Hardware Implementation of a Deep Learning-based Autonomous System for Smart Homes using Field Programmable Gate Array Technology

Mohamed Tounsi
College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia | Automated Systems and Soft Computing Lab (ASSCL), Prince Sultan University, Riyadh, Saudi Arabia
mtounsi@psu.edu.sa

Ali Jafer Mahdi
College of Information Technology Engineering, Al-Zahraa University for Women, 56001, Karbala, Iraq
ali.j.mahdi@alzahraa.edu.iq

Mahmood Anees Ahmed
Medical Instrumentation Techniques Engineering Department, College of Medical Techniques. Al-Farahidi University, Baghdad, 10001, Iraq
mahomd.anees@uoalfarahidi.edu.iq

Ahmad Taher Azar
College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia | Automated Systems and Soft Computing Lab (ASSCL), Prince Sultan University, Riyadh, Saudi Arabia | Faculty of Computers and Artificial Intelligence, Benha University, Benha, Egypt
aazar@psu.edu.sa

Drai Ahmed Smait
College of Engineering, The University of Mashreq, Baghdad 10001, Iraq
draialisawi@uom.edu.iq

Saim Ahmed
College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia | Automated Systems and Soft Computing Lab (ASSCL), Prince Sultan University, Riyadh, Saudi Arabia
zhaider@psu.edu.sa

Ali Mahdi Zalzala
Department of Electronics and Communication, College of Engineering, Uruk University, Baghdad 10001, Iraq
a.zalzala@uruk.edu.iq

Ibraheem Kasim Ibraheem
Department of Electrical Engineering, College of Engineering, University of Baghdad, Baghdad 10001, Iraq
ibraheemki@coeng.uobaghdad.edu.iq (corresponding author)

Received: 12 July 2024 | Revised: 9 August 2024 | Accepted: 11 August 2024

ABSTRACT

The current study uses Field-Programmable Gate Array (FPGA) hardware to advance smart home technology through a self-learning system. The proposed intelligent three-hidden layer system outperformed prior systems with 99.21% accuracy using real-world data from the MavPad dataset. The research shows that FPGA solutions can do difficult computations in seconds. The study also examines the difficulties of maximizing performance with limited resources when incorporating deep learning technologies into FPGAs. Despite these challenges, the research shows that FPGA-based solutions improve home technology. It promotes the integration of sophisticated learning algorithms into ordinary electronics to boost their intelligence.

Keywords-smart home; FPGA; autonomous system; neural networks; deep learning; optimization techniques; scalability; hardware implementation

I. INTRODUCTION

Home control has been revolutionized by recent advancements in FPGA and AI learning technologies. Deep Learning (DL) has attracted the interest of both industry and academia. DL is a method to enable computers to mimic human thought processes and it is employed to predict human behavior in smart homes connected by the Internet of Things (IoT). The predictive power of DL is enhanced by its ability to concurrently analyze multiple datasets. To ensure its effectiveness and speed, particularly in real-time applications, it was tested using actual data.

To meet smart home demands, FPGAs outperform traditional processors by dynamically adapting their hardware for DL models, offering reduced latency, better energy efficiency, and real-time processing capabilities [1]. Implementing DL on FPGAs is challenging, requiring expertise in both software and hardware. However, the benefits—like enabling advanced smart home applications such as adaptive energy management and personalized security—are undeniable. In [2], the authors conducted a detailed survey on autonomous smart machines, covering applications, development, and methodologies. In [3, 4], an FPGA-based DL accelerator was proposed, categorizing contemporary accelerators and reviewing state-of-the-art techniques to enhance hardware implementations. Combining FPGA technology with DL significantly enhances image analysis speed and efficiency. In [5-7], authors improved MEC decision-making, accelerated deep neural networks with FPGAs, and leveraged 5G for safer urban driving. RNNs and CNNs are applied in smart homes [3, 8-19], with FPGAs further optimizing DL models for these environments [20-27]. Direct applications of DL in smart home devices are detailed in [28-30]. Challenges in neural networks are addressed in our work by developing an FPGA-based system to enhance DL in smart homes. Key motivations include:

- Need for Intelligent Systems: There is a rising demand for smart homes that can autonomously adapt and make decisions.
- Hardware Acceleration: FPGAs excel at fast, multi-task processing and are ideal for accelerating DL with limited resources.
- Hardware Optimization: Closing the performance gap of DL models on hardware, particularly for smart home tech.
- Expanding Applications: Exploring FPGA-based DL's potential in energy management, security, and edge computing.

Our work advances the literature by optimizing DL on FPGA hardware for smart homes and beyond. Key contributions include:

- Innovative Hardware Implementation: The practical use of DL on FPGA for smart home systems is demonstrated.
- Optimization Techniques: Specific optimizations for efficient FPGA mapping are applied, achieving high performance.
- Comparative Analysis: Significant speed and efficiency gains over traditional software-based solutions are highlighted.
- Versatile Application Potential: The adaptability for broader applications like energy management and personalized security is showcased.

II. RESEARCH METHODOLOGY

A. Proposed Design

The architecture design for the proposed smart home system begins with eight subsystems, each connected to identical sensors and actuators through specific agents. These agents facilitate communication, data filtering, and security between subsystems and the central Intelligent Agent (IA), the Steward. The local IA is responsible for partner interfaces, communication, learning behaviors, executing actions, security, and data storage. The design emphasizes a prediction system managed locally by the IA, eliminating the need for third-party cloud services. The IA uses DL and Machine Learning (ML) algorithms to predict and execute actions based on node data. This approach addresses challenges posed by cloud computing in real-time systems.

B. Experimental Design

The MavPad dataset, which records interactions between a person and its home environment across a kitchen, bathroom, bedroom, and living room, was analyzed. The dataset includes data from 127 nodes, comprising 86 sensors and 41 actuators, over a period of seven weeks. Each day has data files detailing the date, time, state, level, zone number, and source. After noise removal, we pre-processed the data using MATLAB, consolidating all 49 days into a single file (OP.mat). This file contains over 4 million rows of data. AI techniques were then applied to predict the user's next action based on environmental parameters.

We factorize a twofold perception vector for every sensor as $X_t = (x_{1t}, x_{2t},..., x_{86t})$. Actuators are meant with $Y_t \in 1,..., Q$ for every one of the Q potential states, which portray the state of every actuator at a given time t. We chose to look at client action in the house latrine zone to more readily comprehend the adequacy of the network procedure we used in our examinations.

Two distinct sensor configurations were used to anticipate the partner's next move. Each of the seven sensors present in the restroom is utilized in the main model. All 86 sensors are used in every ecological zone in the following scenario. A binary actuator operates in bathroom B5 during regular use. Actuator B5 is used to control the light above the mirror. We used the first month of the examination, consisting of 28 days, as the preparation dataset and the fifth week as the test dataset.

III. EXPERIMENTAL RESULTS AND DISCUSSION

Artificial Neural Networks (ANNs) offer benefits like self-organization, adaptability, and real-time output, but they also face challenges, such as requiring significant resources for DL and facing the risk of overfitting with insufficient data [31-35]. Advances in GPUs, parallel architectures, and data availability have enhanced ANNs' capabilities. The architecture, including the number of layers and neuron density, affects computation speed and accuracy. We empirically determined the optimal network depth, neuron count, and training window size to achieve the best classification performance with the available resources.

IV. SIMULATION RESULTS

In the beginning, we just used one hidden layer. Then, the ideal representation was found empirically by increasing the number of neurons in the hidden layer. The network's depth was increased by adding a second layer. In order to get the optimal representation, we fixed the number of neurons in the first layer and increased the number in the second layer. Again, we used a similar method for the third layer to optimize the number of nodes in our network. Drop-out layers were used to mitigate the over-fitting problem. A half-layer drop-out was introduced to each pair of fully connected levels. Removing some of a network's nodes avoided excessive parameter updating during training. This drop-out method could aid in lessening the impact of overfitting. In our experiments, we employed two kinds of zones: local and global. While the worldwide zone predicts the B5 actuator using all 86 environmental sensors, the local zone requires only seven input sensors. Tables I and II display the experiment findings.

TABLE I. MULTI-FACET ANN ACCURACY AND PREDICTION TIME RESULTS UTILIZING LOCAL ZONE SENSORS

No. of hidden layers	Accuracy	t (s)
1	0.9589	0.249
2	0.9917	0.391
3	0.9921	0.770

Table I and Figure 1 show the accuracy and forecast time of a multi-faceted ANN utilizing local zone sensors versus the number of secret layers. The table shows that accuracy improves proportionately to the ANN's complexity or the number of hidden layers added. This shows that more complex ANN topologies can produce more accurate predictions by better capturing the complex relationships and patterns found in the input data. Notably, this improved accuracy is accompanied by a longer prediction time. So, selecting the number of hidden layers to use is finding a middle ground between those two, accuracy and calculation time.

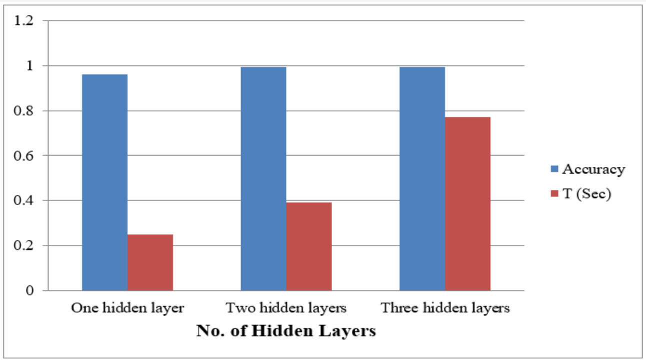

Fig. 1. Multi-facet ANN's accuracy and forecast time results utilizing local zone sensors.

TABLE II. ESTIMATED TIME AND ACCURACY OF THE ANNS WHEN EMPLOYING GLOBAL ZONE SENSORS.

No. of hidden layers	Accuracy	t (sec)
1	0.9021	7.1027
2	0.9540	10.1084
3	0.9615	14.7064

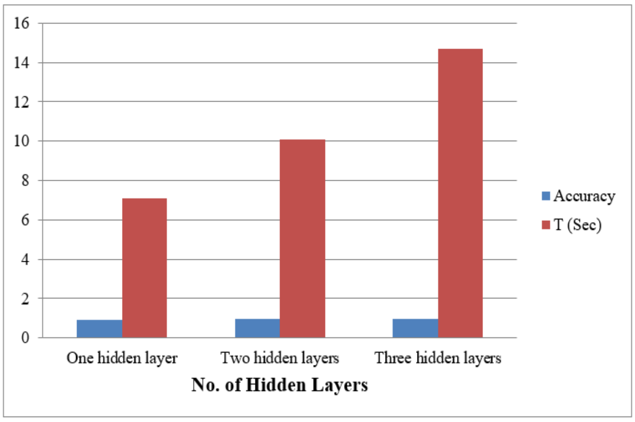

Fig. 2. Multilayer ANNs with global zone sensors are more accurate and better at forecasting time.

Table II and Figure 2 provide insights into the effectiveness of a layered network system when collecting data on a global scale. The results clearly demonstrate that as the number of layers increases, the system's predictive accuracy improves. Regarding the time required for predictions and their accuracy, it becomes apparent that adding more hidden layers results in longer processing times. This again indicates a trade-off that needs careful consideration. In some situations, a model that takes longer to produce more accurate results might be acceptable, but in scenarios where speed is crucial, this may not be feasible. Therefore, it is essential to weigh the importance of speed versus accuracy based on the specific requirements. In cases where quick responses are necessary, opting for a simpler model, even with slightly reduced accuracy, might be the better choice and when precision is the top priority, a more complex model could be appropriate. The decision ultimately depends on the intended use of the model, particularly in applications involving global data sensing.

A. Hardware Implementation

Table III shows how much hardware is used and compares it to other classes of FPGA implementations. The table compares four FPGA platforms: Virtex-4 LX40, XC5VSX50T, XC7Z010T, and XC3S500E. It looks at essential features like DSP48Es/multipliers, Slice Look-Up Tables (LUTs), Slice Registers, slack in Combinational Paths (CPS), and slack in CP with both setup and hold (CPPSL). The XC5VSX50T FPGA needs the most resources since it has 8245 Slice LUTs and 2445 Slice Registers. This means it is good at doing complex math and logical tasks. However, because it can do so much, it also uses more power, as shown by its high CPS (538 M) and CPPSL (11.8 M) values.

TABLE III. FPGA ANN RESOURCE USE AND COMPARISONS TO OTHER FPGA IMPLEMENTATIONS

Platform	Slice LUTs	Slice registers	DSP48Es/ multipliers	CPS	CPPSL
XC5VSX50T	8245	2445	72	538 M	11.8 M
Virtex-4 LX40	4548	None	10	1.4 M	2.09 M
XC7Z010T	4234	3065	30	74.5 M	12.5 M
Proposed XC3S500E	4140	3064	22	483.5 M	33.4 M

When looking at different FPGA systems, each one has its advantages. The XC7Z010T is well-rounded, making it useful for many things. The Virtex-4 LX40 works better for easier jobs that don't require too much strength. The XC5VSX50T is suggested for robust capacity. Lastly, the presented XC3S500E is an excellent middle-ground option. Depending on the application requirements, this paper simplifies information decision-making for selecting the best FPGA platform. Table IV shows the specifics of sensor prediction times in different places and network setups.

TABLE IV. PREDICTION TIME RESULTS FOR HARDWARE-BASED MULTILAYER ANNS USING SENSORS IN THE LOCAL AND GLOBAL ZONES

Zone Name	No. of hidden layers	t (sec)
Local zone sensors	1	0.095
	2	0.203
	3	0.489
Global zone sensors	1	0.333
	2	0.662
	3	1.845

A comprehensive summary of the prediction time findings for a hardware-based multilayer neural network that differentiates between local and global sensor usage is provided in Table IV. The table shows how well the network does at different levels of complexity and with various sensors. As we add more hidden layers to the sensors in the local area, the time it takes to make predictions increases. Specifically, it takes about 0.095 seconds to make a prediction with one hidden layer. With two layers, it takes longer, about 0.203 seconds. And with three layers, it takes even longer, about 0.489 seconds. This shows that networks need more time and power to process information when they get more complex.

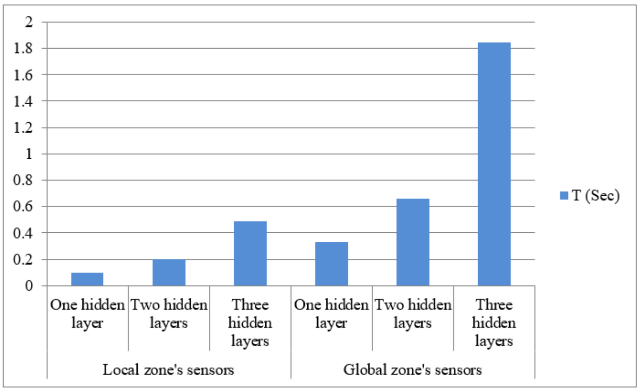

Fig. 3. Prediction time results for hardware-based multilayer ANNs using sensors in the local and global zones.

A similar phenomenon is observed with sensors in the broader area. When additional hidden layers are incorporated, the prediction time is significantly prolonged. The time increases substantially, from 0.333 s with one layer to 0.662 s with two layers, and further escalates to 1.845 s with three layers. This indicates that as the network complexity increases, it becomes more challenging for the hardware to maintain performance. The Figures illustrate the necessity of balancing the desired accuracy of predictions with the speed and efficiency of the network, taking into account the location of the sensors. Also, the results show that sensors in the local area always make quicker predictions than those from the wider region, no matter how complex the network is. This means that local sensors might be the smarter choice for things that need to happen in real-time because they can work faster with the considered equipment.

B. Discussion

Some challenges were encountered during our study on the use of FPGA to enable smart homes to learn and adapt. As we tested these smart systems for real-world applications, we gained valuable insights into the difficulties involved in making them function effectively.

- Recapitulation of Limitations and Challenges: We encountered several significant obstacles. Implementing the smart learning component on the FPGA chip proved challenging due to the chip's limited computational power, and extending its functionality to the entire house was even more difficult. These challenges have the potential to reduce the effectiveness or increase the complexity of the design.

- Influence of Limitations on Experimental Results: Experimental results showing significant increases in processing speed and prediction accuracy indicate effective navigation through these hurdles. Model complexity and system scalability were compromised due to FPGA computational resource limits. The system's prediction precision and efficiency are primarily caused by the optimization tactics that fit the DL model to the FPGA's restrictions. These results also indicate that scaling issues were not entirely resolved, suggesting that future systems must bridge the gap between theoretical computations and actual implementation in varying smart home scenarios.
- Addressing Challenges: We devised smart ways, like doing many things simultaneously and fine-tuning our approach, to deal with the insufficient chip power. Our system's setup and these tricks helped us reach a quality that looks good for future smart homes. The results that the main issues were overcome satisfactorily.
- Implications for Future Research: A significant step has been made with the current research, but some challenges remain. More work is needed to address the issues related with improving the system's performance in additional homes while maintaining speed and accuracy. In the future, smarter methods could be explored to ensure smooth operation, different types of chips might be integrated, or more efficient models could be developed that require less power and can scale effectively.

The experiment in studying how FPGA can be utilized for smart home technology through DL has been both rewarding and challenging. The tests conducted and the findings obtained, despite the encountered problems, provide insights into the current status and guide future directions. As efforts to advance FPGA applications in smart homes continue, the insights and groundwork established here will support future research in this promising field.

V. CONCLUSIONS

This work pioneered using FPGA technology to construct a deep learning-based autonomous smart home system. Our thorough calculations and practical evaluations show that FPGA platforms may improve smart home intelligence and efficiency using sophisticated deep learning techniques. The simulation results show that three hidden layers in the neural network design for local zone sensors yield 99.21% accuracy. This accuracy delivers the durability of our deep learning model and the usefulness of FPGA for real-time demanding computing workloads. Our investigations showed that the FPGA can execute high-performance calculations, with prediction durations ranging from 0.095 s for one hidden layer to 0.489 s for a three-hidden-layer configuration in local zone sensors. Importantly, our findings show the difficulties of scaling FPGA-based deep learning systems for smart home applications. Our study found that the best results were obtained in controlled settings. But, when we tried to apply this more widely, we faced some challenges because FPGA platforms have limited resources and space. We learned that the use of FPGAs improves smart home systems with deep-learning models. But, it is not all smooth sailing - there are significant challenges in making things bigger or using resources better that we need to look more into. So, to make smarter homes work well with the current technology, we need to work better combining deep learning and FPGA technology to get past these challenges.

ACKNOWLEDGMENT

The authors would like to acknowledge the support of Prince Sultan University in paying the Article Processing Charges (APC) for this publication. Special acknowledgment to Automated Systems and Soft Computing Lab (ASSCL), Prince Sultan University, Riyadh, Saudi Arabia..

REFERENCES

[1] S. Vaidyanathan, A. T. Azar, K. Rajagopal, A. Sambas, S. Kacar, and U. Cavusoglu, "A new hyperchaotic temperature fluctuations model, its circuit simulation, FPGA implementation and an application to image encryption," *International Journal of Simulation and Process Modelling*, vol. 13, no. 3, pp. 281–296, Jan. 2018, https://doi.org/10.1504/IJSPM.2018.093113.

[2] J. C. S. D. Anjos *et al.*, "A Survey on Collaborative Learning for Intelligent Autonomous Systems," *ACM Computing Surveys*, vol. 56, no. 4, Aug. 2023, Art. no. 98, https://doi.org/10.1145/3625544.

[3] A. G. Blaiech, K. Ben Khalifa, C. Valderrama, M. A. C. Fernandes, and M. H. Bedoui, "A Survey and Taxonomy of FPGA-based Deep Learning Accelerators," *Journal of Systems Architecture*, vol. 98, pp. 331–345, Sep. 2019, https://doi.org/10.1016/j.sysarc.2019.01.007.

[4] R. Gayathri, "Onboard target detection in hyperspectral image based on deep learning with FPGA implementation," *Microprocessors and Microsystems*, vol. 85, Sep. 2021, Art. no. 104313, https://doi.org/10.1016/j.micpro.2021.104313.

[5] A. Shakarami, A. Shahidinejad, and M. Ghobaei-Arani, "An autonomous computation offloading strategy in Mobile Edge Computing: A deep learning-based hybrid approach," *Journal of Network and Computer Applications*, vol. 178, Mar. 2021, Art. no. 102974, https://doi.org/10.1016/j.jnca.2021.102974.

[6] Y. You *et al.*, "New paradigm of FPGA-based computational intelligence from surveying the implementation of DNN accelerators," *Design Automation for Embedded Systems*, vol. 26, no. 1, pp. 1–27, Mar. 2022, https://doi.org/10.1007/s10617-021-09256-8.

[7] K. Yu, L. Lin, M. Alazab, L. Tan, and B. Gu, "Deep Learning-Based Traffic Safety Solution for a Mixture of Autonomous and Manual Vehicles in a 5G-Enabled Intelligent Transportation System," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4337–4347, Jul. 2021, https://doi.org/10.1109/TITS.2020.3042504.

[8] E. H. C. Tourad and M. Eleuldj, "Survey of Deep Learning Neural Networks Implementation on FPGAs," in *5th International Conference on Cloud Computing and Artificial Intelligence: Technologies and Applications*, Marrakesh, Morocco, Nov. 2020, pp. 1–8, https://doi.org/10.1109/CloudTech49835.2020.9365911.

[9] P. Gayathri, A. Stalin, and S. Anand, "Intelligent Smart Home Security System: A Deep Learning Approach," in *10th Region 10 Humanitarian Technology Conference (R10-HTC)*, Hyderabad, India, Sep. 2022, pp. 438–444, https://doi.org/10.1109/R10-HTC54060.2022.9929516.

[10] C. Pham-Quoc, T. H. Q. Bao, and T. N. Thinh, "FPGA/AI-Powered Architecture for Anomaly Network Intrusion Detection Systems," *Electronics*, vol. 12, no. 3, Jan. 2023, Art. no. 668, https://doi.org/10.3390/electronics12030668.

[11] B. M. H. Alhafidh, A. I. Daood, M. M. Alawad, and W. Allen, "FPGA Hardware Implementation of Smart Home Autonomous System Based on Deep Learning," in *International Conference on Internet of Things*, Seattle, WA, USA, Jun. 2018, pp. 121–133, https://doi.org/10.1007/978-3-319-94370-1_9.

[12] M. Atieh, O. Mohammad, A. Sabra, and N. Rmayti, "IOT, Deep Learning and Cybersecurity in Smart Homes: A Survey," in

Cybersecurity in Smart Homes: Architectures, Solutions and Technologies, New York, NY, USA: Wiley, 2022, pp. 203–244.

[13] S. Mittal and S. Umesh, "A survey On hardware accelerators and optimization techniques for RNNs," *Journal of Systems Architecture*, vol. 112, Jan. 2021, Art. no. 101839, https://doi.org/10.1016/j.sysarc.2020.101839.

[14] A. Azarmi Gilan, M. Emad, and B. Alizadeh, "FPGA-Based Implementation of a Real-Time Object Recognition System Using Convolutional Neural Network," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 4, pp. 755–759, Apr. 2020, https://doi.org/10.1109/TCSII.2019.2922372.

[15] T. Wang, C. Wang, X. Zhou, and H. Chen, "An Overview of FPGA Based Deep Learning Accelerators: Challenges and Opportunities," in *IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, Zhangjiajie, China, Aug. 2019, pp. 1674–1681, https://doi.org/10.1109/HPCC/SmartCity/DSS.2019.00229.

[16] M. Nobari and H. Jahanirad, "FPGA-based implementation of deep neural network using stochastic computing," *Applied Soft Computing*, vol. 137, Apr. 2023, Art. no. 110166, https://doi.org/10.1016/j.asoc.2023.110166.

[17] S. Han, H. Mao, and W. J. Dally, "Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding." arXiv, Feb. 15, 2016, https://doi.org/10.48550/arXiv.1510.00149.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.

[19] S. Zhang *et al.*, "Deep Learning in Human Activity Recognition with Wearable Sensors: A Review on Advances," *Sensors*, vol. 22, no. 4, Jan. 2022, Art. no. 1476, https://doi.org/10.3390/s22041476.

[20] T. Kryjak, M. Komorkiewicz, and M. Gorgon, "Real-time moving object detection for video surveillance system in FPGA," in *Conference on Design & Architectures for Signal & Image Processing*, Tampere, Finland, Nov. 2011, pp. 1–8, https://doi.org/10.1109/DASIP.2011.6136881.

[21] Y. Wang *et al.*, "An FPGA-based online reconfigurable CNN edge computing device for object detection," *Microelectronics Journal*, vol. 137, Jul. 2023, Art. no. 105805, https://doi.org/10.1016/j.mejo.2023.105805.

[22] M. Lin, Q. Chen, and S. Yan, "Network In Network." arXiv, Mar. 04, 2014, https://doi.org/10.48550/arXiv.1312.4400.

[23] A. Shawahna, S. M. Sait, and A. El-Maleh, "FPGA-Based Accelerators of Deep Learning Networks for Learning and Classification: A Review," *IEEE Access*, vol. 7, pp. 7823–7859, Jan. 2019, https://doi.org/10.1109/ACCESS.2018.2890150.

[24] A. Suresh, B. N. Reddy, and C. Renu Madhavi, "Hardware Accelerators for Edge Enabled Machine Learning," in *IEEE REGION 10 CONFERENCE*, Osaka, Japan, Nov. 2020, pp. 409–413, https://doi.org/10.1109/TENCON50793.2020.9293918.

[25] X. Yang, C. Zhuang, W. Feng, Z. Yang, and Q. Wang, "FPGA Implementation of a Deep Learning Acceleration Core Architecture for Image Target Detection," *Applied Sciences*, vol. 13, no. 7, Jan. 2023, Art. no. 4144, https://doi.org/10.3390/app13074144.

[26] Y. Yan, Y. Zhang, J. Zhang, and R. Liu, "Target detection algorithm based on CNN and its FPGA implementation," in *5th International Conference on Innovation in Artificial Intelligence*, Xia men, China, Mar. 2021, pp. 99–102, https://doi.org/10.1145/3461353.3461385.

[27] T. Chen, X. Li, H. Li, and G. Zhu, "Deep learning-based fall detection using commodity Wi-Fi," *Journal of Information and Intelligence*, vol. 2, no. 4, pp. 355–364, Jul. 2024, https://doi.org/10.1016/j.jiixd.2024.04.001.

[28] B. E. Sabir, M. Youssfi, O. Bouattane, and H. Allali, "Towards a New Model to Secure IoT-based Smart Home Mobile Agents using Blockchain Technology," *Engineering, Technology & Applied Science Research*, vol. 10, no. 2, pp. 5441–5447, Apr. 2020, https://doi.org/10.48084/etasr.3394.

[29] V. Tiwari, A. Keskar, and N. C. Shivaprakash, "Design of an IoT Enabled Local Network Based Home Monitoring System with a Priority Scheme," *Engineering, Technology & Applied Science Research*, vol. 7, no. 2, pp. 1464–1472, Apr. 2017, https://doi.org/10.48084/etasr.1033.

[30] T. Alshammari, "Using Artificial Neural Networks with GridSearchCV for Predicting Indoor Temperature in a Smart Home," *Engineering, Technology & Applied Science Research*, vol. 14, no. 2, pp. 13437–13443, Apr. 2024, https://doi.org/10.48084/etasr.7008.

[31] T. Ashfaq *et al.*, "A Machine Learning and Blockchain Based Efficient Fraud Detection Mechanism," *Sensors*, vol. 22, no. 19, Jan. 2022, Art. no. 7162, https://doi.org/10.3390/s22197162.

[32] H. A. Ibrahim, A. T. Azar, Z. F. Ibrahim, and H. H. Ammar, "A Hybrid Deep Learning Based Autonomous Vehicle Navigation and Obstacles Avoidance," in *The International Conference on Artificial Intelligence and Computer Vision*, Cairo, Egypt, Apr. 2020, pp. 296–307, https://doi.org/10.1007/978-3-030-44289-7_28.

[33] R. Shahzad *et al.*, "Multi-Agent System for Students Cognitive Assessment in E-Learning Environment," *IEEE Access*, vol. 12, pp. 15458–15467, Jan. 2024, https://doi.org/10.1109/ACCESS.2024.3356613.

[34] S. M. Shah *et al.*, "A Hybridized Optimal Algorithm for Multimodal Optimal Design of Inverse Problems," *IEEE Access*, vol. 11, pp. 125159–125170, Jan. 2023, https://doi.org/10.1109/ACCESS.2023.3329749.

[35] A. U. R. Butt *et al.*, "An Optimized Role-Based Access Control Using Trust Mechanism in E-Health Cloud Environment," *IEEE Access*, vol. 11, pp. 138813–138826, Jan. 2023, https://doi.org/10.1109/ACCESS.2023.3335984.