Fully automatic Bayesian method for trainable activation function and deep neural networks

Mohamed Fakhfakh, Lotfi Chaari

Fully automatic Bayesian method for trainable activation function and deep neural networks

Mohamed Fakhfakh[1] and Lotfi Chaari[2]

[1] Toulouse INP, IRIT, University of Toulouse, France
mohamed.fakhfakh@toulouse-inp.fr

[2] Toulouse INP, IRIT, University of Toulouse, France
lotfi.chaari@toulouse-inp.fr

***Abstract*—In this study, we introduce a novel technique that incorporates an MCMC-based algorithm within a Bayesian framework to estimate the parameters of trainable activation functions and model weights in deep neural networks. This approach aims to enhance network performance by effectively tackling the challenges of parameter learning and reducing the risk of overfitting. Our method leverages an efficient sampling process to accelerate convergence times. The efficacy of the proposed activation function is demonstrated through experiments on two datasets within the remote sensing domain. These experiments reveal that our approach enables neural networks to achieve high levels of accuracy, reaching up to 92%, while keeping the model complexity low. This suggests that the proposed method could offer significant benefits for deep learning applications, particularly in fields requiring precise and reliable predictive modeling.**

***Index Terms*—** M CMC, ns-HMC, Activation function, Deep neural networks, Optimization

I. Introduction

The field of remote sensing stands at the forefront of machine learning advancements, serving as a critical arena for the application of Convolutional Neural Networks (CNNs) [1–3]. These networks have profoundly transformed our capability to analyze and interpret the intricate details of high-dimensional satellite imagery. Such analyses are indispensable for a myriad of applications, including but not limited to environmental monitoring, urban planning, and agricultural management. The remarkable success of CNNs in these areas underscores the importance of sophisticated computational techniques in deepening our understanding of and interaction with the Earth's surface.

CNNs have significantly altered machine learning by simplifying complex data into digestible, low-dimensional outputs via hierarchical layers that systematically abstract data for improved pattern recognition. The activation function, a core element of CNNs, provides the necessary non-linearity to capture complex relationships in data. The pursuit of the optimal activation function, from fixed to trainable types, underscores the ongoing evolution in neural network research [4].

Within this context, Bayesian methods, particularly Markov Chain Monte Carlo (MCMC) techniques [5, 6], have become prominent for their robustness in integrating prior knowledge with empirical data, offering a nuanced approach to complex data challenges. These methods facilitate more efficient optimization in neural networks compared to traditional approaches.

This paper contributes to the field by introducing an MCMC-based model for estimating the parameters of a trainable activation function and model weights, extending our previous research on non-smooth Hamiltonian methods for fitting sparse neural networks [7, 8]. Our model enhances sampling efficiency, even with non-differentiable energy functions from sparse regularization.

The document is organized as follows: The problem statement is outlined in the initial section. The hierarchical Bayesian model employed is described in detail (Section III), followed by the development of our proposed Bayesian inference scheme (Section IV), and its empirical validation using datasets in the realm of remote sensing (Section V). The conclusion summarizes our findings and outlines potential directions for future research (Section VI).

II. Problem formulation

Activation functions are pivotal in neural networks, adopting various forms such as conventional ones like the sigmoid [9], hyperbolic tangent (tanh) [10], and ReLU [11], alongside trainable versions such as FReLU [12] and MeLU [13], which adjust their parameters through gradient descent. Additionally, non-traditional approaches like the Maxout network [14] broaden the scope of neural computation beyond standard paradigms.

However, these methods face challenges including extensive computational demands and the issue of vanishing gradients, which may cause models to get trapped in local optima, adversely affecting their performance [15]. The adaptability of these functions and the accuracy of parameter estimation remain unresolved questions.

In our work, we propose a refined version of the MeLU [13] activation function, incorporating its parameter optimization within a comprehensive Bayesian optimization framework. This approach is motivated by MeLU's notable efficiency and its potential for encouraging sparsity. Nonetheless, the primary drawbacks of the MeLU function include its computational intensity and significant memory usage.

To mitigate these issues, we introduce the Modified Mexican ReLU (MMeLU) activation function, designed to simplify the complexity and enhance the performance of models. MMeLU

necessitates a reduced number of parameters for estimation compared to MeLU, and it is optimized through a Bayesian framework known for its precision and rapid convergence [7]. We now proceed to define the MMeLU function. The formula

$$f_{\gamma,b}(x) = max(b - |x - \gamma|, 0). \tag{1}$$

describes the calculation within a neural network layer, where x represents the input, and γ, b are parameters represented by real numbers. The MMeLU activation function, incorporating this calculation, is expressed as:

$$MMeLU(x) = ReLU(x) + c\, f_{\gamma,b}(x). \tag{2}$$

with c being another real number parameter alongside γ and b, all of which are to be optimized.

Building upon our preceding contributions [8], this research introduces an innovative trainable activation function, MMeLU, which along with the network's parameters, is optimized within a Bayesian framework, diverging from conventional methods such as the ADAM optimizer. This novel approach, detailed in our recent work, leverages the non-smooth Hamiltonian Monte Carlo algorithm for the sparse optimization of neural network weights [7, 16], offering applications not only in classification tasks but also in regression scenarios aiming to delineate and predict the dynamics between dependent and independent variables.

In this context, the objective is to determine the weights vector $W \in \mathbb{R}^N$ that minimizes the quadratic error across M input data points during the training phase, as articulated by the optimization problem:

$$\begin{aligned}\widehat{W} &= \arg\min_{W} \mathcal{L}(W) \\ &= \arg\min_{W} \sum_{m=1}^{M} \|MMeLU(x^m; W) - y^{(m)}\|_2^2 + \lambda \|W\|_1,\end{aligned} \tag{3}$$

where λ is a regularization coefficient fine-tuning the trade-off between fitting the model to the data and enforcing sparsity through ℓ_1 regularization.

III. Hierarchical Bayesian Model

This section delves into the Bayesian estimation approach for the parameters of the trainable activation function. Within this Bayesian context, both parameters and hyperparameters are treated as random variables, each adhering to specific probability distributions. A likelihood distribution is devised to encapsulate the relationship between the data, the activation function parameters, and the target weights. Simultaneously, a prior distribution is established to integrate existing knowledge about the weights and parameters of the activation function.

A. Likelihood Formulation

The error minimization between the reference vector y (be it labels or continuous values) and its prediction $\widehat{y}$ is based on the premise that a quadratic loss function implies a Gaussian noise model between the true values and their estimates. Consequently, the likelihood function is formalized as:

$$\begin{aligned}&f(y; W, c, \gamma, b, \sigma^2) \propto \times \\ &\prod_{m=1}^{M} \exp\left(-\frac{1}{2\sigma^2}\|MMeLU(x^m; W) - y^{(m)}\|^2\right).\end{aligned} \tag{4}$$

where σ^2 represents the variance, a parameter to be determined.

B. Prior Distributions

The model encapsulates unknown parameters within the vector θ = {W, c, γ, b}, for which we establish prior distributions.

For the Weight Vector W :

A Laplace distribution is selected to encourage sparsity in the neural network's weights:

$$f(W; \lambda_w) \propto \prod_{k=1}^{N} \exp\left(-\frac{|W^{[k]}|}{\lambda_w}\right), \tag{5}$$

where λ_w is a hyperparameter that controls the distribution's spread. $W^{[k]}$ is the weights vector of the k^{th} layer of the network.

For Parameters c, λ, and b:

Each of these parameters is also governed by a Laplace distribution, promoting values near zero to ensure model simplicity and robustness:

$$f(c; \lambda_c) \propto \exp\left(-\frac{|c|}{\lambda_c}\right). \tag{6}$$

and similarly for γ and b with their respective hyperparameters $f(\gamma; \lambda_\gamma)$ and $f(b; \lambda_b)$. These hyperparameters can be finely tuned or estimated from the data, allowing for a flexible modeling approach.

IV. Bayesian inference scheme

Adopting a Maximum A Posteriori (MAP) framework necessitates articulating the conditional posterior distribution. With the target parameter vector θ = {W, c, γ, b} and hyperparameters vector Φ = { σ^2, λ_c, λ_γ, λ_b, λ_w}, the joint posterior is derived from the established likelihood and priors. For a detailed formulation, refer to [8].

The conditional posteriors for W, c, γ, and b involve expressions that combine data fitting with regularization enforced by the respective hyperparameters. The expressions detail the integration of model outputs with prior beliefs, promoting sparsity and model fidelity. For W, the posterior incorporates an energy function $E_\theta^k(W)$, balancing data fidelity against a regularization term, akin to the formulation in [8].

Our approach utilizes the Metropolis-Hastings algorithm alongside a non-smooth Hamiltonian Monte Carlo (ns-HMC)

method for sampling, developed in [16], which refines traditional techniques for increased efficiency in parameter estimation. This sophisticated sampling framework, grounded in recent advancements, facilitates precise adjustment of model parameters through an iterative process, ensuring convergence to optimal solutions. The iterative sampling process is encapsulated in the Gibbs sampler algorithm, which systematically updates each parameter set until convergence is reached. This process is streamlined as follows: The resulting Gibbs sampler is summarized in Algorithm 1.

Algorithm 1: Main steps of the proposed method.

- Fix the hyperparameters Φ ;
while *not convergence* **do**
  - Sample c according to $f(c; \alpha, \lambda_c)$;
  - Sample γ according to $f(\gamma; \alpha, \lambda_\gamma)$;
  - Sample b according to $f(b; \alpha, \lambda_b)$;
  - Sample W as in [8] ;
end

V. Experimental validation

To assess the effectiveness of the proposed approach, we conducted image classification experiments using two distinct datasets: one comprising satellite imagery for land cover classification and another focusing on Brazilian Coffee Scenes. These datasets were chosen to showcase the method's versatility and robustness across different types of remote sensing data.

To benchmark our method with existing techniques, we employed four well-known activation functions in conjunction with the ADAM optimization algorithm [17], setting the learning rate to 10^{-3}. These activation functions include ReLU [11], FReLU [12], ELU [11], and MeLU [13], allowing for a comprehensive comparison across various models to demonstrate the superiority of our modified approach in terms of performance and efficiency. To perform the classification task, the CNN architecture employed in this study has nine convolutional (3XConv-32, 3XConv-64, and 3XConv-128) and three fully connected (FC-128, FC-64, and FC-softmax). Each convolutional layer includes filters with 3×3 Kernels in addition to 2×2 max-pooling layers, with stride size equal to 1. In addition, two regularisation techniques are used: Batch Normalization and Dropout (the dropout rate is set by cross-validation to $p = 0.35$).

A. Sampling Results :

Following the application of our Bayesian optimization technique for training Convolutional Neural Network (CNN) models for Covid-19 CT image classification, we evaluated the convergence patterns. The graphs displayed illustrate the progression of sampling for the γ, b, and c parameters within the proposed MMeLU activation function, showcasing both the sampling paths and the distribution of samples through histograms. These visual representations (sampling paths for γ, b, and c in panels a-c and their histograms in panels d-f) underscore the efficient convergence and robust mixing achieved by our tailored Gibbs sampling method. Notably, post a preliminary phase of 350 iterations, the algorithm demonstrates consistent convergence stability and an effective mixing rate across the parameter samples.

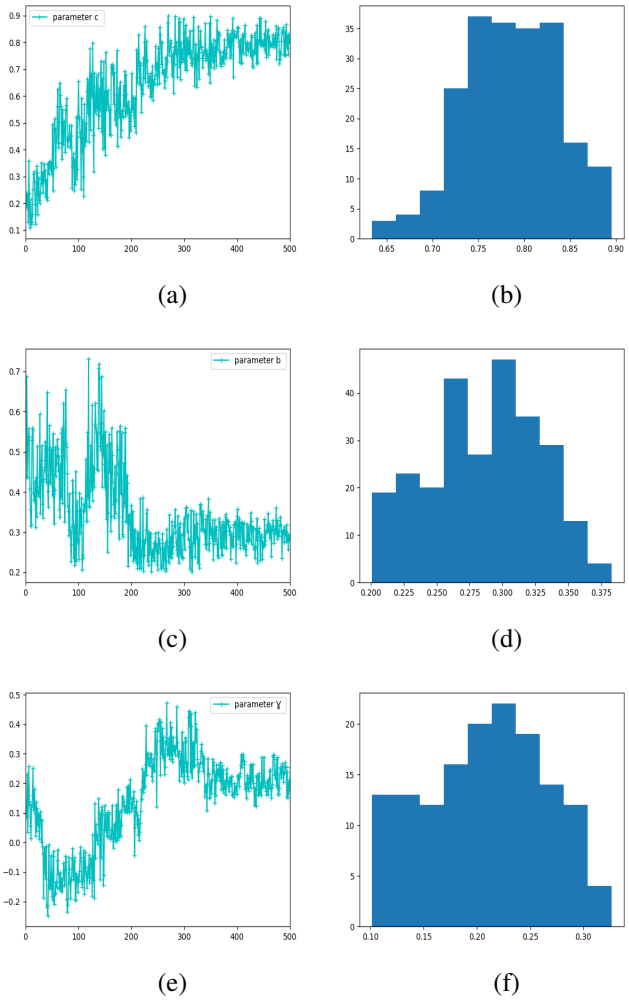


Fig. 1. Sampling of parameters c (a,b), b (c,d), and γ (e,f): chains and histograms.

B. Experiment 1 : Satellite Image Classification

This section assesses the performance of our classification methodology using the RSI-CB256 dataset, which comprises four categories of satellite imagery sourced from various sensors and Google Maps [1]. With 5631 images, the dataset represents a significant challenge for machine learning models. Our goal is to demonstrate the effectiveness of our approach in accurately classifying these images, highlighting its capability to contribute to advancements in remote sensing

[1]https://www.kaggle.com/datasets/mahmoudreda55/satellite-image-classification/data

as well as in various other application areas.
The outcomes in Table I suggest the proposed method exhibits a slight superiority in terms of accuracy when compared to conventional activation functions, albeit the differences are modest. Notably, the proposed method outperforms its competitors significantly in computational efficiency, as indicated by the shorter processing times. While the improvements in loss and accuracy are marginal, the enhanced computational time highlights the method's efficiency. This slight edge in performance metrics, particularly in a task of this complexity, emphasizes the proposed method's capability to balance accuracy with computational demands effectively.

TABLE I
EXPERIMENT 1: RESULTS FOR SATELLITE IMAGE CLASSIFICATION (COMPUTATIONAL TIME IN MIN, ACCURACY, LOSS).

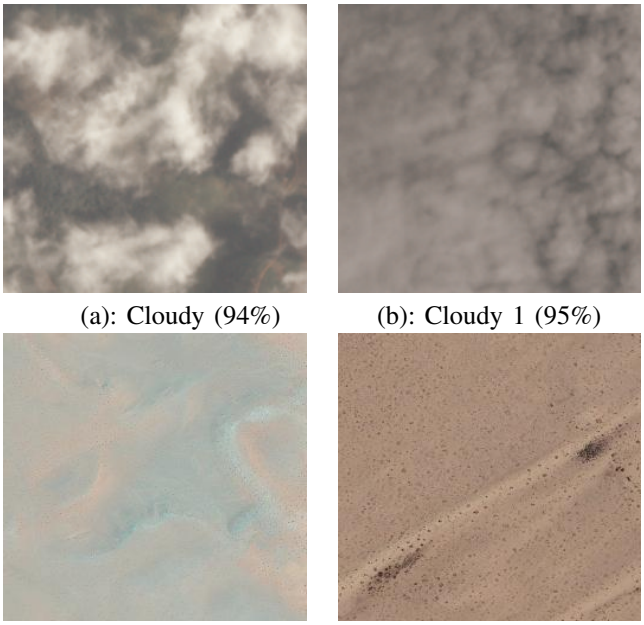
Activation function	Time(min)	Loss.	Acc.
MMeLU	**401**	**0.10**	**0.97**
ReLU	455	0.16	0.95
ELU	485	0.19	0.93
FReLU	475	0.13	0.96
MeLU	508	0.10	0.96

Figure 3 shows examples from each of the four classes along with their detection scores using our MMeLU approach, with probabilities ranging from 94% to 97%. This demonstrates our method's ability to accurately classify satellite images across various categories. It especially shows promise for applications such as agriculture and crop management, where precise image classification is crucial.

C. Experiment 2 : Brazilian Coffee Scenes classification

This section evaluates our classification method on the Brazilian Coffee Scenes Dataset [18], which is comprised of two classes: coffee and non-coffee, containing 2876 images captured by the SPOT sensor in 2005 across four counties in the State of Minas Gerais, Brazil: Arceburgo, Guaranesia, Guaxupé, and Monte Santo. This dataset features multispectral high-resolution scenes of coffee crops and non-coffee areas, presenting significant intraclass variance due to different crop management techniques, as well as scenes with varying plant ages and/or spectral distortions caused by shadows.
Table II suggests that the performance of all competing activation functions on the Brazilian Coffee Scenes classification dataset was suboptimal. However, MMeLU emerged as the superior option, achieving an accuracy of up to 92%. Furthermore, our proposed approach also demonstrated reduced computational time relative to other methods, reinforcing its effective performance.

Figure 3 displays three images for each category—Coffee and Non-Coffee—complete with their detection scores. In this experiment, our MMeLU approach exhibits good performance, with all probability scores exceeding 95%. These results not only underscore the method's robust



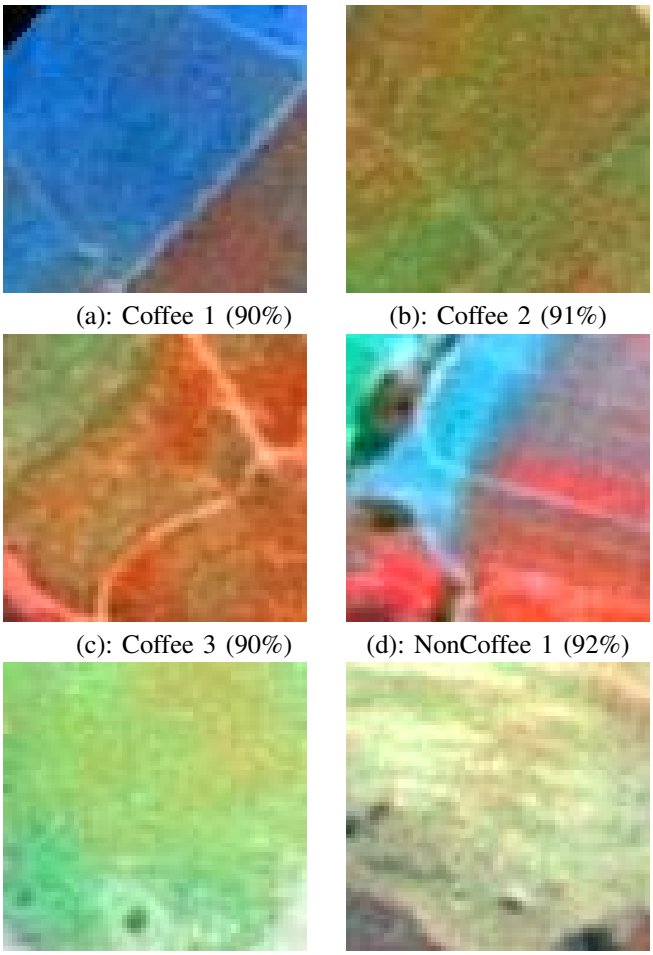
(a): Cloudy (94%) (b): Cloudy 1 (95%)
(c): Desert (96%) (d): Desert 1 (94%)
(c): Green area (96%) (d): Green area 1 (94%)
(c): water (96%) (d): Water 1 (94%)

Fig. 2. Experiment 1: Two Examples of Images for Each Class with Their Reported Detection Scores.

TABLE II
EXPERIMENT 2: RESULTS FOR BRAZILIAN COFFEE SCENES CLASSIFICATION (COMPUTATIONAL TIME IN MIN, ACCURACY, LOSS).

Activation function	Time(min)	Loss.	Acc.
MMeLU	**157**	**0.20**	**0.92**
ReLU	185	0.35	0.83
ELU	193	0.47	0.79
FReLU	206	0.30	0.85
MeLU	229	0.29	0.87

capability in accurately distinguishing between coffee and non-coffee satellite images but also demonstrate its potential for enhancing precision in agricultural monitoring and land use classification.



(a): Coffee 1 (90%) (b): Coffee 2 (91%)

(c): Coffee 3 (90%) (d): NonCoffee 1 (92%)

(e): NonCoffee 2 (90%) (f): NonCoffee 3 (90%)

Fig. 3. Experiment 2: Three Examples of Images for Each of the Two Classes, Coffee and Non-Coffee, with Their Reported Detection Scores.

VI. Conclusion

This study introduces a Bayesian approach designed for sparse deep neural networks, incorporating trainable activation functions through the application of Hamiltonian dynamics and non-smooth regularization techniques. The method achieves notable classification accuracy, superior generalization capabilities, and reduced computational time in comparison to traditional models employing different activation functions and standard optimization methods.

Looking ahead, our research will aim to enhance the proposed algorithm by enabling parallel processing and GPU support, which is expected to further reduce computational times.

References

[1] M. Fakhfakh, B. Bouaziz, F. Gargouri, and L. Chaari, "Prognet: Covid-19 prognosis using recurrent and convolutional neural networks," *The Open Medical Imaging Journal*, vol. 12, no. 1, 2020.

[2] T. K. Sajja and H. K. Kalluri, "Image classification using regularized convolutional neural network design with dimensionality reduction modules: Rcnn–drm," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–12, 2021.

[3] Y. Li, H. Zhang, X. Xue, Y. Jiang, and Q. Shen, "Deep learning for remote sensing image classification: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 6, pp. 1264, 2018.

[4] A. Apicella, F. Donnarumma, F. Isgrò, and R. Prevete, "A survey on modern trainable activation functions," *Neural Networks*, vol. 138, pp. 14–32, 2021.

[5] C. Andrieu, A. Doucet, and R. Holenstein, "Particle markov chain monte carlo methods," *Journal of the Royal Statistical Society: Series B*, vol. 72, no. 3, pp. 269–342, 2010.

[6] L. Chaari, H. Batatia, N. Dobigeon, and J.-Y. Tourneret, "A hierarchical sparsity-smoothness bayesian model for l0+l1+l2 regularization," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 1901–1905.

[7] M. Fakhfakh, B. Bouaziz, F. Gargouri, and L. Chaari, "Bayesian optimization using hamiltonian dynamics for sparse artificial neural networks," in *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2022, pp. 1–4.

[8] M. Fakhfakh, L. Chaari, B. Bouaziz, and F. Gargouri, "Non-smooth bayesian learning for artificial neural networks," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–24, 2022.

[9] A. C Marreiros, J. Daunizeau, S. J. Kiebel, and K. J. Friston, "Population dynamics: variance and the sigmoid activation function," *Neuroimage*, vol. 42, no. 1, pp. 147–157, 2008.

[10] F.-C. Chen, "Back-propagation neural networks for nonlinear self-tuning adaptive control," *IEEE control systems Magazine*, vol. 10, no. 3, pp. 44–48, 1990.

[11] S. Sun, Z. Cao, H. Zhu, and J. Zhao, "A survey of optimization methods from a machine learning perspective," *IEEE transactions on cybernetics*, vol. 50, no. 8, pp. 3668–3681, 2019.

[12] S. Qiu, X. Xu, and B. Cai, "Frelu: flexible rectified linear units for improving convolutional neural networks," in *2018 24th international conference on pattern recognition (icpr)*. IEEE, 2018, pp. 1223–1228.

[13] G. Maguolo, L. Nanni, and S. Ghidoni, "Ensemble of convolutional neural networks trained with different activation functions," *Expert Systems with Applications*, vol. 166, pp. 114048, 2021.

[14] I. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, "Maxout networks," in *International conference on machine learning*, 2013, pp. 1319–1327.

[15] X. Wang, Y. Qin, Y. Wang, S. Xiang, and H. Chen, "Reltanh: An activation function with vanishing gradient resistance for sae-based dnns and its application to rotating machinery fault diagnosis," *Neurocomputing*, vol. 363, pp. 88–98, 2019.

[16] L. Chaari, J.-Y. Tourneret, C. Chaux, and H. Batatia, "A Hamiltonian Monte Carlo method for non-smooth energy sampling," *IEEE Trans. on Signal Process.*, vol. 64, no. 21, pp. 5585 – 5594, Jun. 2016.

[17] D.P. Kingma and J. Ba, "Adam: a method for stochastic optimization 3rd int," in *Conf. for Learning Representations, San*, 2014, pp. 1–15.

[18] Huizhen Zhao, Fuxian Liu, Han Zhang, and Zhibing Liang, "Convolutional neural network based heterogeneous transfer learning for remote-sensing scene classification," *International Journal of Remote Sensing*, vol. 40, no. 22, pp. 8506–8527, 2019.