log parameters), while $\theta_{ij}^{(k)}$ are the weights connecting neuron j in layer $k-1$ to neuron i in layer k. The vectorized form allows efficient computation across all neurons simultaneously.

The overall complexity of the neural network is contingent upon its architecture, specifically the quantity of hidden layers and the number of neurons in each layer. Nonetheless, as highlighted by **Soroush et al. (2015)**, the network must be constructed with adequate complexity to discern data patterns while preventing overfitting.

An algorithm is employed to train the neuron network in order to reduce the error between the objective values and the network output values. This was accomplished through an iterative process that was aimed to determine the most optimal weights and biases. The Levenberg–Marquardt (LM) algorithm is the most well-known of the numerous algorithms that have been proposed in the literature for the purpose of training the network (**Wang et al., 2023**).

This study employed a MLFFNN architecture for the developed ANN model, trained using the gradient-based backpropagation algorithm to minimize prediction errors (**Brown et al., 2010**). The term backpropagation refers to the iterative process in which the network adjusts its internal weights to reduce the discrepancy between predicted and observed outputs. The error function of a specific input-output patterns is defined using the Root Mean Square Error (RMSE), as expressed in **Equation 6**:

$$RMSE(a,p) = \sqrt{\frac{1}{n}\Sigma_{i=0}^{n}(a_i - p_i)^2} \tag{6}$$

where a_i is the actual (measured) value for i^{th} sample, p_i is he predicted value produced by the network, and n is the total number of samples. The RMSE quantifies the model's prediction error, with lower values indicating higher accuracy and reliability. Model performance was further evaluated using multiple statistical metrics to ensure robustness across different magnitudes of data.

The backpropagation method, while popular for its simplicity and efficacy, possesses several shortcomings, including delayed convergence, incapacity to manage multiple targets, and a significant propensity to become ensnared in local minima during training. To overcome these issues, this study introduces an improved convergence approach by incorporating an acceleration factor, as formulated in **Equation 7**:

$$w(t+1) = w(t)\beta[\Delta w(t)] + \alpha[w(t-1)] \tag{7}$$

where w represents the connection weight, t is the iteration index (epoch), β is the learning constant, and α is the momentum (energy) constant. These constants, typically ranging between 0 and 1, are used to control the rate of learning and to smooth out abrupt gradient variations, thereby accelerating the convergence of the network while preventing oscillations during training.

3.3. Deep Feed Forward Neural Networks (DFFNN)

As a subset of machine learning, deep learning (DL) employs multiple layers of nonlinear transformations to construct hierarchical representations of data at increasing levels of abstraction. Through this multi-layered structure, deep learning models can automatically learn complex features and relationships directly from raw inputs, minimizing the need for prior domain knowledge, manual feature extraction, or heuristic coding rules (**LeCun et al., 2015**). In this study, a Deep Feed Forward Neural Network (DFFNN) was implemented and trained using the error backpropagation algorithm (**Rumelhart et al., 1986**) to optimize the weight matrices of each layer. The algorithm iteratively adjusts the model parameters by propagating the error, defined as the difference between the predicted and actual outputs from the output layer back through the hidden layers. During each iteration, the weight coefficients are updated to minimize this error, leading the network to converge toward an optimal set of parameters.

For each neuron in the (l^{th}) hidden layer, the forward propagation is expressed in **Equations 8** and **9**, as follows:

$$z^{(l)} = W^{(l)}a^{(l-1)} + b^{(l)} \tag{8}$$

$$a^{(l)} = f\left(z^{(l)}\right) \tag{9}$$

where:

$W^{(l)}$ = weight matrix at the l^{th} layer,

$b^{(l)}$ = bias vector at the l^{th} layer,

$a^{(l)}$ = activation output of the l^{th} layer,

$f(.)$ = nonlinear activation function (e.g. ReLU, sigmoid, or tanh).

In the output layer, the predicted response $\hat{y}$ is given by **Equation 10**:

$$\hat{y} = f_{out}\left(W^{(L)}a^{(L-1)} + b^{(L)}\right) \tag{10}$$

Where L denotes the total number of layers in the network. For hidden Layers, the Rectified Linear Unit (ReLU) activation function was used, defined as **Equation 11**:

$$f(x) = max(0,x) \tag{11}$$

ReLU introduces nonlinearity while mitigating vanishing gradient problems during training. For regression tasks, model performance was evaluated using the Root Mean Squared Error (RMSE) metric (**Equation 6**). The network weights were optimized via gradient-based learning, typically using Gradient Descent or its adaptive variants, such as Adam or RMSProp (**Raghavan et al., 2024**). The gradient of the loss function (∂L) with respect to the weight matrix (∂W) of the l^{th} layer is computed as **Equation 12**:

$$\frac{\partial L}{\partial W^{(l)}} = \delta^{(l)}\left(a^{(l=1)}\right)^T \tag{12}$$

The weights are then updated iteratively using the learning rate η, as **Equation 13**:

$$W^{(l)} \leftarrow W^{(l)} - \eta \frac{\partial L}{\partial W^{(l)}} \quad (13)$$

where η controls the step size during optimization. The iterative process continues until convergence is achieved, yielding an optimal mapping between input features and target outputs.

3.4. Convolutional Neural Networks (CNNs)

One particular type of deep neural network is the Convolutional Neural Network (CNN), which uses convolutional operations to automatically identify, extract, and categorize characteristics from input data. CNNs have achieved remarkable success across a wide range of domains, particularly in image recognition, computer vision, and pattern classification (**O'Shea and Nash, 2015; Li et al., 2021**). Over time, numerous advanced architectures have been developed, including GoogLeNet, ResNet (Residual Network), and SENet (Squeeze-and-Excitation Network), each introducing novel strategies to enhance representational efficiency and mitigate training limitations.

The term convolution refers to a linear mathematical operation involving a dot product between a kernel (or filter) a small matrix of learnable parameters, and localized regions (slices) of the input data. The kernel slides systematically over the input feature map to compute weighted sums, resulting in a transformed output known as a feature map. This operation enables the network to capture local spatial patterns such as edges, textures, or gradients within the data (**Li et al., 2021; Alakeely and Horne, 2020b**).

A typical CNN architecture consists of several layers, including input, convolutional, pooling, and fully connected layers. Among these, the convolutional and pooling layers play a critical role in feature extraction and dimensionality reduction. In the convolutional layer, local features are extracted by applying convolutional kernels to the input feature map, while the pooling layer aggregates information by summarizing neighboring features, thus improving translation invariance and computational efficiency (**Li et al., 2021**).

Equation 14 below is the modified calculation algorithm for convolutional network features (**Chen et al., 2020**).

$$F_j^i = \sum_{i \in mj} F_j^{t-1} * k_{ij}^l + b \quad (14)$$

The convolution process between the kernel k and input feature F is denoted by (*), and bias b is introduced following each convolution mapping.

While increasing the number of layers can enhance the model's expressiveness, doing so introduces more parameters and may lead to overfitting or training instability (**Guo et al., 2024**). Advanced CNN architectures address these challenges in different ways: (1) Inception Networks reduce dimensionality using parallel convolutions with kernels of size 1×1, 3×3, and 5×5, concatenating their outputs into a single composite feature map. (2) ResNet model introduce skip connections, allowing gradients to flow directly across layers and effectively mitigating vanishing or exploding gradient issues in very deep networks (**Alzubaidi et al., 2020**).

The convolution operation uses a small kernel/filter that slides along the input data (**Equation 15**). A commonly used activation function is ReLU, (**Equation 11**), to make the network capable of modelling nonlinear relationships efficiently.

$$h_{i,j}^{(k)} = f\left(\sum_{m,n} x_{i+m,j+n} \cdot w_{m,n}^{(k)} + b^{(k)}\right) \quad (15)$$

where:

x = input data,
w = kernel (filter) weight,
b = bias term,
$h_{i,j}^{(k)}$ = output feature map, and
$f(.)$ = activation function, typically the ReLU (see **Equation 11**).

The pooling operation reduces the spatial dimensionality of the feature maps while retaining the most relevant information. For instance, 2×2 max-pooling operation can be represented as **Equation 16**:

$$p_{i,j} = \max\{h_{2i+2j}, h_{2i,1,2j}, h_{2i,2j+1}, h_{2i+1,2j+1}\} \quad (16)$$

After convolution and pooling, the resulting feature maps are flattened and passed to one or more fully connected (dense) layers, similar in structure to those used in deep feed-forward networks (**Equation 17**). The CNN is then trained via the backpropagation algorithm and optimized using gradient-based methods such as Adam (**Equation 12**). The final output layer produces the target prediction (estimated V_s) using a linear function.

$$z^{(l)} = W^{(l)} a^{(l-1)} + b^{(l)}, \qquad a^{(l)} = f\left(z^{(l)}\right) \quad (17)$$

3.5. Integrated Convolutional Neural Networks (I-CNNs)

An Integrated Convolutional Neural Network (I-CNN) – also referred to as a deep CNN – is an extension of the standard CNN in which multiple convolution–pooling blocks are stacked sequentially. This multi-stack configuration enables the network to extract hierarchical features at different levels of abstraction. In contrast to shallow CNNs, which primarily capture low-level local features (e.g. short-range variations in GR or RHOB logs), I-CNNs progressively learn higher-order representations such as lithological boundaries, facies transitions, and nonlinear relationships between petrophysical logs and the target V_s (**LeCun et al., 2015; Goodfellow et al., 2016**).

Each convolution layer performs a localized linear transformation followed by a nonlinear activation, expressed as **Equation 18**:

$$z_{i,j}^{(k)} = \sum_m \sum_n X_{i+m,j+n} \cdot w_{m,n}^{(k)} + b^{(k)} \quad (18)$$

where:

X = local input patch,
$w^{(k)}$ = kernel (filter) for the k^{th} feature map,
$b^{(k)}$ = bias term, and
$z_{i,j}^{(k)}$ = output activation before applying the nonlinearity.

The pooling layer then aggregates the maximum or mean value within a small neighborhood to reduce dimensionality while preserving the most informative spatial features. In the fully connected stage of the network, the standard linear operation is defined as **Equation 19**:

$$h = f(W_x + b) \quad (19)$$

where W and b are the weights and bias of the dense layer, and $f(.)$ is the activation function ReLU (**Equation 11**). Mathematically, in an L-layer Integrated CNN (I-CNN), the output of the ℓ^{th} convolutional layer is represented as **Equation 20**:

$$z^{(\ell)} = f\left(\text{Conv}\left(a^{(\ell-1)}, W^{(\ell)}\right) + b^{(\ell)}\right) \quad (20)$$

where:

$a^{(\ell-1)}$ = input feature map from the previous layer,
$W^{(\ell)}$ = set of learnable kernels (filters),
$b^{(\ell)}$ = bias vector term, and
$f(\cdot)$ = non-linear activation function such as ReLU (**Equation 11**).

This operation allows the I-CNN to learn both spatially local and depth-wise dependencies among log attributes across multiple wells. Pooling layers are commonly inserted between convolutional layers to further reduce spatial dimensions and improve translation invariance. A typical Max-Pooling operation is expressed as **Equation 21**:

$$a^{(\ell)}(i,j) = \max_{(m,n)\in\Omega_{ij}} \left[z^{(\ell)}(m,n)\right] \quad (21)$$

where Ω_{ij} denotes the local receptive window around location (i,j). After several convolution-pooling blocks, the feature maps are flattened and passed to one or more fully connected (dense) layers to integrate the learned representations **Equation 22**:

$$\hat{y} = W^{(FC)} a^{(L)} + b^{(FC)} \quad (22)$$

where $\hat{y}$ is the predicted target, such as V_s. To train the network, a loss function such as the Root Mean Squared Error (RMSE) (**Equation 6**) and backpropagation with gradient descent (or Adam optimizer) is used to update all convolutional and fully connected weights (**Equation 13**).

3.6. Workflow overview

The proposed workflow for shear wave prediction from conventional well logs is illustrated in **Figure 2**. This study adopts a systematic, comparative modelling framework integrating five machine learning algorithms: Probabilistic Neural Network (PNN), Multilayer Feedforward Neural Network (MLFFNN), Deep Feedforward Neural Network (DFFNN), single-stack one-dimensional Convolutional Neural Network (CNN), and a custom Integrated Convolutional Neural Network (I-CNN) architecture. The workflow encompasses model training, validation, and cross-well testing under a rigorous train/test/validation protocol to ensure robustness and reproducibility.

Training and testing were performed using data from RCW-1, which contains a comprehensive suite of conventional logs (gamma-ray, bulk density, resistivity, compressional sonic, clay volume and water saturation) along with dipole sonic-derived V_s. This configuration aligns with best practices in well-level supervised learning and model reproducibility (**Rajabi et al., 2023; Zeng et al., 2022**). To evaluate model generalization, an independent well (RCW-2) from a distinct stratigraphic interval was reserved exclusively for out-of-sample validation, enabling assessment of cross-well transferability critical for field-scale deployment (**Wang et al., 2022; Olutoki et al., 2024**).

Data preprocessing involved standard cleaning, outlier removal, depth alignment, and feature scaling (min–max and/or z-score normalization) to ensure numerical stability across algorithms and mitigate bias caused by inconsistent log units (**Rajabi et al., 2023; Ghorbani et al., 2025**).

The PNN model was implemented as a kernel-based Parzen window estimator generating probabilistic predictions, with its smoothing parameter (σ) optimized via nested cross-validation on RCW-1, following established geoscientific probabilistic regression procedures (**Earp et al., 2020; Alqahtani et al., 2018**). The MLFFNN baseline employed a single hidden layer trained via backpropagation with L2 regularization, representing local nonlinear mappings between log features and V_s while maintaining computational efficiency for moderate datasets (**Silva et al., 2013**). The DFFNN extended this architecture into multiple stacked dense layers with ReLU activations, batch normalization, and dropout regularization to capture hierarchical and compositional relationships. Hyperparameters (depth, width, learning rate, weight decay) were optimized through grid/random search with early stopping to minimize overfitting (**Tolstaya et al., 2023; Khan and Kirmani, 2024**).

The 1-D CNN treated depth-ordered logs as continuous one-dimensional signals, applying successive convolution and pooling operations to extract depth-invariant local features that enhance the detection of thin beds and transition zones (**Li et al., 2025; Chen et al., 2022**).

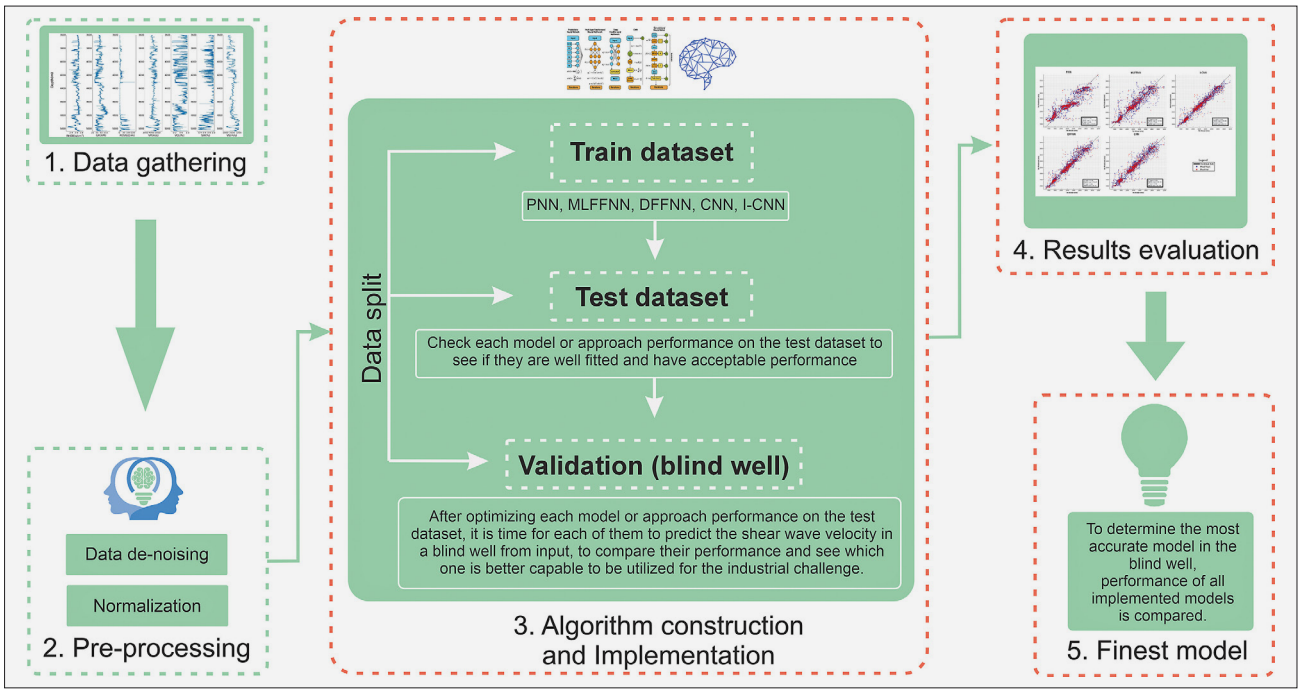


Figure 2: Procedure of the machine and deep learning workflow for V_s prediction

Building on this design, the I-CNN architecture incorporated three parallel convolutional streams with distinct kernel sizes – representing small, medium, and large receptive fields to simultaneously capture fine-scale, intermediate, and long-range depth patterns. Each stream consisted of Conv → BatchNorm → Activation → Dropout blocks, followed by the concatenation of feature maps and regression through fully connected layers (**Rajabi et al., 2023; Olutoki et al., 2024**). This multi-scale fusion strategy was devised to reconcile the competing requirements of resolving thin stratigraphic layers and modelling broader basin-scale variations, effectively addressing the single-scale limitation of conventional CNNs (**Wang et al., 2022; Alqahtani et al., 2018**).

All neural network models were trained using mean squared error (MSE) loss with adaptive optimizers (Adam or equivalent) and controlled via early stopping and validation patience to prevent overfitting. The PNN model employed MSE and likelihood-based criteria for optimal smoothing (σ) selection (**Earp et al., 2020; Ghorbani et al., 2025**).

To evaluate robustness and reproducibility, the RCW-1 dataset was split into randomized 70:30 train–test partitions and repeated for 150 independent random seeds, producing statistical distributions of performance metrics (R^2, RMSE, MAE). This repeated-split approach mitigates endpoint selection bias and allows a fairer comparison of algorithmic stability (**Rajabi et al., 2023**). The best-performing models (selected based on validation R^2 and lowest RMSE) are then applied directly to RCW-2 without retraining for strict cross-well validation. The differences between in-sample and validation performance were analyzed to assess model transferability, lithological sensitivity, and resilience to variations in log distributions (**Wang et al., 2022; Olutoki et al., 2024**).

Comprehensive diagnostics and visualization tools, including scatter plots, residual histograms, multi-track depth logs, and feature-importance or standardized coefficient analyses were used to interpret results and identify depth intervals where model predictions diverge. These analyses enable transparent assessment of model behaviour and provide practical guidance for field implementation (**Yang et al., 2023; Shi et al., 2025**).

All models were implemented and evaluated in MATLAB (R2023b/2024a) using reproducible scripts. Fixed random seeds ensured deterministic results, while explicit seed sweeps supported robustness checks. Final deliverables included trained model files, standardized performance tables, and project-specific folders containing: Visualization figures (scatter, residuals, learning curves), Excel result sheets (depth, actual V_s, predicted V_s, residuals), and all code used for training and evaluation, thereby facilitating reproducibility and future independent validation (**Rajabi et al., 2023; Ghorbani et al., 2025**).

3.7. Data gathering and preparation

The dataset used in this study consists of conventional wireline logs acquired from two anonymized wells, referred to as RCW-1 and RCW-2, provided with formal authorization from the data owner exclusively for academic research purposes. Both wells penetrate comparable stratigraphic intervals within a siliciclastic sand-

stone–shale succession, ensuring geological consistency. At the same time, RCW-1 and RCW-2 exhibit differences in shale volume distribution, resistivity response, and depth coverage, which provide sufficient heterogeneity to evaluate cross-well generalization. Consequently, RCW-1 is used for model training and testing, while RCW-2 serves as a blind validation well, which is reserved exclusively for independent validation to evaluate the cross-well generalization capability of each predictive model. This partitioning strategy aligns with the best practices for supervised learning in geoscientific applications, minimizing information leakage and ensuring that model performance reflects true predictive capability rather than the memorization of site-specific patterns (**Rajabi et al., 2023; Ghorbani et al., 2025; Wang et al., 2022**). Each well includes conventional logging measurements acquired during open-hole conditions, as well as from well log analysis (e.g. clay volume and water saturation). The recorded logs comprise gamma ray (GR), bulk density (RHOB), deep resistivity (ILD), compressional sonic (V_p), clay volume (VCL), and water saturation (SW) (see **Table 1**). These parameters were selected as input features because they are standard in formation evaluation and provide complementary sensitivity to key petrophysical properties such as lithology, porosity, and fluid type (**Gardner et al., 1974; Castagna et al., 1985; Chen et al., 2022**). Feature selection was guided by both statistical assessment and geological reasoning (see **Figure 3**). Correlation analysis was used to evaluate potential redundancy among predictors; however, no geologically meaningful features were removed solely based on correlation thresholds. The objective was not aggressive dimensionality reduction, but the preservation of physically interpretable input variables. All retained features (GR, RHOB, ILD, *Vp*, VCL, and SW) have well-established petrophysical relevance to elastic properties, as demonstrated in previous studies, and therefore contribute complementary information to V_s prediction. The target variable, V_s, was derived from dipole sonic log measurements where available and treated as ground-truth calibration data for supervised learning. Prior to model training, Vs logs underwent quality control procedures, including the removal of isolated spikes associated with tool noise and mild smoothing using a short-window median filter. No anisotropy correction was applied due to the absence of cross-dipole measurements. These preprocessing steps ensure that Vs values represent reliable elastic properties suitable for model calibration and validation. Dipole sonic data are widely recognized as the most reliable source for V_s calibration, as they provide high-resolution information on the formation's elastic properties, an essential component for rock physics and geomechanical analysis (**Tonni and Simonini, 2013**).

The well logs were acquired under standard wireline logging practices using industry-grade sensors calibrated prior to field acquisition to ensure consistent response curves and accurate depth calibration (**Shi et al., 2025**). Logging tools were operated under controlled borehole conditions to minimize measurement artifacts caused by mud invasion, washouts, or tool eccentricity. These conditions minimize non-geological influences on log responses and ensure that observed variations primarily reflect subsurface properties rather than acquisition artifacts. Each raw log was then subjected to initial quality control (QC) at the acquisition stage, where non-physical spikes, depth mismatches, and tool dropout zones were flagged for correction or removal. This QC step was necessary to reduce the risk of introducing spurious features that could mislead the learning algorithms (**Zeng et al., 2022; Olutoki et al., 2024**).

Following data acquisition, all log curves were depth-matched and resampled to a consistent vertical resolution of 0.15 m using linear interpolation to align measurements across logs with different sampling rates (**Rajabi et al., 2023**). Missing intervals, particularly in resistivity and sonic logs, were imputed using localized regression and median filtering where appropriate, while severely incomplete intervals were excluded from training to preserve data integrity (**Geng and Wang, 2020**). Outlier detection was performed using a combination of z-score analysis and interquartile range (IQR) thresholds to identify values exceeding three standard deviations from the mean. Detected anomalies were visually inspected against neighbouring depth intervals to distinguish between geological anomalies and measurement errors (**Wang et al., 2022; Alqahtani et al., 2018**).

To ensure comparability among features, all logs were normalized prior to modelling. Normalization was carried out using min–max scaling to a [0,1] range and z-score standardization for networks sensitive to scale differences, such as the DFFNN and CNN models (**Tolstaya et al., 2023; Khan and Kirmani, 2024**). This step prevents disproportionate weighting of high-magnitude logs (e.g. resistivity) and enhances convergence stability during training. For each well, normalization parameters were computed from the training data of RCW-1 and applied consistently to RCW-2 to simulate real-world deployment where models are calibrated on a reference RCW-1 and used for prediction on other wells within the same basin (**Rajabi et al., 2023; Olutoki et al., 2024**).

The lithological context of the wells was confirmed through the integration of well reports by the operating company, which identify alternating of sand–shale sequences typical of fluvial to shallow-marine depositional environments. This geological heterogeneity makes the dataset ideal for testing the adaptability of different neural network architectures in capturing nonlinear relationships between petrophysical properties and V_s (**Li et al., 2025; Ali and Sheng-Chang, 2020**). In total, RCW-1 contained approximately 3,000 valid data points after cleaning, while RCW-2 contained about 2,500 samples used solely for validation. The resulting dataset thus provides both sufficient volume for deep model training and

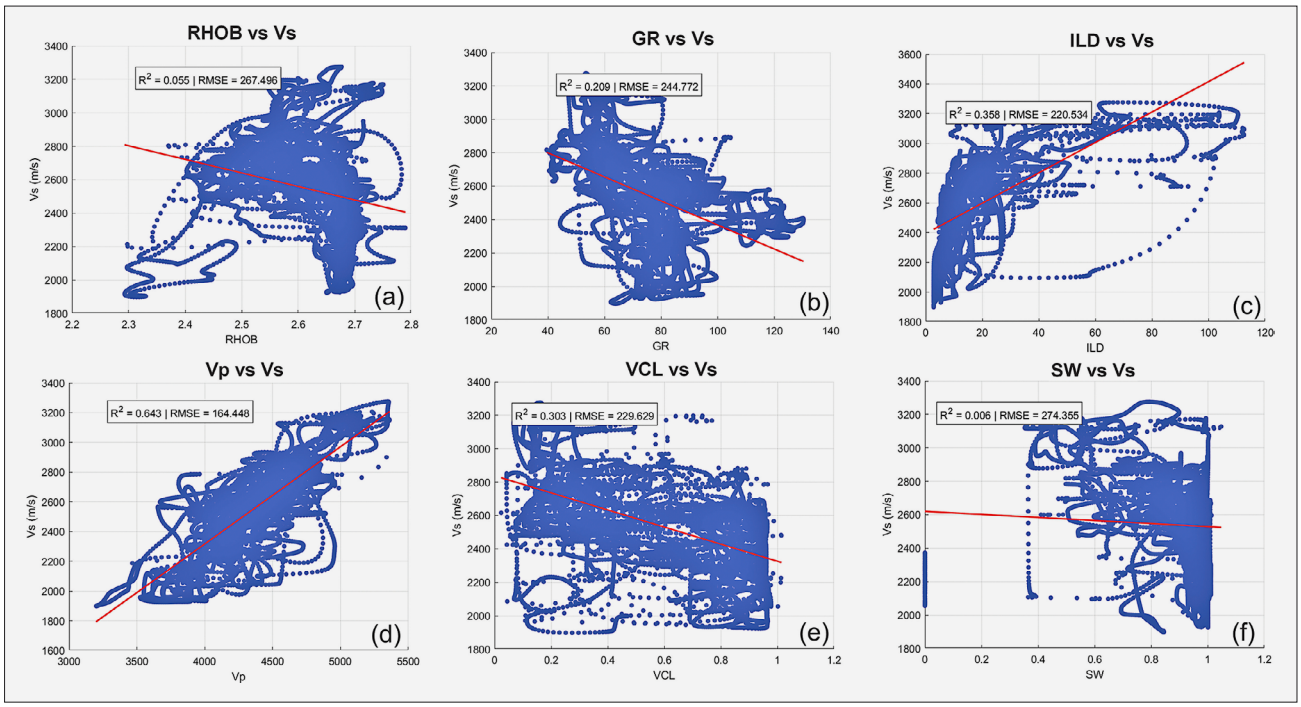


Figure 3: Scatter plots showing pairwise relationships among (a) gamma-ray (GR), (b) bulk density (RHOB), (c) deep resistivity (ILD), (d) compressional velocity (V_p), (e) clay volume (VCL), and (f) water saturation (SW) used as predictors for V_s. The proximity of data points to the 1:1 line indicates prediction accuracy, whereas systematic deviations reflect model bias or under- or over-estimation.

Table 1: Summary of petrophysical input parameters used for model training and validation

Well	Variables	Bulk Density	Gamma Ray	Deep Resistivity	Compressional Wave Velocity	Shear Wave Velocity	Clay Volume	Water Saturation
	Symbol (units)	RHOB (g/cc)	GR (GAPI)	ILD (ohms-m)	V_p (m/s)	V_s (m/s)	VCL (%)	SW (%)
RCW-1	Mean	2.62	75.92	14.52	4343.87	2541.89	0.57	0.86
	Std. Deviation	0.07	17.47	16.09	337.22	275.15	0.29	0.22
	Variance	-0.00	-305.30	-259.20	-113721.80	-75708.70	-0.08	-0.05
	Minimum	2.29	39.54	2.82	3199.07	1897.87	0.02	0.35
	Maximum	2.78	130.42	112.59	5362.23	3275.43	1.00	1.00
RCW-2	Mean	2.59	109.53	4.75	3912.50	2195.69	0.46	0.92
	Std. Deviation	0.09	21.76	8.99	443.79	336.21	0.28	0.15
	Variance	0.01	473.84	80.99	196949.99	113039.85	0.08	0.02
	Minimum	2.10	35.55	0.10	2892.55	1500.89	0.04	0.21
	Maximum	3.41	154.10	98.89	6154.53	3196.71	1.00	1.00

complexity to evaluate robustness under varying lithologic and stratigraphic conditions.

Figure 3 presents exploratory cross-plots between *Vs* and the selected input well logs using data from the RCW-1 well. This well was chosen for exploratory analysis because it provides the reference dataset for model training and testing. The blue dots represent smoothed data trends that reflect the dominant petrophysical relationships between *Vs* and each predictor, while the red line show fitted regression trends used to assess the general behaviour of the data. The observed patterns confirm that *Vs* exhibits strong dependence on elastic and density-related parameters, particularly *Vp* and RHOB (see **Figure 3a** and **3d**), while GR, VCL, and SW primarily reflect lithological and fluid-related influences. These relationships provide physical justification for the inclusion of the selected features in subsequent machine-learning modelling.

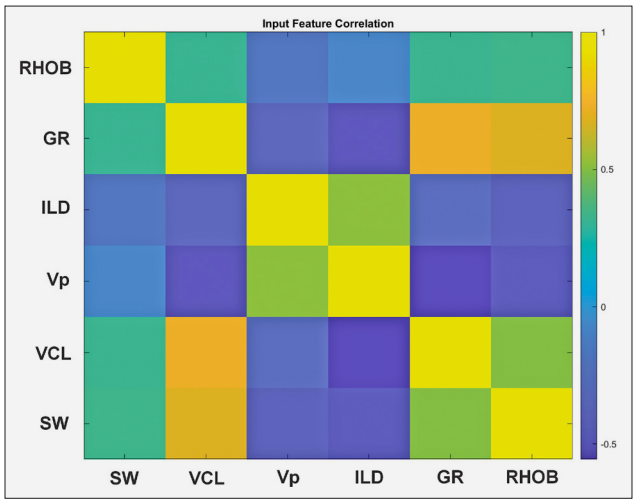


Figure 4: Correlation matrix of the input features used for V_s prediction, including RHOB, GR, ILD, Vp, VCL, and SW. Colours represent Pearson correlation coefficients, ranging from –1 (strong negative correlation, dark blue) to +1 (strong positive correlation, yellow), as indicated by the colour bar. The diagonal elements correspond to self-correlation.

The analysis of traditional well-logging data in this work uses two feature selection methods: non-relevant feature removal and heat-map correlation. Heatmap correlation identifies redundant or irrelevant elements by utilizing a correlation matrix to determine the relationships between them. The heatmap correlation matrix of every feature that is accessible is shown in **Figure 4**. That figure shows the correlation matrix of the input features used for *Vs* prediction. Several expected petrophysical relationships are observed, including a strong positive correlation between GR and VCL, reflecting their shared sensitivity to clay content. RHOB exhibits moderate correlation with GR and VCL, consistent with lithological and compaction effects. Compressional-wave velocity (*Vp*) shows moderate correlation with ILD, indicating coupled elastic and fluid-related influences, while negative correlations between *Vp* and GR–VCL highlight contrasts between sand-dominated and shale-rich intervals. As described in the methodology section, this correlation analysis was used to assess potential redundancy; however, no geologically meaningful features were removed solely based on correlation magnitude. All inputs were retained to preserve physical interpretability and complementary petrophysical information.

Finally, to ensure reproducibility and data governance, all preprocessed logs for both wells were stored in structured Excel files (.xlsx) with standardized naming conventions. Each file contains separate sheets for depth, log inputs, predicted outputs, and residuals, allowing direct interfacing with MATLAB workflow for model training, evaluation, and visualization. This standardized data management ensures full traceability from raw acquisition to final model output, supporting transparent scientific workflow practices as advocated in recent geoinformatics research (**Alqahtani et al., 2018; Rajabi et al., 2023**).

3.8. Neural network implementation

3.8.1. Architecture

This study employs five distinct neural network architectures to evaluate their performance in predicting V_s from conventional well log inputs: Probabilistic Neural Network (PNN), Multi-Layer Feedforward Neural Network (MLFFNN), Deep Feedforward Neural Network (DFFNN), Convolutional Neural Network (CNN), and Integrated CNN (I-CNN). These models were selected to represent both shallow and deep learning paradigms with varying levels of complexity and learning capacity.

The PNN model is based on a statistical approach that uses kernel-based probability estimation and Bayes' decision theory for classification and regression tasks (**Specht, 1990**). In this implementation, the PNN receives six normalized log inputs: density (RHOB), gamma ray (GR), resistivity (ILD), compressional velocity (V_p), clay volume (VCL), and water saturation (SW), and propagates them through the radial basis function layer. The output layer integrates these probabilistic contributions to estimate the target V_s value. The model's fast training and strong generalization ability on smaller datasets make it attractive for geophysical applications where noise and non-linearity are present (**Gu et al., 2018; Ali and Sheng-Chang, 2020**).

The MLFFNN, a classical neural network, is structured with a single hidden layer containing 10 to 20 neurons, activated by a sigmoid or ReLU function. Inputs are fully connected to the hidden layer, which is then connected to a single linear output neuron representing V_s. This architecture is trained using backpropagation and stochastic gradient descent, minimizing the mean squared error between predicted and actual values. Although limited in representational capacity, MLFFNNs remain robust for many regression tasks and have been successfully applied in petrophysical property prediction (**Silva et al., 2013**).

To capture more complex, nonlinear relationships, the DFFNN extends the MLFFNN by incorporating two to four hidden layers with increasing neuron counts (e.g. 32–64–128). Each layer utilizes the ReLU activation function, and batch normalization is applied to stabilize learning and accelerate convergence. Dropout layers are also included to reduce overfitting. These enhancements allow the DFFNN to model deeper representations of the petrophysical log–V_s relationship, particularly in heterogeneous formations (**Khan and Kirmani, 2024; Shi et al., 2025**).

The CNN architecture leverages spatial locality in well logs by treating the input features as sequential data. A typical CNN in this study includes one-dimensional convolutional layers with kernel sizes of 2 or 3,

Table 2: Statistical summary of the input dataset for Wells RCW-1 (training & test) and RCW-2 (validation)

Statistical parameters	PNN	MLFFNN	DFFNN	CNN	I-CNN
	Training Set				
RMSE	113.21	114.56	57.42	63.18	46.21
R^2	0.83	0.83	0.96	0.95	0.97
Number of Data	9,234	9,234	9,234	9,234	9,234
	Test Set				
RMSE	110.63	110.76	61.72	65.68	51.03
R^2	0.84	0.84	0.95	0.94	0.96
Number of Data	3,927	3,927	3,927	3,927	3,927
	Validation Set				
RMSE	133.87	120.04	81.37	91.39	80.73
R^2	0.85	0.87	0.94	0.92	0.95
Number of Data	5,930	5,930	5,930	5,930	5,930

followed by max-pooling layers to reduce dimensionality. Convolutional filters automatically learn localized feature interactions among the log inputs, which is useful in capturing abrupt lithological transitions. The CNN is capped by fully connected layers to regress the V_s output. CNNs have proven effective in geophysical signal analysis and log data modelling, providing both robustness and interpretability (**Chen et al., 2020**).

Finally, the Integrated CNN (I-CNN) architecture enhances the standard CNN by introducing three consecutive convolution–batch normalization–dropout (Conv–BN–Dropout) blocks before the fully connected output stage. This stacked design increases the model's ability to extract hierarchical features from the input logs, capturing both low-level and high-level signal characteristics. Each convolutional block contains multiple filters (e.g. 16–32–64), followed by 1D batch normalization and dropout with a rate of 0.25 to prevent overfitting. The stacking approach allows the model to exploit both local and global features for improved generalization across wells. This architecture is particularly suited for learning from noisy, stratigraphically variable logs common in real-world datasets (**Alqahtani et al., 2018; Zeng et al., 2022**).

Each model receives the same six input features, normalized consistently, and is trained on the RCW-1 dataset using 70% of the data for training and 30% for internal testing. The RCW-2 dataset is reserved for independent validation. Optimization is performed using the Adam optimizer with a learning rate tuned for each model. Loss is computed using root mean squared error (RMSE), and evaluation metrics include R^2 and RMSE on both test and validation sets. This standardized modelling framework enables a fair comparison of architecture performance while highlighting the strengths of deep convolutional designs for V_s estimation.

In addition to the network architecture, model training required the specification of several hyperparameters to ensure stable convergence and reproducibility. Hyperparameter tuning was conducted using a Bayesian optimization strategy, which has been shown to be efficient for high-dimensional search spaces. For all neural network models, the learning rate was fixed at 1×10^{-3}, which provided a stable balance between convergence speed and training stability. Model training was performed using a mini-batch size of 64 samples, selected to optimize computational efficiency while maintaining gradient stability. Each network was trained for a maximum of 150 epochs, with an early-stopping criterion applied when the validation loss did not improve for 15 consecutive epochs, thereby preventing overfitting. Model convergence was monitored based on the stabilization of the RMSE loss function on the validation set. The final hyperparameter configuration was selected based on the minimum validation loss obtained during the optimization process. This procedure ensured that the reported model performance reflects optimized and reproducible training conditions rather than arbitrary parameter selection. The same hyperparameter tuning strategy and convergence criteria were applied consistently across all neural network architectures to ensure a fair comparison between models.

3.8.2. Training, test, and validation processes

The networks described above were trained using RMSE as the loss function. In order to help avoid overfitting of the networks, the early halting strategy was also applied in the final fitting. Following pre-processing, the validation data was fed into the model that ultimately performed the best across the train and test datasets. Each model's output was stored for blind RCW-2 analysis and assessment.

Each model receives the same six input features, normalized consistently, and is trained on the RCW-1 dataset using 70% of the data for training and 30% for internal testing. The RCW-2 dataset is reserved for independ-

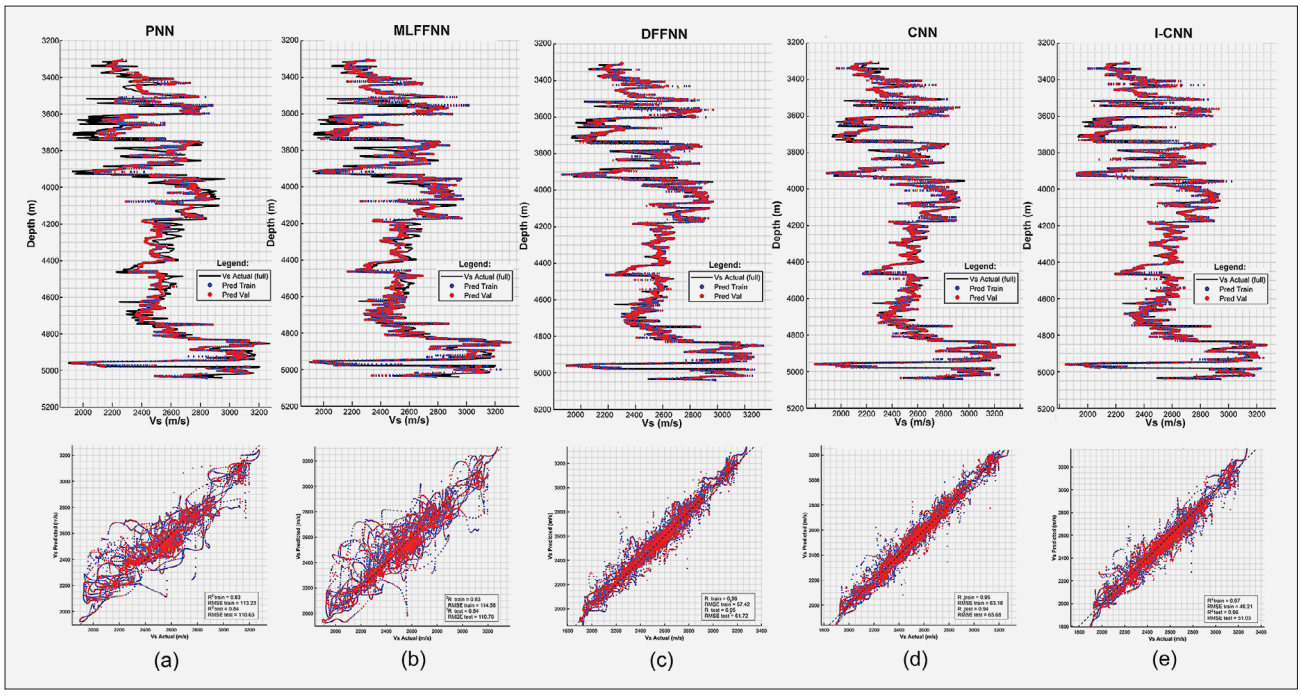


Figure 5: Comparison between measured *Vs* and predicted *Vs* obtained from five neural network architectures: (a) PNN, (b) MLFFNN, (c) DFFNN, (d) CNN, and (e) I-CNN in RCW-1 well. For each model, the upper panel shows depth-based *Vs* profiles, where the black curve represents measured *Vs*, blue markers denote training predictions, and red markers indicate test predictions. The lower panel presents cross-plots of predicted *Vs* versus measured *Vs*, where the dashed diagonal line corresponds to the ideal 1:1 relationship. Depth-wise comparison of measured and predicted V_s for training and testing phases, demonstrating that deep architectures (DFFNN, CNN, I-CNN) better capture stratigraphic variations than shallow networks (PNN, MLFFNN). The closer alignment of the I-CNN results to higher predictive accuracy and minimal bias across the training and testing datasets

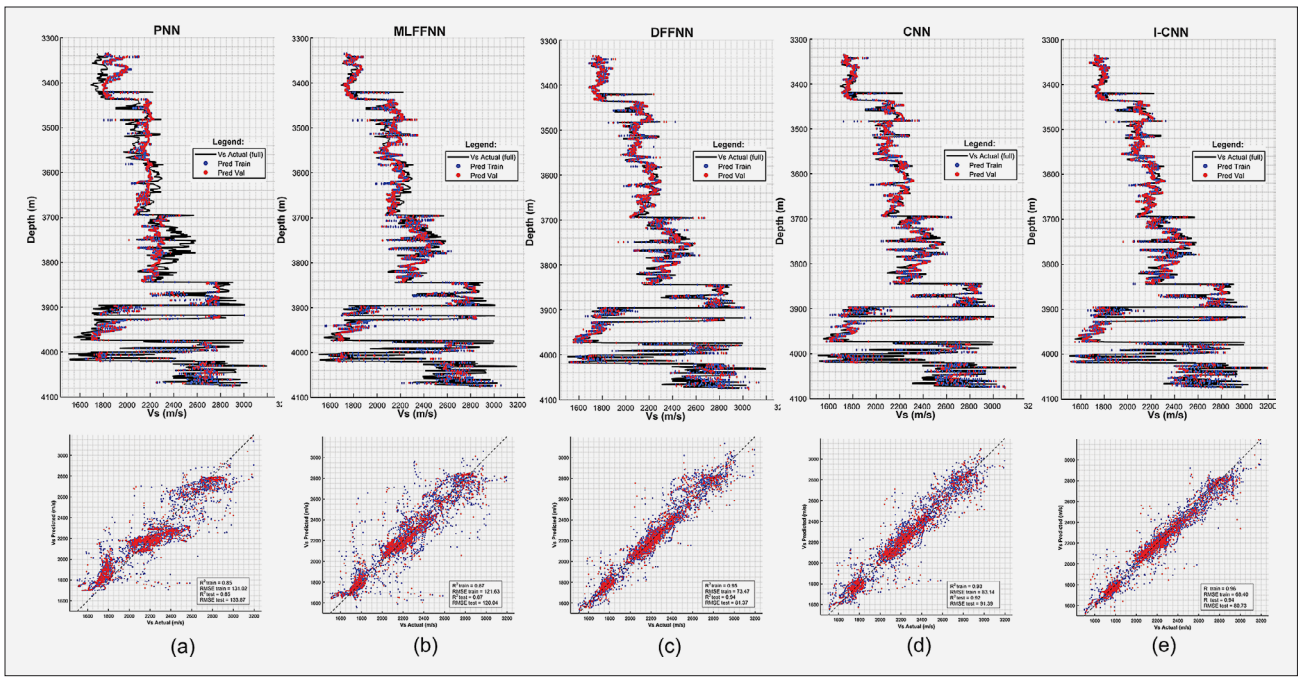


Figure 6: Validation well performance showing measured V_s alongside model predictions obtained from five neural network architectures: (a) PNN, (b) MLFFNN, (c) DFFNN, (d) CNN, and (e) I-CNN in RCW-2 well. The lower panel presents cross-plots of predicted *Vs* versus measured *Vs*, where the dashed diagonal line corresponds to the ideal 1:1 relationship. The I-CNN maintains the most geologically consistent response, confirming strong cross-well generalization. Blind-well validation (RCW-2) cross-plot showing correlation between measured and predicted V_s values. The I-CNN displays the tightest clustering around the 1:1 line, reflecting robust transfer learning from Well RCW-1

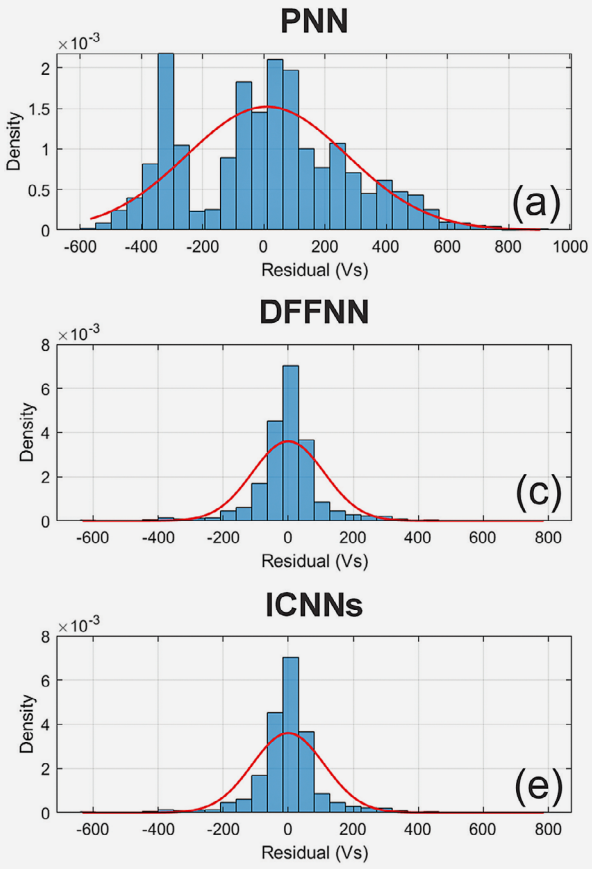


Figure 7: Residual distributions of V_s predictions for the five neural network architectures: (a) PNN, (b) MLFFNN, (c) DFFNN, (d) CNN, and (e) I-CNN. Residuals are defined as the difference between predicted *Vs* and measured *Vs*. Histograms represent the frequency distribution of residuals, while the red curves indicate fitted normal distributions. The horizontal axis shows residual values (m/s), and the vertical axis represents probability density.

ent validation. Optimization is performed using the Adam optimizer with a learning rate tuned for each model. Well RCW-2 was not involved in model training or hyperparameter optimization. It was used exclusively as an independent blind validation dataset to evaluate model generalization. Any validation metrics reported for RCW-2 reflect post-training model performance only. Loss is computed using RMSE, and evaluation metrics include R^2 and RMSE on both test and validation sets. This standardized modelling framework enables a fair comparison of architecture performance while highlighting the strengths of deep convolutional designs for V_s estimation.

3.9. Evaluation of models

The models' performances were evaluated using three metrics: R^2 as defined in **Equation 23** and RMSE as outlined in **Equation 6**. R^2 quantifies the extent to which independent factors account for the variation in the dependent variable, while RMSE assesses the average squared deviation between anticipated and actual values. RMSE assesses the performance of regression models, with lower values indicating superior performance (see **Table 2**).

$$R^2(a,p) = \frac{\Sigma_{i=1}^{n}(a_i - p_i)^2}{\Sigma_{i=1}^{n}(a_i - p_i)^2} \qquad (23)$$

4. Results

The statistical outcomes of the PNN, MLFFNN, DFFNN, CNNs, and I-CNNs on the training, testing, and validation datasets are presented in **Table 2**. **Figure 5** presents a comparison of the applied models throughout the training, testing, and validation datasets, evaluated using R-squared and RMSE metrics.

Figure 5 illustrates the performance of five neural network architectures by comparing predicted and measured V_s values in both depth and cross-plot domains. **Figures 5a** and **5b** show that the PNN and MLFFNN models reproduce the general *Vs* trend with depth; however, noticeable scatter and deviations from the measured profile are observed, particularly in intervals with rapid velocity variation. **Figures 5c–e** demonstrate improved agreement for the DFFNN, CNN, and I-CNN models. In these cases, predicted *Vs* profiles closely follow the measured Vs curve across most depth intervals, indicating enhanced capability to capture nonlinear elastic responses. This improvement is further confirmed in the corresponding cross-plots, where predictions cluster more tightly around the 1:1 line and exhibit reduced dispersion compared to simpler network architectures. Among all models, the CNN-based approaches show the most consistent performance for both training and blind validation datasets, suggesting superior generalization and reduced sensitivity to lithological heterogeneity. De-

viations observed at higher *Vs* values are minor and likely reflect local elastic contrasts rather than systematic model bias.

Figure 6 presents the blind cross-well validation results for RCW-2, providing a stringent test of model generalization beyond the training well. For clarity, training and validation labels shown for RCW-2 correspond to the evaluation of predictions generated by models trained on RCW-1 and do not indicate retraining or parameter updating using RCW-2 data. **Figures 6a** and **6b** show that the PNN and MLFFNN models reproduce the overall *Vs* trend but exhibit increased scatter and local mismatches, particularly in intervals with abrupt velocity changes. Improved agreement is observed for the DFFNN, CNN, and I-CNN models (see **Figures 6c–e**), where predicted *Vs* profiles closely follow the measured *Vs* curve across most depth intervals. This behaviour is corroborated by the corresponding cross-plots, which display tighter clustering around the 1:1 reference line and reduced dispersion relative to simpler architectures. Among the tested models, CNN-based approaches demonstrate the most consistent performance for RCW-2, indicating robust transferability of learned elastic–lithological relationships to an independent well. Minor deviations at higher *Vs* values are limited and likely reflect local stratigraphic or elastic contrasts rather than systematic model bias.

Figure 7 presents the residual distributions for all evaluated neural network models, providing a complementary assessment of prediction accuracy and bias. The PNN model (see **Figure 7a**) exhibits a relatively broad and asymmetric residual distribution, indicating higher variance and reduced stability. The MLFFNN model (see **Figure 7b**) shows a narrower distribution; however, minor skewness suggests residual bias in certain velocity ranges. In contrast, the DFFNN, CNN, and I-CNN models (see **Figures 7c–e**) display residual distributions that are more symmetric and tightly centered around zero. The reduced spread and near-Gaussian shape of these distributions indicate lower prediction error and improved robustness. Among these models, CNN-based architectures exhibit the most compact residual distributions, consistent with their superior performance observed in depth-based and cross-well validation results. The absence of pronounced skewness or multimodal behaviour in the residuals suggests that prediction errors are largely random rather than systematic, confirming that the models effectively capture the dominant elastic–lithological controls on *Vs*.

4. Discussion

The comparative results among the five neural network architectures: Probabilistic Neural Network (PNN), Multi-Layer Feedforward Neural Network (MLFFNN), Deep Feedforward Neural Network (DFFNN), Convolutional Neural Network (CNN), and Integrated I-CNNs, demonstrate the progressive advantages of deep learning approaches in modelling the nonlinear behaviour between petrophysical logs and V_s. Quantitatively, the I-CNN consistently outperformed the other architectures, achieving an R^2 of 0.97 on the training data, 0.96 on the testing data, and 0.95 on the validation dataset, while maintaining the lowest RMSE range of 46–81 m/s. These metrics clearly indicate that the I-CNN captures both fine-grained and global patterns in the input data more effectively than shallower or single-stack architectures (**Rajabi et al., 2023; Olutoki et al., 2024; Ghorbani et al., 2025**).

From a qualitative perspective, the I-CNN model exhibited superior capability in replicating the detailed depth-dependent variations of V_s, especially across heterogeneous intervals characterized by interbedded sandstone and shale facies. The architecture's multiple convolutional blocks allowed for hierarchical feature extraction, learning local-scale variations at early convolutional layers and broader stratigraphic dependencies in deeper layers. Consequently, the I-CNN yielded smoother and more geologically consistent V_s profiles compared to the DFFNN and CNN, which tended to underfit abrupt velocity transitions. In contrast, the PNN and MLFFNN models displayed higher prediction variance and instability across the validation dataset, implying overfitting to localized data features, as similarly reported by **Gu et al. (2018)**.

Deeper insight into the model convergence process can be derived from Figure 8, which illustrates the evolution of Mean Squared Error (MSE) versus epoch for both training and validation phases. The learning-curve behaviour differs among the evaluated models due to their architectural characteristics and optimization mechanisms. For the PNN model, the observed increase in MSE after an initial decrease reflects its sensitivity to the smoothing parameter and kernel-based formulation, which can lead to reduced generalization when model complexity increases beyond an optimal range. This behaviour is consistent with over-smoothing effects commonly reported for probabilistic neural networks when applied to heterogeneous datasets. In contrast, the MLFFNN, DFFNN, CNN, and I-CNN models exhibit monotonically decreasing or stabilizing MSE values with increasing epochs. These architectures employ gradient-based optimization with adaptive weight updates, enabling progressive refinement of nonlinear relationships between input features and shear-wave velocity. For the deeper models (DFFNN, CNN, and I-CNN), the validation curves closely overlap with the training curves and are therefore visually indistinguishable at the plotting scale. This overlap indicates stable convergence and minimal generalization gap rather than the absence of validation results. Differences in the number of training epochs among models and wells arise from model-specific convergence behaviour and early-stopping criteria. Each architecture reaches convergence at a different rate depending on network depth, parameter count,

and data complexity, and training was terminated once the validation error stabilized. Consequently, the reported epoch ranges reflect optimal convergence rather than uniform iteration counts.

The I-CNN displayed a rapid decline in training loss during the first 50 epochs, followed by a gradual stabilization as the model approached optimal convergence. The validation loss curve maintained a close trajectory to the training loss, confirming the absence of significant overfitting and indicating efficient generalization (**Alqahtani et al., 2018; Shi et al., 2025**). By contrast, shallower models such as the MLFFNN and DFFNN exhibited slower convergence rates and larger oscillations in validation MSE, suggesting a less stable optimization process and limited capacity to generalize nonlinear dependencies in the data. The consistently low validation MSE in I-CNN implies that the model not only learned the intricate relationships between input logs and V_s but also effectively suppressed noise inherent to field-acquired log data. These convergence dynamics reinforce the robustness of deep convolutional architectures for petrophysical regression tasks, particularly when managing datasets with mixed lithological signatures and measurement uncertainties (**Chen et al., 2022; Zeng et al., 2022**).

The hierarchical learning structure of the I-CNN, composed of repeated convolution–batch normalization–dropout stacks, proved particularly advantageous for identifying latent correlations among input parameters such as RHOB, GR, ILD, and V_p. The multiple convolutional filters captured subtle relationships associated with lithologic composition, porosity, and fluid content, enabling improved prediction fidelity across depth intervals. This hierarchical extraction capability explains the reduced residuals observed in the validation well and the model's ability to maintain performance across varying stratigraphic conditions (**Li et al., 2025; Ali and Sheng-Chang, 2020**). Conversely, the DFFNN and MLFFNN architectures, while computationally simpler, lacked the feature locality mechanisms present in CNN-based models, leading to less sensitivity to spatial transitions within the log sequences.

The strong contribution of Vp and RHOB identified through feature-importance analysis is therefore geologically meaningful and consistent with elastic theory under conditions of rapid burial and compaction. The superior performance of nonlinear models, especially I-CNN, can be attributed to their ability to capture nonlinear elastic responses arising from stratigraphic heterogeneity, compaction disequilibrium, and lithological transitions. Linear models are inherently limited in representing these effects, which explains their reduced performance in mixed sandstone–shale successions. Residual analysis further supports this geological interpretation. Larger residuals at higher *Vs* values are likely associated with well-compacted sandstone intervals, where small variations in cementation, grain sorting, or diagenetic alteration can result in disproportionate changes in elastic response. Increased residual dispersion at lower *Vs* values corresponds to shale-dominated or transitional intervals, where clay content, fluid pressure, and mechanical anisotropy introduce additional heterogeneity. The absence of systematic depth-related bias suggests that the model captures intrinsic lithological and compaction-related controls rather than merely learning stratigraphic position.

Despite its strong performance, the I-CNN architecture is not without limitations. The model's depth and complexity increase training time and computational demand, which may constrain its applicability in field-scale deployment where computational resources are limited. Additionally, the reliance on large, high-quality datasets makes I-CNN performance vulnerable to data incompleteness or inconsistent acquisition conditions, a common challenge in subsurface datasets (**Geng and Wang, 2020; Yang et al., 2023**). While normalization and quality control procedures mitigated these issues, slight discrepancies between RCW-1 (training/testing) and RCW-2 (validation) still influenced prediction residuals, suggesting that model generalization remains partly dependent on inter-well similarity. The limited interpretability of deep learning models also poses a barrier for operational geoscience applications, as model outputs cannot be easily traced back to explicit geological reasoning (**Alqahtani et al., 2018; Olutoki et al., 2024**).

The successful blind validation across wells indicates that the trained model learns stratigraphically transferable elastic–lithological relationships, which is particularly important in thick Cenozoic clastic successions where lateral facies variations are common but governed by similar depositional and burial processes. This transferability implies that the proposed approach is applicable to neighbouring stratigraphic intervals within the same generalized basin framework, even when direct shear-wave measurements are unavailable.

From a geophysical perspective, the ability to predict Vs reliably within such stratigraphic systems has important implications for seismic-to-well calibration, elastic impedance analysis, and rock-physics modelling. In rapidly subsiding basins, accurate Vs estimation is especially valuable for characterizing mechanical stratigraphy, overpressure development, and elastic contrasts associated with sandstone–shale alternations. Although the geological description is intentionally generalized to comply with data-use restrictions, the integration of stratigraphic understanding with machine-learning analysis demonstrates that statistically robust models can yield geologically meaningful insights when guided by appropriate geological context.

Overall, the study confirms that deep convolutional frameworks, particularly Integrated CNN (I-CNNs), provide a significant advancement over conventional probabilistic and feedforward neural networks for V_s prediction. The improved convergence behaviour, lower generalization error, and enhanced representation of lithologic variation support the growing consensus that I-CNNs are currently the most powerful tool for petrophysical parameter

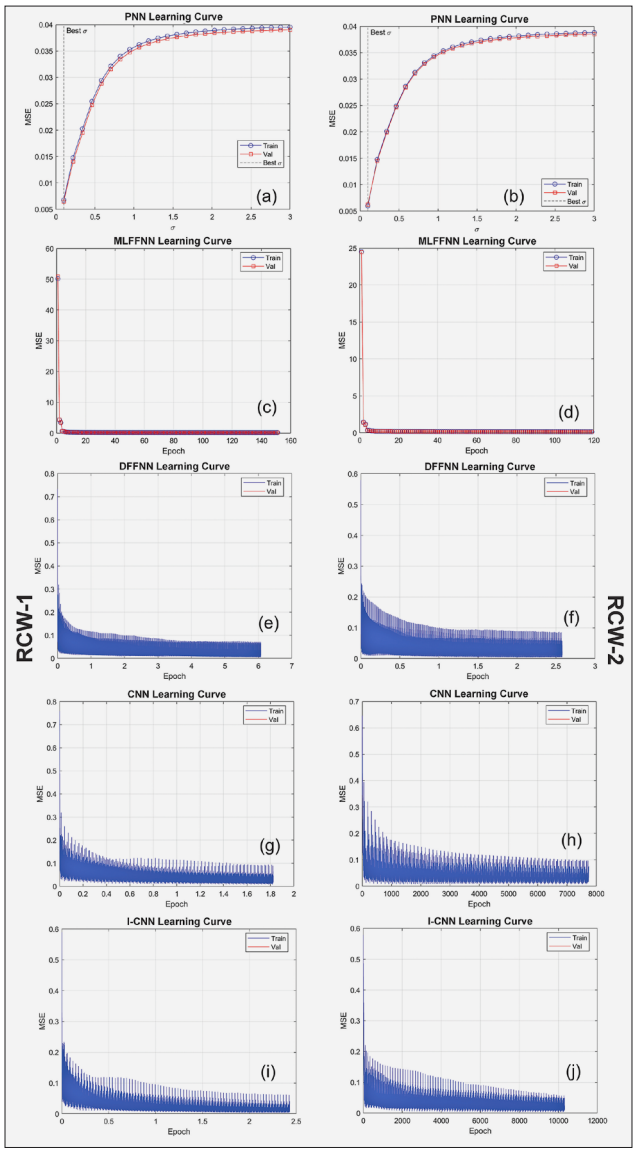


Figure 8: Learning curves showing the evolution of mean squared error (MSE) as a function of training epochs for the five neural network architectures evaluated in this study. Panels (a), (c), (e), (g), and (i) correspond to results obtained using the RCW-1 well (training and testing dataset), while panels (b), (d), (f), (h), and (j) show the corresponding results for the RCW-2 well (blind cross-well validation). The I-CNN converges rapidly within the first 50 epochs and stabilizes with minimal divergence between curves, confirming efficient generalization and absence of overfitting.

estimation in complex subsurface environments (**Rajabi et al., 2023; Ghorbani et al., 2025**).

5. Conclusions

This study demonstrates that among the five evaluated architectures: PNN, MLFFNN, DFFNN, CNN, and I-CNNs, the Integrated I-CNNs achieved the most accurate and geologically coherent shear wave velocity predictions from conventional well log data. Quantitative evaluation revealed that the I-CNNs consistently achieved the highest R^2 (0.95–0.97) and lowest RMSE (46–81 m/s) across all phases, indicating superior precision and generalization ability. Qualitatively, the I-CNNs reproduced the fine-scale lithologic transitions and global velocity trends with minimal residuals, outperforming both traditional and single-stack networks. The results verify that multi-stack convolutional architectures can effectively capture multi-scale dependencies between petrophysical logs and elastic parameters, bridging the gap between empirical rock physics models and modern data-driven methods.

Nevertheless, this research identifies several limitations and opportunities for improvement. The computational demand associated with deep multi-stack training remains substantial, emphasizing the need for optimized architectures that balance model depth and computational efficiency. Furthermore, the dataset used – though representative – was limited to two wells within a single basin, which restricts the generalization potential to other geological settings. Variability in acquisition parameters and incomplete logs continue to pose challenges for real-world applications. Additionally, while normalization ensured input consistency, the study found that model performance decreased slightly when validated on stratigraphically distinct intervals, underscoring the importance of adaptive calibration across multiple wells.

Future research should aim to extend the model to a basin-scale application by incorporating a larger and more diverse set of wells that capture variations in depositional environments and diagenetic conditions. This expansion would allow for testing the robustness of I-CNNs models under broader geological heterogeneity. Incorporating hybrid modelling strategies that combine deep neural architectures with physics-informed constraints could further enhance model interpretability and physical realism. Additionally, implementing transfer learning and attention-based mechanisms may improve the adaptability of I-CNNs models when applied to new fields with limited data availability. The integration of explainable AI frameworks, such as SHAP or Grad-CAM visualization, could provide new insight into feature importance, bridging the gap between black-box predictions and geological reasoning. Finally, coupling the I-CNNs with seismic or core image data may open new opportunities for holistic elastic property prediction and real-time subsurface monitoring, establishing a pathway toward comprehensive data-driven reservoir characterization.

In conclusion, the I-CNNs presents a transformative approach for V_s prediction, combining computational intelligence with geophysical interpretability. Through continued refinement in data integration, model optimization, and interpretability, this framework has the potential to become an indispensable tool in the digital transformation of subsurface modelling and reservoir evaluation.

Acknowledgement

For this research, I am profoundly appreciative of the support provided by DIPA FT research funding. I'd like

to thank all of my coworkers in the PPDG Laboratory at the University of Lampung's Department of Geophysics Engineering for their helpful discussions and ideas throughout this study. Lastly, we are grateful for the anonymous reviewer and editor's feedback and recommendations regarding the manuscript's enhancement.

Data Authorization and Ethical Compliance

All well-log data used in this study are authentic industrial datasets, provided with formal authorization from the data owner for exclusively academic research, including model training, testing, and validation. In accordance with the data-use agreement, geographical coordinates, field names, basin identifiers, and location-specific information are not disclosed. The data owner restricts the use of the dataset to methodological development and performance evaluation; therefore, the data are presented in an anonymized form, and only the derived results are reported. This limitation does not affect the scientific validity of the study, as the primary objective is to evaluate the methodological performance and generalization capability of machine-learning models rather than site-specific geological interpretation. The authors confirm full compliance with applicable data-use agreements and publication-ethics standards. No confidential, proprietary, or commercially sensitive information is disclosed, and no personal data is involved.

Funding

The University of Lampung has provided financial support for this investigation through the DIPA FT research project (No.4211/UN26.15/LK.03/2025).

6. References

Alakeely, A., & Horne, R. N. (2020a). Simulating the behaviour of reservoirs with convolutional and recurrent neural networks. *SPE Reservoir Evaluation & Engineering*, *23*(03), 0992-1005. https://doi.org/10.2118/201193-PA

Alakeely, A. A., & Horne, R. N. (2020b). Application of deep learning methods in evaluating well production potential using surface measurements. In *SPE Annual Technical Conference and Exhibition* (p. D031S022R008). SPE.

Alkinani, H. H., Al-Hammedi, & Dunn-Norman, S. (2021). Data-driven recurrent neural network model to predict the rate of penetration: *Upstream Oil and Gas Technology, 7* (100047) https://doi.org/10.1016/j.upstre.2021.100047.

Al-Mudhafar, W. J. (2017). Integrating well log interpretations for lithofacies classification and permeability modeling through advanced machine learning algorithms. Journal of Petroleum Exploration and Production Technology, 7(4), 1023–1033. https://doi.org/10.1007/s13202-017-0360-0

Al-Mudhafar, W. J., & Bondarenko, S. (2015). Integrating K-means clustering analysis and generalized additive model for efficient reservoir characterization. *In: 77th EAGE conference and exhibition incorporating SPE EUROPIC*, Madrid, Spain. https://doi.org/10.3997/2214-4609.201413024

Al-Mudhafar, W. J., He, H., & Sun, J. (2022). Deep learning for lithofacies classification and petrophysical property prediction. Journal of Petroleum Science and Engineering, 208, 109352. https://doi.org/10.1016/j.petrol.2021.109352

Alzubaidi, L., Zhang, J., & Humaidi, A. J. (2020). Deep learning models for lithology classification from core images. Neural Computing and Applications, 32(21), 16105–16124. https://doi.org/10.1007/s00521-020-04885-8

Alqahtani, N., Armstrong, R. T., & Peyman, M. (2018). Deep learning convolutional neural networks to predict porous media properties. *Paper presented at the SPE Asia Pacific Oil and Gas Conference and Exhibition*, Brisbane, Australia, October. doi: https://doi.org/10.2118/191906-MS

Babadagli, T., & Al-Salmi, S. J. S. R. E. (2004). A review of permeability-prediction methods for carbonate reservoirs using well-log data. *SPE Reservoir Evaluation & Engineering*, *7*(02), 75-88.

Blake, R.W., Mathew, R., George, A., & Papakostas, N. (2021). Impact of artificial intelligence on engineering: past, present and future. Procedia CIRP. Pages 1728-1733, https://doi.org/10.1016/j.procir.2021.11.291.

Brown, W. M., Gedeon, T. D., Groves, D. I., & Barnes, R. G. (2000). Artificial neural networks: a new method for mineral prospectivity mapping. *Australian Journal of Earth Sciences*, *47*(4), 757–770. https://doi.org/10.1046/j.1440-0952.2000.00807.x

Castagna, J. P., Batzle, M. L., & Eastwood, R. L. (1985). Relationships between compressional-wave and shear-wave velocities in clastic silicate rocks. Geophysics, 50(4), 571–581. https://doi.org/10.1190/1.1441933

Chen, G., Chen, M., Hong, G., Lu, Y., Zhou, B., & Gao, Y. (2020). A new method of lithology classification based on convolutional neural network algorithm by utilizing drilling string vibration data. *Energies*, 13(4), 888. https://doi.org/10.3390/en13040888

Dolatshahi, A. & Molladavoodi, H. (2024). Prediction of rock tensile fracture toughness: hybrid ANN-WOA model approach. (2024). *Rudarsko-geološko-Naftni Zbornik*, *39*(3), 1-12. https://doi.org/10.17794/rgn.2024.3.1

Durrani, M. Z. A., Talib, M., Ali, A., Sarosh, B., & Rahman, S. A. (2021). Characterization of carbonate reservoir using post-stack global geostatistical acoustic inversion approach: A case study from a mature gas field, onshore Pakistan, *Journal of Applied Geophysics*, 188 (104313), https://doi.org/10.1016/j.jappgeo.2021.104313.

Earp, S., Curtis, A., Zhang, X., & Hansteen, F. (2020). Probabilistic neural network tomography across Grane field (North Sea) from surface wave dispersion data. *Geophysical Journal International*, 223(3), 1741–1757. https://doi.org/10.1093/gji/ggaa328

Eriavbe, F. E., & Okene, U. O. (2019). Machine learning application to permeability prediction using log & core measurements: A realistic workflow application for reservoir characterization. In *SPE Nigeria Annual International Conference and Exhibition* (p. D033S017R001).

Faust, L. Y. (1953). A velocity function including lithologic variation. Geophysics, 18(2), 271–288. https://doi.org/10.1190/1.1437869

Gardner, G. H. F., Gardner, L. W., & Gregory, A. R. (1974). Formation velocity and density: the diagnostic basics for

stratigraphic traps. Geophysics, 39(6), 770–780. https://doi.org/10.1190/1.1440465

Ghorbani, H., Hazbeh, O., Rajabi, M., Tabasi, S., Lajmorak, S., Alvar, M. A., Radwan, A. E. (2025). Prediction of formation shear wave velocity using a new hybrid deep learning algorithm. *Physics and Chemistry of the Earth, Parts A/B/C*, 140, 104063. https://doi.org/10.1016/j.pce.2025.104063

Goodfellow, I., Bengio, Y., Courville, A. (2016). Deep learning. MIT Press, Cambridge, MA.

Greenberg, M. L., & Castagna, J. P. (1992). Shear-wave velocity estimation in porous rocks: Theoretical formulation, preliminary verification and applications. Geophysical Prospecting, 40(2), 195–209. https://doi.org/10.1111/j.1365-2478.1992.tb00371.x

Gu, Y., Bao, Z., & Rui, Z. (2018). Complex lithofacies identification using improved probabilistic neural networks. *Petrophysics*, *59*(02), 245-267. https://doi.org/10.30632/PJV59N2-2018a9

Guo, K., Wu, M., Li, X., Nauman, A., Li, C., Zhu, M. (2025). NOMA-Enabled RIS-Assisted SATINs under multi-objective optimization: a deep reinforcement learning approach. In: Jamshed, M.A., Nauman, A. (eds) Integrated Terrestrial and Non-Terrestrial Networks. *Synthesis Lectures on Communications*. Springer, Cham. https://doi.org/10.1007/978-3-031-63705-6_7

Hampson, D. P., Schuelke, J. S., & Quirein, J. A. (2001). Use of multiattribute transforms to predict log properties from seismic data. *Geophysics*. 66. 10.1190/1.1444899.

Khan, M., Ullah, Z., Mašek, O., Raza N. S., & Nouman A. K. M. (2022). Artificial neural networks for the prediction of biochar yield: A comparative study of metaheuristic algorithms. *Bioresource technology*, *355*, 127215. https://doi.org/10.1016/j.biortech.2022.127215

Kellogg, R. P., Chessum, W., & Kwong, R. (2018). Machine learning application for wellbore damage removal in the wilmington field. In *SPE Western Regional Meeting* (p. D041S011R001)

LeCun, Y., Bengio, Y. & Hinton, G. (2015). Deep learning. *Nature* 521, 436–444. https://doi.org/10.1038/nature14539

Li, Z. & Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*. PP. 1-21. 10.1109/TNNLS.2021.3084827.

Li, Y., Zhu, Y., Li, Z., Zhang, X., & Cai, G. (2025). Reservoir parameter prediction technology based on deep learning and its application in the Panyu 4 Sag, Pearl river mouth Bain. Front. Earth Sci. 13:1512811. doi: 10.3389/feart.2025.1512811

Lei, L., Hao, W., Tiantian, W., Pengyun, Z., Zhi, Z., & Chenglong, L. (2023). Missing well-log reconstruction using a sequence self-attention deep-learning framework. Geophysics, 88 (6): D391–D410. doi: https://doi.org/10.1190/geo2022-0757.1

Mahmoudian, M. S., Shiri, Y., & Vaezian, A. (2022). Prediction of shear wave velocity and modification of Castagna and Carroll relationships in one of the Iranian oil fields. *Rudarsko-geološko-Naftni Zbornik*, *37*(4), 137-144. https://doi.org/10.17794/rgn.2022.4.11

Mohebali, B., Tahmassebi, A., Meyer-Baese, A., & Gandomi, A. H. (2020). Probabilistic neural networks. Handbook of Probabilistic Models. 347–350. DOI: 10.1016/b978-0-12-816514-0.00014-x

Negash, B. M., & Yaw, A. D. (2020). Artificial neural network based production forecasting for a hydrocarbon reservoir under water injection. *Petroleum Exploration and Development*, *47*(2), 383-392.

Olutoki, J.O., Zhao, J., Siddiqui, N.A., Elsaadany, M., Haque, A. K. M. E., Akinyemi, O. D., *et al.* (2024). Shear wave velocity prediction: A review of recent progress and future opportunities. Energy Geosciences, 5(4), 100338. https://doi.org/10.1016/j.engeos.2024.100338

O'Shea, K., & Nash, R. (2015). An Introduction to Convolutional Neural Networks. ArXiv, abs/1511.08458.

Pickett, G.R. (1963) Acoustic character logs and their application in formation evaluation. *Journal of Petroleum Technology*, 15, 659-667. http://dx.doi.org/10.2118/452-PA

Raghavan, G., Tharwat, B., Hari, S.N. *et al.* (2024). Engineering flexible machine learning systems by traversing functionally invariant paths. *Nat Mach Intell* 6, 1179–1196. https://doi.org/10.1038/s42256-024-00902-x

Rajabi, M., Hazbeh, O., Davoodi, S. *et al.* (2023). Predicting shear wave velocity from conventional well logs with deep and hybrid machine learning algorithms. *J Petrol Explor Prod Technol* 13, 19–42. https://doi.org/10.1007/s13202-022-01531-z

Rumelhart, D., Hinton, G. & Williams, R. (1986). Learning representations by back-propagating errors. *Nature* 323, 533–536. https://doi.org/10.1038/323533a0

Saikia, P., Baruah, R. D., Singh, S. K., & Chaudhuri, P. K. (2020). Artificial Neural Networks in the domain of reservoir characterization: A review from shallow to deep models. *Computers & Geosciences*, *135*, 104357.

Shi, Y., Liao, J., Gan, L., & Tang, R. (2024). Lithofacies prediction from well log data based on deep learning: a case study from southern sichuan, china. *Applied Sciences*, 14(18), 8195. https://doi.org/10.3390/app14188195

Silva, D. F., Moazzeni, S., & Haffar, A. (2013). Backpropagation neural network for lithology prediction using well logs. *2nd International Conference on Pattern Recognition Applications and Methods (ICPRAM 2013)*, 701–708. https://doi.org/10.5220/0004380707020706

Specht, D. F. (1990). Probabilistic neural networks. *Neural Netw*. 3 (1), 109–118. https://doi.org/10.1016/0893-6080(90)90049-Q

Soroush, E., Mesbah, M., Shokrollahi, A., Rozyn, J., Lee, M., Kshiwao, T., & Bahadori, A. (2015). Evolving a robust modeling tool for prediction of natural gas hydrate formation conditions, *Journal of Unconventional Oil and Gas Resources*,12,45-55. https://doi.org/10.1016/j.juogr.2015.09.002.

Spandonidis, C., Theodoropoulos, P., Giannopoulos, F., Galiatsatos, N., & Petsa, A. (2022). Evaluation of deep learning approaches for oil & gas pipeline leak detection using wireless sensor networks. *Eng. Appl. Artif. Intell.* 113, C. https://doi.org/10.1016/j.engappai.2022.104890

Tang, H., & White, C. D. (2008). Multivariate statistical log log-facies classification on a shallow marine reservoir.

Journal of Petroleum Science and Engineering, 61(2-4), 88–93. https://doi.org/10.1016/j.petrol.2008.05.004

Tolstaya, E., Shakirov, A., Mezghani, M., & Safonov, S. (2023). Fast reservoir characterization with ai-based lithology prediction using drill cuttings images and noisy labels. *Journal of Imaging*, 9(7), 126. https://doi.org/10.3390/jimaging9070126

Tonni, L., & Simonini, P. (2013). Shear wave velocity as function of cone penetration test measurements in sand and silt mixtures. Engineering Geology, 163, 55-67. https://doi.org/10.1016/j.enggeo.2013.06.005

Todorov, T. I. (2000). Integration of 3C-3Dseismic data and well logs for rock property estimation: M.S. thesis, University of Calgary

van der Spek, A., & Thomas, A. (1999). Neural-net identification of flow regime with band spectra of flow-generated sound. *SPE Reservoir Evaluation & Engineering*, *2*(06), 489-498.

Vázquez-Ayala, J. A., Ortiz-Alemán, J. C., López-Juárez, S., Couder-Castañeda, C., & Trujillo-Alcántara, A. (2025). Prediction of Sonic Well Logs Using Deep Neural Network: Application to Petroleum Reservoir Characterization in Mexico. *Geosciences*, 15(11), 424. https://doi.org/10.3390/geosciences15110424

Wang, X., Wang, P., Zhang, X., Wan, Y., Liu, W., & Shi, H. (2023). Efficient and robust Levenberg–Marquardt Algorithm based on damping parameters for parameter inversion in underground metal target detection. *Computers & Geosciences*, 176, 105354. https://doi.org/10.1016/j.cageo.2023.105354

Yang, Y., Wang, J., Li, Z., Liu, N., Liu, R., Gao, J., & Wei, T. (2023). Automated stratigraphic correlation of well logs using Attention Based Dense Network. *Artificial Intelligence in Geosciences*, 4, 128-136. https://doi.org/10.1016/j.aiig.2023.09.001

Zeng, L., Ren, W., Shan, L., Huo, F., & Meng, F. (2022). Lithology spatial distribution prediction based on recurrent neural network with kriging technology. *Journal of Petroleum Science and Engineering*, *214*, 110538. https://doi.org/10.1016/j.petrol.2022.110538

SAŽETAK

Modeliranje posmičnih valova iz konvencionalnih karotažnih podataka korištenjem integrirane konvolucijske neuronske mreže dubokoga učenja (I-CNN)

Brzina posmičnoga vala, zajedno s brzinom kompresijskoga vala, pruža ključni izvor informacija za geomehaničke i geofizičke istraživanja. Dok se brzina kompresijskoga vala rutinski mjeri u većini bušotina, podatci o brzini posmičnoga vala često nisu dostupni, posebno u starijim bušotinama. To je uglavnom posljedica nedostatka odgovarajućih tehnologija prikupljanja podataka u to vrijeme i ograničenja trenutnih alata u zacijevljenim bušotinama. Uz to, izravno mjerenje brzine posmičnoga vala ostaje relativno skupo. Kako bi se zaobišla ta ograničenja, za predviđanje brzine posmičnoga vala iz tradicionalnih karotažnih mjerenja stvoreni su sljedeći algoritmi strojnoga učenja: vjerojatna neuronska mreža (PNN), višeslojna neuronska mreža s unaprijednim povratnim informacijama (MLFFNN), duboka neuronska mreža s unaprijednim povratnim informacijama (DFFNN), jednodimenzionalna konvolucijska neuronska mreža (1D-CNN) i integrirana konvolucijska neuronska mreža (I-CNN). Iz dviju bušotina prikupljen je 19 121 podatak. Metodologija se sastojala od triju glavnih dijelova: 1) predobrada podataka kako bi se uklonila buka i prilagodio odgovarajući format; 2) korištenje stručnoga znanja za pokretanje inženjeringa i odabira značajki; i 3) treniranje, testiranje i optimizacija modela. Rezultati su pokazali da je I-CNN model u bušotini RCW-1 postigao najbolje performanse, s R-kvadratnom vrijednošću od 0,971. Kada je primijenjen na drugu bušotinu (RCW-2), I-CNN model zadržao je snažnu sposobnost generalizacije, postižući prosječni koeficijent determinacije od 0,956. Ova zapažanja upućuju na to da I-CNN nadmašuje druge metode u rukovanju složenim, nelinearnim odnosima u predviđanju brzine posmičnih valova. Sveukupno, ova studija doprinosi rastućem opusu literature o primjeni strojnoga učenja u petrofizičkoj analizi uvođenjem integriranoga okvira dubokoga učenja koji nadilazi tradicionalne pristupe. Korištenje slijepe bušotine za testiranje dodatno potvrđuje robusnost i pouzdanost predloženoga I-CNN modela.

Ključne riječi:
posmični val, strojno učenje, neuronska mreža, integrirani CNN

Author's contribution

Rahmat Catur Wibowo: conceptualization, methodology, software, data curation, formal analysis, investigation, writing - original draft, visualization. **Ida Bagus Suananda Yogi**: data curation, writing - review & editing. **Indra Arifianto**: validation, data curation, formal analysis, writing - review & editing. **Muh Sarkowi**: methodology, supervision, project administration, resources, writing - review & editing.
All authors have read and agreed to the published version of the manuscript.