Modeling natural convection in porous media using convolutional neural networks

Mohammad Reza Hajizadeh Javaran[1], Amadou-oury Bah[2], Mohammad Mahdi Rajabi[1], Gabriel Frey[3], Florence Le Ber[3], Marwan Fahs[2]

[1]Civil and Environmental Engineering Faculty, Tarbiat Modares University, PO Box 14115-397, Tehran, Iran.

[2]Université de Strasbourg, CNRS, ENGEES, ITES UMR 7063, 67000 Strasbourg, France.

[3]Université de Strasbourg, CNRS, ENGEES, ICube UMR 7357, 67000, Strasbourg, France

mrhj75@gmail.com, amadou-oury.bah@etu.unistra.fr, mmrajabi@modares.ac.ir, g.frey@unistra.fr, florence.leber@engees.unistra.fr, fahs@unistra.fr

Abstract. Deep learning has become increasingly prevalent in a wide range of engineering contexts. In this work, we tried to make a connection between the groundwater engineering community and the field of deep learning. Natural convection in porous media is usually simulated using common numerical modeling tools with high computational costs. In this work, we aim to use supervised learning in input-output pairs (porous media characteristics-heat map distribution) in an image regression task, employing an encoder-decoder convolutional neural network (ED-CNN) to develop a meta-model that is able to predict the distribution of heat map resulting from a natural convection process in porous media or to estimate the characteristics of the porous domain when the heat map distribution is given. In order to achieve this objective, a training data set of samples is prepared using Comsol Multiphysics numerical modeling and is trained with the proposed encoder-decoder CNN. We also employed several evaluation metrics such as root mean squared error (RMSE), coefficient of determination (R^2-score) to assess the robustness of the developed network. We observed promising results in both approaches, as well as accuracy and speed, indicating the network's relevance in a variety of groundwater engineering applications to come in the future.

Keywords: natural convection; porous media; convolutional neural network; encoder-decoder.

1 Introduction

Natural convection is an important concept in porous media problems [1]. It is encountered in several applications such as in heat storage in aquifers, CO_2 sequestration in geological formations, geothermal energy extraction, and geological deposition of nuclear waste. Physics-based numerical models are commonly used for simulating natural convection in porous media. Despite the effectiveness of these models in most cases, they encounter some critical challenges. One key challenge is the computational time cost, which is more noticeable at large time and space scales, especially in repetitive runs. In recent years, several meta-models, such as polynomial chaos expansions and feed-forward neural networks, have been proposed in order to reduce the simulation time of natural convection models. These meta-models have demonstrated acceptable performance in low-dimensional domains, but they do not scale well to high-dimensional problems [2]. To overcome this challenge, we propose the use of a convolutional neural network (CNN) architecture [3]. We apply the proposed ED–CNN in the context of '*image-to-image* regression to (a) estimate the entire heat distribution resulting from a specified permeability or (b) estimate the permeability from a heat map.

2 Methodology

We first develop a numerical model based upon a hypothetical square porous media example, generating heat map distribution images as training data. Each image references a unique value of a porous domain characteristic, known as the Rayleigh number. The generated data are then trained and validated using an encoder-decoder CNN, and results are analyzed using various methods.

2.1 Example description and governing equations

A hypothetical, two-dimensional saturated square porous media is considered. As demonstrated in **Fig. 1**, Dirichlet temperature boundary conditions are assigned to the side walls. TL and TR have constant values and $TL > TR$. We also consider Neumann boundary conditions for the bottom and the upper boundaries, which emphasizes impermeable and thermally adiabatic conditions. The flow is assumed to be steady-state with a Newtonian and incompressible fluid following Darcy's law. The test case is a homogeneous media, with equal hydraulic and thermal properties considered as Rayleigh number (Ra). The natural convection in porous media is explained by the heat transfer equation showing the energy balance, the continuity equation for mass balance, coupled with a variable fluid density function. The governing equations are [4]:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{1}$$

$$u = -\frac{\partial_p}{\partial_x} \tag{2}$$

$$v = -\frac{\partial p}{\partial y} + Ra.T \tag{3}$$

$$Ra = \frac{k.\rho_c.\beta.g.\Delta T.H}{\mu.\alpha} \tag{4}$$

$$u\frac{\partial T}{\partial x} + v\frac{\partial T}{\partial y} = \frac{\partial^2 T}{\partial^2 x} + \frac{\partial^2 T}{\partial^2 y} \tag{5}$$

Where $u(\frac{m}{s})$ and $v(\frac{m}{s})$ are velocity in the x and y directions, respectively, p is the pressure, and T is temperature. The dimensionless Rayleigh value is defined while k (m^2) is hydraulic conductivity, ρ_c (kg/m^3) is the fluid density, $\beta(1/K)$ is the fluid thermal expansion, $g(m/s^2)$ is the acceleration due to gravity, ΔT (K) is the temperature gradient between the left and right walls (i.e., $TL - TR$), H (m) is the size of the domain, μ $(kg/m.s)$ is the fluid viscosity, and α (m^2/s) is the medium equivalent thermal diffusivity coefficient.

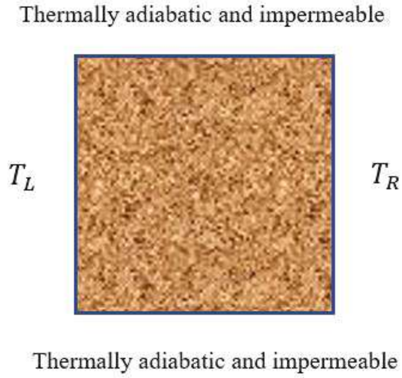


Fig.1. Schematic of the problem domain

2.2 Training Data preparation

In order to train the CNN models, we generated data using a COMSOL Multiphysics model, solving the above-mentioned equations, which takes about 600 minutes to reach a steady state. Using uniform probability distribution, sampling is done using Latin hypercube sampling, and independent Rayleigh numbers are chosen on the interval [10, 210] to generate 2000 heat map images. To train an image-to-image regression model, we converted the Rayleigh value numbers to 32×32 images using Numpy and Matplotlib packages, each representing a specific Rayleigh value pertaining to a heat map image and pixel values of images are normalized between 0 and 1 in the preprocessing step of neural network training. All pixels of Rayleigh images have the same values for each image due to homogeneity; this is because we are developing a methodology, and though it might seem counterintuitive, we are using a homogeneous case as a first step. The input-output pairs are used to train an encoder-decoder CNN.

2.3 Encoder-Decoder CNN

We employ an encoder-decoder architecture for this problem, consisting of two separate subnetworks; encoder is a subnetwork that extracts features through a contracting process, followed by a decoder, which reconstructs the image [5],[6]. Decoder CNNs usually have the same network architecture as encoders, except that they are oriented in the opposite direction [7]. They recover the spatial resolution lost at the encoder by deconvolution and up-sampling and construct output maps based on the feature maps from the encoder [8],[9]. After data preparation, we trained the model with a maximum number of 2,000 samples, where 50% are used for training, 30% for validation, and 20% for testing. We developed two ED-CNNs, one as a meta-model and the other as an optimizer. The meta-model is trying to estimate the heatmap distribution as an output while the input Rayleigh parameter images are fed to the model. Furthermore, a similar methodology has also been employed to develop a model that acts as an *optimizer* to estimate the Rayleigh number from the heat distribution. The ED-CNN models have been developed using Keras and Tensorflow python machine learning libraries. **Fig.2** shows our proposed ED-CNN [2], which is constructed using convolutional layers, each of which is followed by a batch normalization layer, which regularizes the network while enhancing the accuracy [10] Two times, down-sampling and rebuilding is done using two pooling layers and two upsampling layers, respectively in the middle of the network. Furthermore, the activation function is rectified linear unit (Relu), but the sigmoid function is also used in some layers, and the loss function is mean squared error. The model is trained with 300 epochs using batch size 24 and the learning rate of 0.0001 with Adam optimizer.

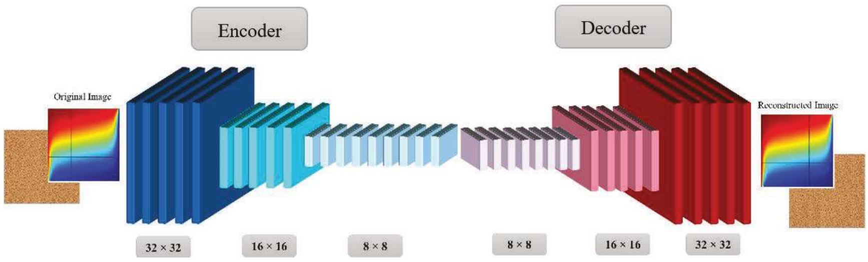


Fig 2. The architecture of the proposed encoder-decoder CNN

3 Result and discussion

3.1 CNN as meta-model

Different numbers of training input-output images (including 100, 500, 1000, and 2000) generated from the numerical model are employed to train the proposed networks. Two evaluation criteria are used to assess the performance of the developed ED-CNN modes: (1) the root mean squared error (RMSE) [11], and (2) the coefficient of determination (R^2-score), a number that shows a good prediction as it gets closer to 1 [12].

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(y_{pred} - y_{true}\right)^2} \tag{6}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}\left(y_{pred} - y_{true}\right)^2}{\sum_{i=1}^{N}\left(y_{pred} - y_{mean\ true\ values}\right)^2} \tag{7}$$

Fig3.a illustrates RMSE decay with different numbers of sample sets. It is apparent from the plot that training the network with about 60 epochs could be enough to reach a stable value of errors. Increasing the number of samples to 2000, the RMSE converges to an acceptable value of 0.0186. **Fig3.b** shows R^2-score changes with the number of samples. As it is apparent from the plot, increasing the number of samples from 100 to 2000 samples slightly improves the accuracy, which is more than 0.97, conforming the RMSE plot results. As an example of the results, the performance of ED-CNN used as the meta-model is visualized for a specific value of the Rayleigh number in **Fig4** using different numbers of sample sets to assess the effect of the number of samples. In this figure, we compare the CNN's predicted heat map with the numerical modeling result, which shows a prediction with a decreasing error as we increase the samples from 100 to 2000. Using only 100 images shows a noticeable spatial error with a total error of

0.05, but increasing samples to 2000 decreases the total error to about 0.01. The spatial distribution of the error, that is, the absolute value of the difference between CNN and numerical model predictions of temperature, is calculated pixel by pixel. In the meta-model case, we can see that in the middle of the domain, errors are more prominent.

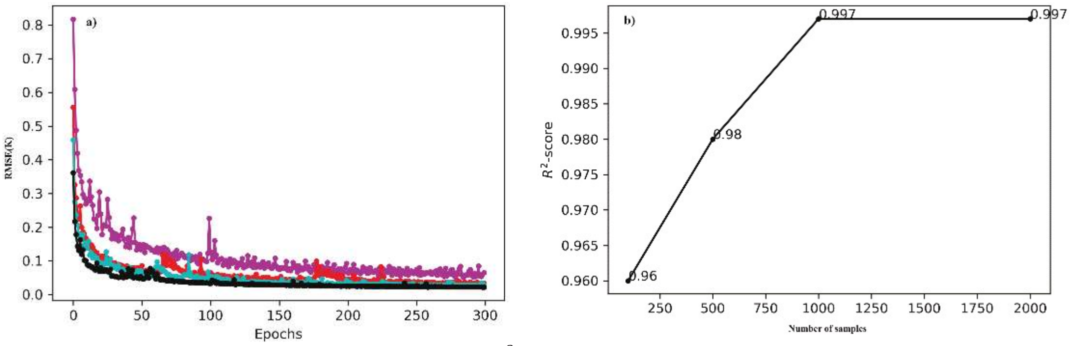
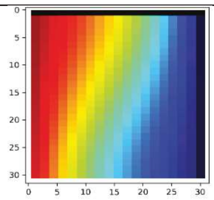
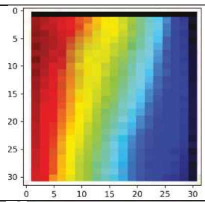
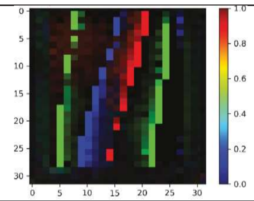
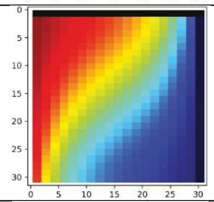
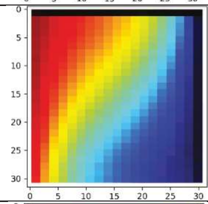
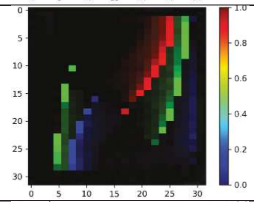
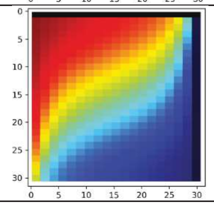
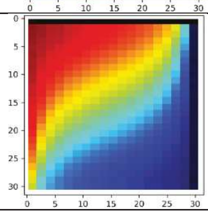
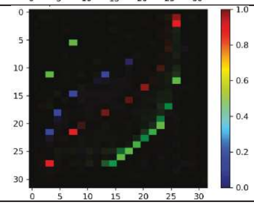


Fig 3. a) RMSE(K) and b) R^2-score ED-for CNN as meta model

Number of samples	Real HeatMap	Predicted HeatMap	Spatial distribution of the error
100			
500			
1000			

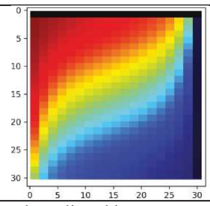
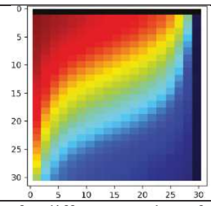
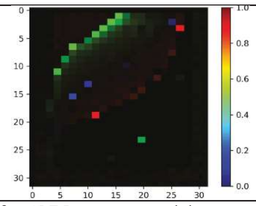
2000			

Fig 4. Real and predicted heat map comparison for different number of samples for CNN as meta-model

3.2 CNN as an optimizer

To evaluate the proposed network for parameter estimation, the same performance metric, RMSE and R^2-score are employed. **Fig5 a.** illustrates the RMSE decay with the number of epochs. It can be inferred from the plot that increasing the number of samples from 100 to 500 significantly decreases the error while using more than 500 does not affect the RMSE noticeably; this fact is also approved by **Fig5. b**, while using 500 samples instead of 100, enhances the R^2-score from 0.59 to a value of more than 0.98. The other assessment method is shown in **Fig. 6** as a scatter plot, comparing the predicted and actual values using a maximum training data of 2000. As it is apparent from the plot, the majority of predicted and real cases cluster around the 45° line, approving the network's effectiveness. Furthermore, an exemplary table of random predicted and test values also confirms the robustness of the network showing a deficient relative error.

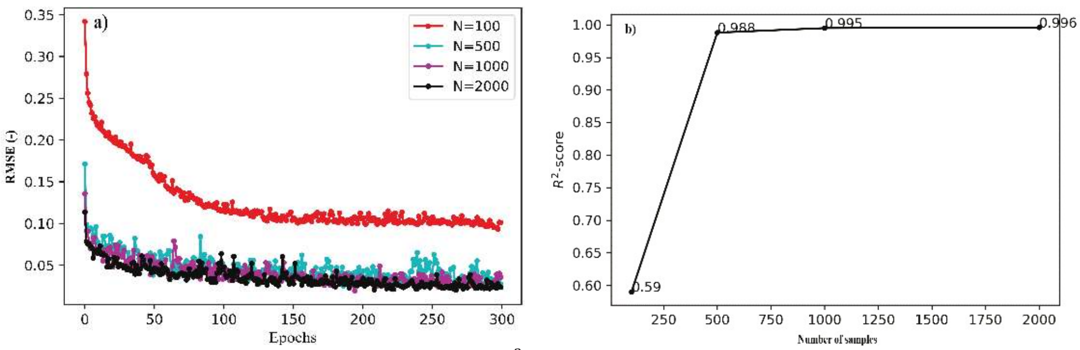


Fig5. a) RMSE(-) and b) R^2-score for ED-CNN as optimizer

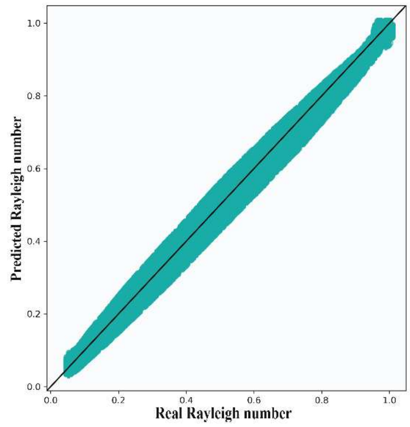


Fig 6. Real and predicted Rayleigh value scatter plot

Table 2. Relative error of real and predicted Rayleigh value for random test cases

Real Rayleigh number	Predicted Rayleigh number	Relative error (%)
49.11	50.16	2.13
56.25	57.30	1.86
60.0	60.8	1.33
89.4	89.8	0.44
117.5	117.7	0.17
192.68	190.16	1.30

4. Conclusion

In this paper, we have developed an encoder-decoder CNN to achieve a meta-model and optimizer for natural convection problems in porous media, considering the time cost and accuracy of the model. We initially generated 2000 heat map images; The data is then trained using similar architectures as meta-model and optimizers. It is apparent from the accurate results that the proposed methodology can be employed as a tool for estimating natural convection heat distribution as a meta-model and estimating the properties of the porous cavity in inverse modeling. It is also observed that this network is trained in about 40 minutes while the numerical modeling process takes more than 600 minutes, which means it saves time, more than 93% compared to numerical modeling tools, showing its robustness in solving the time cost problem. In summary, this network can be used as a meta-model and optimizer and should be also useful for uncertainty analysis.

References

1. Nield, D. A., & Bejan, A. (2017). *Convection in Porous Media*. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-49562-0
2. Rajabi, M. M., Hajizadeh Javaran, M. R., Bah, A. O., Frey, G., Le Ber, F., Lehmann, F., & Fahs, M. (2021). Analyzing the Efficiency and Robustness of Deep Convolutional Neural Networks for Modeling Natural Convection in Heterogeneous Porous Media, submitted to *International Journal of Heat and Mass Transfer*.
3. Ji, X., Yan, Q., Huang, D., Wu, B., Xu, X., Zhang, A., ... & Wu, M. (2021). Filtered selective search and evenly distributed convolutional neural networks for casting defects recognition. Journal of Materials Processing Technology, 292, 117064.
4. Fajraoui, N., Fahs, M., Younes, A., & Sudret, B. (2017). Analyzing natural convection in porous enclosure with polynomial chaos expansions: Effect of thermal dispersion, anisotropic permeability and heterogeneity. *International Journal of Heat and Mass Transfer*, *115*, 205-224.
5. Mo, S., Zabaras, N., Shi, X., & Wu, J. (2019a). Deep autoregressive neural networks for high-dimensional inverse problems in groundwater contaminant source identification Water Resources Research, 55(5), 3856-3881.
6. Selvin, S., Vinayakumar, R., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2017, September). Stock price prediction using LSTM, RNN and CNN-sliding window model. In 2017 international conference on advances in computing, communications and informatics (icacci) (pp. 1643-1647). IEEE
7. Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham.
8. Jiang, Z., Tahmasebi, P., & Mao, Z. (2021). Deep residual U-net convolution neural networks with autoregressive strategy for fluid flow predictions in large-scale geosystems. Advances in Water Resources, 150, 103878.
9. Tahmasebi, P., Kamrava, S., Bai, T., & Sahimi, M. (2020). Machine Learning in Geo-and Environmental Sciences: From Small to Large Scale. *Advances in Water Resources*, 103619
10. Zhu, Y., & Zabaras, N. (2018). Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification. Journal of Computational Physics, 366,415-447.
11. Zhu, Y., Zabaras, N., Koutsourelakis, P. S., & Perdikaris, P. (2019). Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data. *Journal of Computational Physics*, *394*, 56-81.
12. Kumar, D., Roshni, T., Singh, A., Jha, M. K., & Samui, P. (2020). Predicting groundwater depth fluctuations using deep learning, extreme learning machine and Gaussian process: a comparative study. Earth Science Informatics, 13(4), 1237-1250.