




Scaling Deep-Learning Pneumonia Detection Inference on a Reconfigurable Self-Contained Hardware Platform

Jiangkun Wang , Khanh N. Dang , Abderazek Ben Abdallah 

Student Member, IEEE, Member, IEEE, Senior Member, IEEE

Graduate School of Computer Science and Engineering

The University of Aizu

Aizu-Wakamatsu 965-8580, Fukushima, Japan

d8222108,khanh,benab@u-aizu.ac.jp

Abstract—Artificial Intelligence (AI) has been used in applications to alleviate specific problems in academia and industry. For instance, in healthcare, where edge-based computing platforms are heavily used, when it comes to latency and security issues, the increased demands of application of AI applications such as deep learning require a specific platform to meet the latency, security, and power consumption challenges. This work presents methods and architectures for scaling deep learning inference for pneumonia detection in chest X-ray images based on a reconfigurable self-contained hardware platform named AIRBiS¹. The performance evaluation results show that the proposed approach achieves 95.2% detection accuracy of pneumonia over the collected test data with the computer-aided diagnosis scenario. The secure collaborative-learning approach achieves comparable accuracy to the conventional training scenario. However, for rapid batch detection, the detection could be accelerated by 0.023s. Moreover, the system inference acceleration is 13 times (on average) more energy-efficient than conventional approaches.

Index Terms—Scaling Deep-Learning; Pneumonia Detection Inference; Reconfigurable; Self-Contained Platform

I. INTRODUCTION

For many years, Artificial Intelligence (AI) has been used in applications to alleviate specific problems in academia and industry [1]. For example, in healthcare, Deep Learning (DL) models are used for applications such as timely detection of anomalies in patient health monitoring [2], lung ultrasonography classification [3], and recently in the ongoing efforts to combat the COVID-19 pandemic [4]. In order to meet the computing requirement of such applications, a common approach is to deploy DL models on cloud computing platforms [5]. However, some of these applications require secure real-time analysis of generated medical data [6, 7], making cloud computing platforms facing security [8] and latency [9] challenges unsuitable for use. Edge computing, on the other hand, provides a viable approach to meet the low-latency [10], privacy-preserving [9], and security [11] requirements of such applications [12].

As of April 2023, the number of coronavirus cases reported globally exceeds 687 million, with over 6 million deaths [13].

¹AIRBiS project, the University of Aizu: <https://www.u-aizu.ac.jp/misc/benablab/airbis.html>

One major key in fighting the COVID-19 pandemic is detecting and isolating infected patients. To achieve this, an efficient rapid diagnosis method for testing is required.

The standard method to detect COVID-19 is the Reverse Transcription Polymerase Chain Reaction (RT-PCR), which tests for genetic materials of the SARS-CoV-2 virus in the upper respiratory specimens collected from patients. The sensitivity of this test ranges from 60% to 97% [14]. However, there are significant differences among different patients, resulting in substantial false-negative results, as shown by the reduction in detection sensitivity from 60% to 71% [15]. Although RT-PCR can be tested in batches, the samples to be tested must be manually collected from test subjects. Another method is to analyze lung X-ray images of patients, which has accuracy in the range of 80% to 90% [16]. This approach requires doctors to examine the lung X-ray images of a patient one at a time and combine the result with the patient's physical condition to complete a diagnosis. However, with the increase in COVID-19 cases, this approach has become ineffective due to a lack of quick response and reporting. Moreover, privacy and security are essential requirements during patient treatment.

To address these issues, computer-aided diagnosis systems that leverage DNNs have been considered as a potential solution [17, 18]. However, medical institutions generally need more resources to afford large-scale computer-aided diagnosis systems. In addition, traditional biomedical information security measures conflict with distributed learning mechanisms, so it is difficult to aggregate diagnostic models of multiple medical institutions to improve detection accuracy and speed. At the same time, it is also challenging to implement fast test results reporting, quick response mechanisms, and streamlined collaboration with the government. Therefore, clinicians and researchers made great efforts to find other alternative and complementary methods to improve the detection accuracy of COVID-19.

To solve these problems, this work extends our previous work AIRBiS [19, 20, 21, 22] and presents a high-performance and low-power forward-thinking hardware-software co-design system for pneumonia (i.e., COVID-19) detection. The rest

of this paper is organized as follows. Section II describes the proposed system. Section III provides a comprehensive evaluation and results of the proposed system, and in section IV, we present the conclusion and future work.

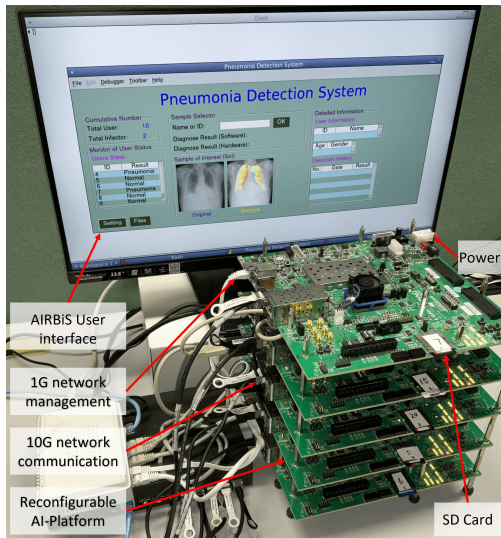


Fig. 1. AIRBiS System Platform in a Real Deployment Demonstration.

II. SYSTEM ARCHITECTURE

Diagnosing in the conventional system requires patients to visit a hospital, line up for an X-ray scan and wait for physician diagnosis results. The drawback is that the hospital will be congested, physicians will be exhausted, and the result from the test may not get in time for the final decision. Our proposed system overcomes these challenges. A patient's lung X-ray image is captured in digital format and sent to the physician within the same hospital or another for diagnosis; however, this process could be automated. Our approach further proposed that the X-ray images be uploaded to the AIRBiS system connected locally for detection and diagnosis. The results are returned immediately by running the inference onsite, and physicians are only informed if their intervention is needed.

The central computing unit in the AIRBiS system is a reconfigurable AI platform that handles the detection and diagnosis tasks as shown in Figure 1. The platform is a five Zynq Ultra Scale+ MPSoc ZCU102 FPGA boards cluster having reconfigurable nodes connected by Ethernet, enabling scalability. With a secure Ethernet connection, remote nodes can form larger clusters to support higher computing requirements. When an active node becomes faulty, other reconfigurable nodes could continue the work of the faulty node. Using the user interface shown in Figure 1, physicians can get the statistical result of infection cases and view the diagnosis results of all users and the sample of interest (SOI).

A. Neural Network Model and Structure

The proposed neural network detection model is CNN based and is described in Figure 3. The X-ray images are

fed to the CNN for feature extraction and the final layer outputs probabilities for each class. The input is X-ray images which are either in grayscale or RGB. The CNN model also comprises three convolution layers with a kernel size of 3×3 , 32 biases, and *ReLU* activation function. A Max pooling layer is used after every convolution layer; two fully connected layers are used after the flatten operation, and the final layer outputs probabilities that are assigned to two classes (Infected and Non-infected). Two dropout layers lie between the max-pooling and fully connected layers to suppress the network from over-fitting. Dropout in the neural network improves the performance by randomly disabling some neurons' activation in each layer during training. This method reduces loss due to over-training during learning. At the output layer, the *softmax* activation function is used to estimate the probability of output classes [19]. Finally, the loss function cross-entropy is described in Equation 1:

$$E(w_o, b_o) = \frac{1}{N} \sum_{i=1}^N y_i \times \log(y'_i(w_o, b_o)) \quad (1)$$

Where w_o and b_o are network parameters, y , and y' denote the real labels and predicted labels, respectively. By minimizing the loss function using stochastic gradient descent and back-propagation learning algorithm, the procedures of optimizing the parameters and making classification decisions can be performed. A summary of the CNN model with input of 256×256 (px) grayscale images is described in Figure 3.

TABLE I
DATASET DESCRIPTION.

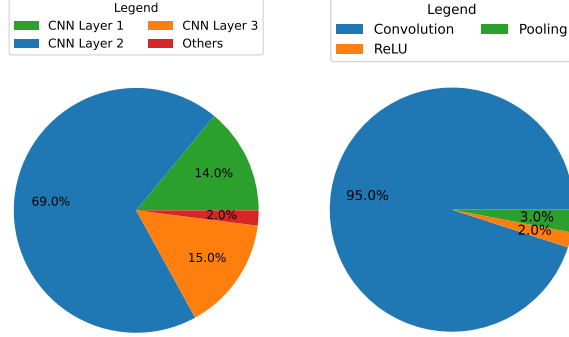
Label	Class	Train	Test
COVID	COVID	2870	701
	COVID (augmented)	14349	-
Non_COVID	Normal	9791	400
	Lung_Opacity	5762	250
	Viral_Pneumonia	1288	49
Sum		34060	1400

To train the neural network model, the lung X-ray image dataset was used [23, 24] from *Kaggle*, which is a benchmark dataset used in comparing the performance of deep learning models on X-ray image classification tasks. Table I describes the specific number of data used for the experiment. The dataset includes 21,111 original X-ray images divided into two labels; COVID-19 (3,571) and Non-COVID. Non-COVID consists of three types of image data, Lung_Opacity (6,012), Viral_Pneumonia (1,337), and normal (10,191). We used a collaborative learning approach to enhance diagnostic accuracy by utilizing X-ray images from multiple medical institutions while safeguarding patient privacy. This approach permits hospitals to share trained models without disclosing actual datasets, thereby preserving patients' data privacy.

B. Inference Acceleration

For large-size datasets, GPUs can be used to improve the inference time of neural networks where model training is

usually performed in high-performance cloud servers. However, GPU-based systems cannot obtain good energy efficiency due to high power consumption. A configurable AI chip system can save bandwidth and reduce the application's latency while requiring less power than GPUs. Our approach uses an FPGA-based cluster to scale the inference and decrease power consumption.



(a) The profiling results of CNN Layer1, CNN Layer2, CNN Layer3, and others (including flattening, forward and sigmoid.) (b) The profiling results of the Convolution, Pooling and ReLU.

Fig. 2. Profile AIRBiS inference program with the average time cost of computation

With the theoretical analysis of the execution time and memory utilization of neural networks, we have analyzed the complexity of AIRBiSNet and the edge FPGA-based low-power inference cluster. As shown in Figure 3, there are three CNN layers and other operations such as load model, flatten, forward, and sigmoid in the AIRBiS inference process. In addition, each CNN layer has the computation of Convolution, ReLU, and Pooling. The average time cost of AIRBiS neural network inference computation is shown in Figure 2(a). The first three layers and other operations like flattening take 14%, 69%, 15%, and 2% of the execution time, respectively. The second layer takes the most time compared to the other two layers because it has the most input data. Figure 2(b) presents the profiling results of the Convolution, Pooling, and ReLU computation in the three CNN layers. The convolution operation takes 95% of the total time cost, the Pooling 3%, and ReLU 2%. Since the convolution mainly computes the massive multiplication operations, we noticed that most of the computation time is consumed in this operation [25].

C. AIRBiS Inference Acceleration Design

We propose two architectures, namely; (a) non-pipelined inference and (b) pipelined inference for mapping neural networks on software into programmable FPGA clusters to achieve high performance and low power consumption. The non-pipelined inference architecture described in Figure 3 (a) is used for mapping neural networks that can be flexible and scalable, thus allowing adjustable scale according to the system load and providing fault tolerance. For larger DNNs, the pipelined inference architecture described in Figure 3 (b) can

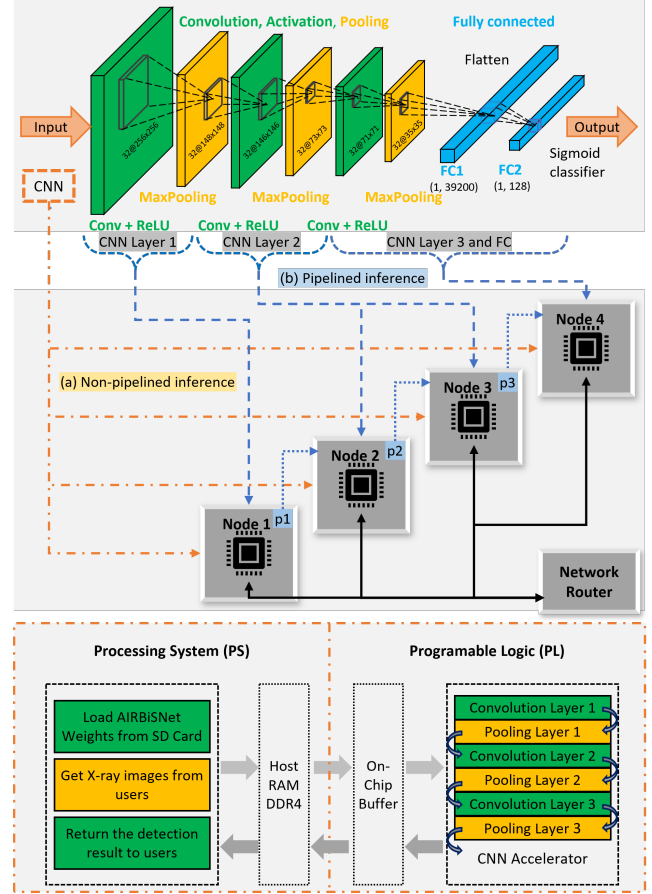


Fig. 3. AIRBiS parallel inference architecture. (a) Non-pipelined inference. (b) Pipelined inference.

be applied to split and map the network into different nodes of the programmable logic cluster according to the layer level. This allows the system to support high-precision inference tasks and reduce network latency and power consumption. Also, the proposed programmable clusters are used for edge collaborative learning, allowing the network to learn from new data accumulated in each edge device to improve detection accuracy and robustness.

As shown in Figure 3, the proposed AIRBiS parallel inference architecture is based on a scalable and reconfigurable AI chip that consists of clustered FPGA development boards (Zynq UltraScale+ MPSoC ZCU102 Evaluation Kit). The processing unit in the AI chip is an ARM processor which initializes the Programmable Logic (PL) through the control bus to construct the convolution layer parameters during run time.

III. EVALUATION

A. Evaluation Methodology

We used the X-ray image dataset described in Table I to evaluate our proposed system. We randomly selected 700 images from each of the two labels as the test data. The AIRBiSNet was trained with conventional training and collab-

orative learning techniques. The pneumonia detection model (AIRBiSNet) was trained on multiple CPU and GPU servers. Each server has 64GB DDR4 RAM and NVIDIA GeForce RTX 2060 GPU.

The inference FPGA cluster was implemented using Xilinx's EDA suite (SDSoC, Vivado, and Vivado HLS). The average classification time of predicting each image on our proposed system, CPU, and GPU are measured and compared (see Figure 5). The power consumption was measured by a power meter, and also refer to related studies [26] and software simulation results.

B. Evaluation Results

1) *AIRBiS Detection Accuracy*: Table II presents the detection accuracy of COVID-19 and Non-COVID-19 infected images calculated by Equation 2.

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + FP + TN} \times 100\% \quad (2)$$

TABLE II
PNEUMONIA (COVID-19) DETECTION ACCURACY OF THE 256×256 (PX)
INPUT MODEL.

Real \ Predict	Infected	Non_COVID	Sum	Accuracy
Infected	(TP) 640	(FN) 60	700	91.4%
Non_COVID	(FP) 7	(TN) 693	700	99.0%
Lung_Opacity	4	246	250	98.4%
Viral_Pneumonia	3	47	50	94.0%
Normal	0	400	400	100.0%
Correct	640	693	1400	95.2%

TP: True positive. TN: True negative.
FP: False positive. FN: False negative.

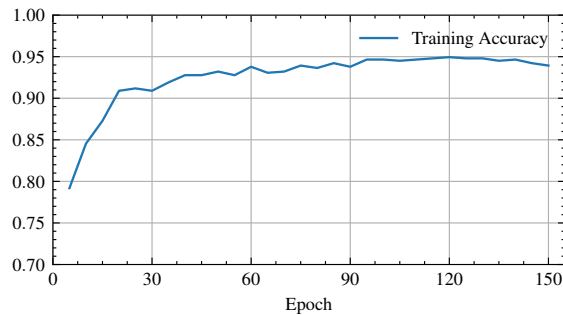


Fig. 4. AIRBiS collaborative learning accuracy over a Non-IID case.

The concentrated training scheme recorded a final accuracy of 95.2% as shown in Table II. In general, as the number of nodes and the amount of data increase, the accuracy of the final model gradually improves and eventually approaches that of the traditional training method. Figure 4 shows the training process of the non-identical and independent distribution (Non-IID) scenario with a final accuracy of 94.7%. Centralized training requires collecting all training data, which sometimes has privacy issues. Distributed collaborative learning solves this issue by aggregating weights from distributed learning

nodes. Although part of the original image data can be inferred from the weights, inexhaustive X-ray images do not cause privacy issues. The collaborative machine learning scheme can improve the accuracy of the detection model by learning from more data provided by different hospitals. It still achieves good performance accuracy even with uneven data distribution. At the same time, because the training data is not transferred over the network to a centralized server for learning, the privacy of the medical data is ensured.

We compared the performance of AIRBiS with some other state-of-the-art studies. These studies also used machine learning techniques like CNN and a COVID-19 dataset. The studies in [27, 28, 29, 30, 31, 32, 24] used not more than 2000 lung X-ray images and 1000 COVID-19 images respectively. This disproportionate number of COVID-19 images leads to data imbalance, affecting the CNN model's robustness. Similarly, Lin in [33] used a more extensive dataset with 15,478 chest X-ray images, yet only 473 COVID-19 images, which still left the dataset heavily imbalanced. Compared to other studies, the proposed system used a better proportion of COVID-19 images, which validates the robustness of the system [34].

We optimized the system parameters through extensive experiments with feature visualization feedback. We considered several data augmentation methods, such as shift, scale, contrast, noise, rotation, and reflection, some of which did not perform well. Also, some typical CNN optimization methods, such as the Adam optimizer, batch normalization, and dropout, were applied. Furthermore, AIRBiSNet (256×256) utilized only 3.7 million parameters, which is about a 53% reduction in the number of parameters compared to the work in [24].

The proposed AIRBiSNet-64 further improves the inference speed to better fit the demand for high-speed detection in pneumonia detection for different scenarios with only a 3% loss in detection accuracy and three times faster detection speed than the high-precision AIRBiSNet-256. This is especially important for rapid end-side detection.

2) *Inference Time and Power Consumption*: As explained in the previous section, AIRBiS inference is accelerated on a reconfigurable and scalable FPGA cluster to improve the speed of pneumonia detection and achieve low-power consumption. Figure 5 shows a comparison of the average energy consumption per inference and power consumption over various platforms (GPU, Desktop CPU, ARM CPU, AIRBiS-HW) using the test dataset. The four subplots show the average energy required for each inference for the four platforms with four different input sizes, namely, $64 \times 64 \times 1$, $128 \times 128 \times 1$, $256 \times 256 \times 1$, and $128 \times 128 \times 3$. Experiments with four different input data sizes demonstrate that the proposed FPGA-based inference platform is 1.8 to 13 times more energy-efficient than the GPU, 2.6 to 5.2 times that of the desktop CPU, and 2.2 to 2.6 times that of the ARM CPU.

Table III shows the hardware resource utilization of the $256 \times 256 \times 1$ AIRBiS-HW design. It utilizes approximately 19.9% of the lookup table (LUT) on the Xilinx ZCU102 FPGA board. Many of the configurable logic blocks (CLB) on the ZCU102 are implemented using a minimal amount of RAM

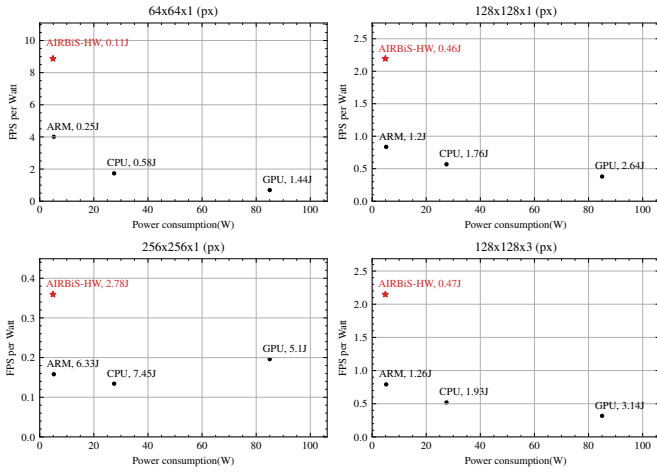


Fig. 5. AIRBiS Inference power consumption (W) and fps per watt (FPS/Watt) comparison with different computing platforms when giving different input sizes.

in the form of LUT that contains all the combinatorial logic used for the AIRBiS design. About 19.9% of LUT and 9.7% of the FF, among other resources, were also utilized for the design. On average, about 11.6% of the FPGA resources were utilized, which is an acceptable result for the target application.

TABLE III
HARDWARE COMPLEXITY OF AIRBiS INFERENCE ACCELERATOR.

Resource	Utilization	Available	Utilization rate (%)
LUT	54585	274080	19.9
LUTRAM	3668	144000	2.5
FF	53035	548160	9.7
DSP	35	2520	1.4
BUFG	4	404	1.0
MMCM	1	4	25

IV. CONCLUSION

In this work, we presented methods and architectures for scaling deep learning inference for pneumonia detection in chest X-ray images based on a reconfigurable self-contained hardware platform named AIRBiS. AIRBiS is a deep learning-based pneumonia detection system in chest X-ray images. The system is based on a high-performance, low-power, reconfigurable FPGA cluster for inference, a robust collaborative-learning mechanism for privacy preservation, and an interactive user interface for effective operation and monitoring. We were able to optimize and satisfy design constraints such as power consumption. The performance evaluation results show that the proposed AIRBiS system achieves 95.2% detection accuracy of pneumonia over the collected test data with the computer-aided diagnosis scenario. In the rapid batch detection scenario, the detection could be accelerated to 0.023s. Moreover, the system inference acceleration is 13 times more energy-efficient than GPUs, 5.2 times more than CPUs, and 2.6 times that of ARM CPUs. Overall, the proposed approach demonstrates the potential for using reconfigurable hardware platforms for AI applications in healthcare and other fields.

For future work, we plan to address the training scheme with a decentralized approach for better performance and security. Investigating how to optimize deeper CNN models such as ResNet50 or ResNet101 to fit into the FPGA system is one of our future directions. We also plan to improve the fault tolerance of the AIRBiS platform.

REFERENCES

- [1] P. S. Muthukumarana and A. C. Aponso, "A review on deep learning based image classification of plant diseases," *International Journal of Computer Theory and Engineering*, vol. 12, no. 5, 2020.
- [2] T. H. Vu, R. Murakami, Y. Okuyama, and A. Ben Abdallah, "Efficient optimization and hardware acceleration of cnns towards the design of a scalable neuro inspired architecture in hardware," in *2018 IEEE International Conference on Big Data and Smart Computing (Big-Comp)*, 2018, pp. 326–332.
- [3] S. Roy, W. Menapace, S. Oei, B. Luijten, E. Fini, C. Saltori, I. Huijben, N. Chennakeshava, F. Mento, A. Sentelli *et al.*, "Deep learning for classification and localization of covid-19 markers in point-of-care lung ultrasound," *IEEE transactions on medical imaging*, vol. 39, no. 8, pp. 2676–2687, 2020.
- [4] M. Yasefiani, A. Z. Hamadani, A. I. Maghsoodi, and A. Mosavi, "Pneumonia detection proposing a hybrid deep convolutional neural network based on two parallel visual geometry group architectures and machine learning classifiers," *IEEE Access*, vol. 10, pp. 62 110–62 128, 2022.
- [5] W. Lie, B. Jiang, and W. Zhao, "Obstetric imaging diagnostic platform based on cloud computing technology under the background of smart medical big data and deep learning," *IEEE Access*, vol. 8, pp. 78 265–78 278, 2020.
- [6] I. Ahmed, G. Jeon, and F. Piccialli, "A deep-learning-based smart healthcare system for patient's discomfort detection at the edge of internet of things," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10 318–10 326, 2021.
- [7] M. Almutairi, L. A. Gabralla, S. Abubakar, and H. Chiroma, "Detecting elderly behaviors based on deep learning for healthcare: Recent advances, methods, real-world applications and challenges," *IEEE Access*, vol. 10, pp. 69 802–69 821, 2022.
- [8] B. Alouffi, M. Hasnain, A. Alharbi, W. Alosaimi, H. Alyami, and M. Ayaz, "A systematic literature review on cloud computing security: Threats and mitigation strategies," *IEEE Access*, vol. 9, pp. 57 792–57 807, 2021.
- [9] F. Wang, M. Zhang, X. Wang, X. Ma, and J. Liu, "Deep learning for edge computing applications: A state-of-the-art survey," *IEEE Access*, vol. 8, pp. 58 322–58 336, 2020.
- [10] X. Wang, Y. Han, V. C. M. Leung, D. Niyato, X. Yan, and X. Chen, "Convergence of edge computing and deep learning: A comprehensive survey," *IEEE Communica-*

- tions *Surveys and Tutorials*, vol. 22, no. 2, pp. 869–904, 2020.
- [11] K. Cao, S. Hu, Y. Shi, A. W. Colombo, S. Karnouskos, and X. Li, “A survey on edge and edge-cloud computing assisted cyber-physical systems,” *IEEE Transactions on Industrial Informatics*, vol. 17, no. 11, pp. 7806–7819, 2021.
 - [12] J. Chen and X. Ran, “Deep learning with edge computing: A review,” *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1655–1674, 2019.
 - [13] “Coronavirus Update (Live) from COVID-19 Virus Pandemic - Worldometer,” 2022. [Online]. Available: <https://www.worldometers.info/coronavirus/>
 - [14] T. Ai, Z. Yang, H. Hou, C. Zhan, C. Chen, W. Lv, Q. Tao, Z. Sun, and L. Xia, “Correlation of chest ct and rt-pcr testing for coronavirus disease 2019 (covid-19) in china: a report of 1014 cases,” *Radiology*, vol. 296, no. 2, pp. E32–E40, 2020.
 - [15] Y. Fang, H. Zhang, J. Xie, M. Lin, L. Ying, P. Pang, and W. Ji, “Sensitivity of chest ct for covid-19: comparison to rt-pcr,” *Radiology*, vol. 296, no. 2, pp. E115–E117, 2020.
 - [16] S. A. Harmon, T. H. Sanford, S. Xu, E. B. Turkbey, H. Roth, Z. Xu, D. Yang, A. Myronenko, V. Anderson, A. Amalou *et al.*, “Artificial intelligence for the detection of covid-19 pneumonia on chest ct using multinational datasets,” *Nature communications*, vol. 11, no. 1, pp. 1–7, 2020.
 - [17] A. Asraf, M. Z. Islam, M. R. Haque, and M. M. Islam, “Deep learning applications to combat novel coronavirus (covid-19) pandemic,” *SN Computer Science*, vol. 1, no. 6, pp. 1–7, 2020.
 - [18] M. Mosley, *Covid-19: what you need to know about the coronavirus and the race for the vaccine*. Short Books Ltd, 2020, oCLC: 1180152280.
 - [19] J. Wang, M. Nakamura, and A. B. Abdallah, “Efficient ai-enabled pneumonia detection in chest x-ray images,” in *2022 IEEE 4th Global Conference on Life Sciences and Technologies (LifeTech)*. IEEE, 2022, pp. 470–474.
 - [20] J. Wang, O. M. Ikechukwu, K. N. Dang, and A. B. Abdallah, “Spike-event x-ray image classification for 3d-NoC-based neuromorphic pneumonia detection,” *Electronics*, vol. 11, no. 24, p. 4157, Dec 2022.
 - [21] T. Fukuchi, M. I. Ogbodo, J. Wang, K. N. Dang, and A. Ben Abdallah, “Efficient pneumonia detection method and implementation in chest x-ray images based on a neuromorphic spiking neural network,” in *Conference on Computational Collective Intelligence Technologies and Applications*. Springer, 2022, pp. 311–321.
 - [22] O. Yuuki, J. Wang, T. Fukuchi, and A. B. Abdallah, “Parallelization and hardware mapping of deep neural network on reconfigurable platform for ai-enabled biomedical system,” in *SHS Web of Conferences*, vol. 139. EDP Sciences, 2022, p. 03005.
 - [23] M. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M. Kadir, Z. Mahbub, K. Islam, M. Khan, A. Iqbal, N. Al-Emadi *et al.*, “Can ai help in screening viral and covid-19 pneumonia? arxiv 2020,” *arXiv preprint arXiv:2003.13145*, 2003.
 - [24] T. Rahman, A. Khandakar, Y. Qiblawey, A. Tahir, S. Kiranyaz, S. B. A. Kashem, M. T. Islam, S. Al Maadeed, S. M. Zughaier, M. S. Khan *et al.*, “Exploring the effect of image enhancement techniques on covid-19 detection using chest x-ray images,” *Computers in biology and medicine*, vol. 132, p. 104319, 2021.
 - [25] O. Yuuki, J. Wang, O. M. Ikechukwu, and A. B. Abdallah, “Hardware acceleration of convolution neural network for ai-enabled realtime biomedical system,” in *SHS Web of Conferences*, vol. 102. EDP Sciences, 2021, p. 04019.
 - [26] C. N. Hesse, “Analysis and comparison of performance and power consumption of neural networks on cpu, gpu, tpu and fpga,” 2021.
 - [27] X. Wu, H. Hui, M. Niu, L. Li, L. Wang, B. He, X. Yang, L. Li, H. Li, J. Tian *et al.*, “Deep learning-based multi-view fusion model for screening 2019 novel coronavirus pneumonia: a multicentre study,” *European Journal of Radiology*, vol. 128, p. 109041, 2020.
 - [28] R. M. Pereira, D. Bertolini, L. O. Teixeira, C. N. Silla Jr, and Y. M. Costa, “Covid-19 identification in chest x-ray images on flat and hierarchical classification scenarios,” *Computer methods and programs in biomedicine*, vol. 194, p. 105532, 2020.
 - [29] A. Mobiny, P. A. Cicalese, S. Zare, P. Yuan, M. Abavisani, C. C. Wu, J. Ahuja, P. M. de Groot, and H. Van Nguyen, “Radiologist-level covid-19 detection using ct scans with detail-oriented capsule networks,” *arXiv preprint arXiv:2004.07407*, 2020.
 - [30] E. E.-D. Hemdan, M. A. Shouman, and M. E. Karar, “Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images,” *arXiv preprint arXiv:2003.11055*, 2020.
 - [31] A. Abbas, M. M. Abdelsamea, and M. M. Gaber, “Classification of covid-19 in chest x-ray images using detrack deep convolutional neural network,” *Applied Intelligence*, vol. 51, no. 2, pp. 854–864, 2021.
 - [32] M. Heidari, S. Mirniaharikandehei, A. Z. Khuzani, G. Danala, Y. Qiu, and B. Zheng, “Improving the performance of cnn to predict the likelihood of covid-19 using chest x-ray images with preprocessing algorithms,” *International journal of medical informatics*, vol. 144, p. 104284, 2020.
 - [33] T.-C. Lin and H.-C. Lee, “Covid-19 chest radiography images analysis based on integration of image preprocess, guided grad-cam, machine learning and risk management,” in *Proceedings of the 4th International Conference on Medical and Health Informatics*, 2020, pp. 281–288.
 - [34] F. Bao, Y. Deng, Y. Kong, Z. Ren, J. Suo, and Q. Dai, “Learning deep landmarks for imbalanced classification,” *IEEE transactions on neural networks and learning systems*, vol. 31, no. 8, pp. 2691–2704, 2019.