Identifying Internet of Things Devices through Unique Digital Signatures and Advanced Machine Learning Techniques

Taiwo Abdulahi Akintayo [1], **Richards Obada Okiemute** [2], **Moyosore Celestina Owoeye** [3], **Oluwaseyi Sulaimon Balogun** [4], **Chadi Paul** [5], **Madumere Chiamaka Queenet** [6], **Ruth Onyekachi Okereke** [7], **Richie Chukwunalu Moluno** [2], **Adedokun Seyi Adediran** [8], **Christian Chukwuemeka Nzeanorue** [9], **Egenuka Rhoda Ngozi** [6], **Chika Moses Madukwe** [10]

[1] *National Centre of Artificial Intelligence and Robotics*
No 28, Port Harcourt Crescent, Off Gimbiya Street, P. M. B 564, Area 11, Garki, Abuja, Nigeria

[2] *University of Benin*
P. M. B. 1154, Ugbowo, Benin City, Edo State, Nigeria

[3] *Federal University Oye-Ekiti*
Oye-Are Road, Oye Ekiti, Ekiti State, Nigeria

[4] *Tai Solarin University of Education*
P.M.B. 2118, Ijagun, Ijebu Ode, Ogun State, Nigeria

[5] *The Federal University of Technology Owerri*
P. M. B. 1526, Ihiagwa, Owerri, Nigeria

[6] *Federal Polytechnic Nekede Owerri*
Nekede Ihiagwa Road, Nekede, P. M. B. 1036, Owerri, Imo State, Nigeria

[7] *National Open University of Nigeria*
Plot 91, Cadastral Zone, Nnamdi Azikwe Expressway, Jabi, Abuja, Nigeria

[8] *Ladoke Akintola University of Technology*
P. M. B. 4000, Ogbomoso, Oyo State, Nigeria

[9] *The George Washington University*
1918 F Street, NW, Washington, DC 20052, USA

[10] *Federal Polytechnic Idah*
P. M. B. 1037, Idah Kogi State, Nigeria

DOI: 10.22178/pos.106-5

LCC Subject Category: T58.5-58.64

Received 25.06.2024
Accepted 28.07.2024
Published online 31.07.2024

Corresponding Author:
Taiwo Abdulahi Akintayo
taiwoabdulahi15@gmail.com

© 2024 The Authors. This article is licensed under a Creative Commons Attribution 4.0 License

Abstract. The rapid growth of the Internet of Things (IoT) has led to a surge in connected devices across various sectors, necessitating reliable device recognition techniques. Device fingerprinting, which involves analysing network behaviour, communication patterns, and hardware features, offers a solution. Our proposed method leverages machine learning algorithms to analyse and categorise device fingerprints, achieving exceptional accuracy in identifying diverse devices, including sensors, actuators, and intelligent appliances. Moreover, it effectively detects suspicious devices and has a low computational overhead, making it suitable for real-time deployment. Our model demonstrates its effectiveness through rigorous testing and validation on multiple IoT datasets. The benefits of device fingerprinting for IoT device identification include enhanced security, improved network management, and increased visibility into device behaviour, making it a valuable tool for IoT ecosystem management.

Keywords: IoT; Device recognition; Machine learning; Real-Time Deployment; Device Fingerprint.

INTRODUCTION

The Internet of Things (IoT) is a network of physical objects people call "things" embedded with software, electronics, networks, and sensors that allow these objects to collect and exchange data. IoT aims to extend internet connectivity from standard devices like computers, mobile, and tablets to relatively dumb devices like a toaster. IoT makes virtually everything "smart" by improving aspects of our lives with the power of data collection, AI algorithms, and networks. The IoT process starts with devices like smartphones, smartwatches, and electronic appliances like TVs and Washing machines, which help communicate with the IoT platform [1].

The rise of the Internet of Things (IoT) has led many devices connected to networks to collect and transmit data for continuous further analysis. With advancements in deep learning, many applications now use these techniques to analyse collected data, enabling "intelligence" and "automation". Consequently, leveraging data analysis and IoT infrastructure, "Smart Cities" have emerged, encompassing intelligent grids, transportation, manufacturing, and buildings [2].

As a rapidly evolving field, the Internet of Things (IoT) involves the interconnection and interaction of smart objects, i.e., IoT devices with embedded sensors, onboard data processing capabilities, and means of communication, to provide automated services that would otherwise not be possible [3]. Trillions of network-connected IoT devices will emerge in the global network around 2020 [4]. The IoT is becoming a pervasive part of everyday life, enabling a variety of emerging services and applications in cities and communities [5], including in health [6], transportation [7], energy/utilities, and other areas. Furthermore, big data analytics enable the move from the IoT to real-time control [6-9].

However, the IoT is subject to threats stemming from increased connectivity [10, 11]. For example, rogue IoT devices, defined as devices claiming a falsified identity or compromised legitimate devices, have exposed the IoT to untold risks with severe consequences. Rogue IoT devices could conduct various attacks: forging the identity of trusted entities to access sensitive resources, hijacking legitimate devices to participate in distributed denial of service (DDoS) attacks [11], etc. The problem of rogue devices becomes even more hazardous in wirelessly connected IoT, as the network traffic is more accessible to intercept, falsify, and broadcast broadly. Hence, from the perspective of network operators, the first step in securing the IoT from risks due to rogue devices is identifying known (or unknown) devices and detecting compromised ones. This survey defines Device Detection and Identification as having two perspectives: a) Identity verification of known devices and b) Detection of falsified or compromised devices.

This paper advances into the fascinating field of "IoT Device Recognition Using Device Fingerprinting", an innovative approach to effectively identifying and classifying IoT devices in a network through network packets. An effective DI system can locate devices in the network and enable the implementation of necessary security measures, such as upgrades, access restrictions, or isolating vulnerable devices. This model uses adaptable features to operate at the packet level, ensuring versatility and high detection accuracy. This sophisticated technique helps precise identification using IoT devices' unique traits and behavioural patterns, improving network security, simplifying device administration, and facilitating personalised user experiences. This paper emphasises the underlying principles and procedures of IoT device recognition utilising device fingerprinting. We explore the critical components of device fingerprinting, including data collection, feature extraction, and classification algorithms.

This study examines machine learning techniques for detecting and identifying Internet of Things (IoT) devices. It includes evaluating deep learning methods, such as recurrent and convolutional neural networks, and supervised, unsupervised, and semi-supervised learning approaches. The research also addresses challenges such as limited computational resources, energy constraints, and data privacy issues, assessing these methods' effectiveness, efficiency, and scalability. This paper introduces an innovative method for precisely identifying IoT devices using zero-bias deep learning algorithms. It tackles the problem of imbalanced data distributions that can lead to biased predictions by incorporating bias-correction layers and label-smoothing regularisation. This approach enhances its applicability across different IoT domains and industries.

METHODOLOGY

Device Fingerprinting for IoT Device Recognition: This innovative methodology creates distinctive digital fingerprints for each IoT device by analysing its unique attributes and communication behaviours. Unlike traditional methods relying on IP or MAC addresses, this approach ensures precise identification of IoT devices within a network. It employs 1) Feature Extraction and Selection To craft unique fingerprints; 2) Machine Learning Algorithms, Such as decision trees, SVM, or kNN, tailored to data characteristics and accuracy requirements; 3) Performance Evaluation To validate the methodology.

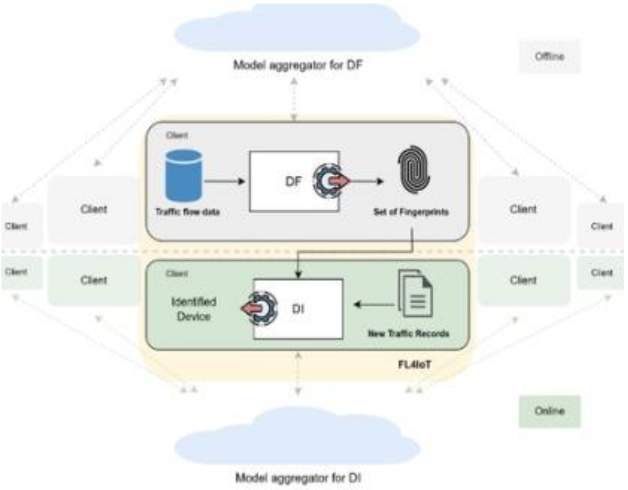


Figure 1 – Data Collection for IoT Device Recognition

The method achieves swift and accurate device recognition by following these steps, bolstering security and simplifying control within the IoT ecosystem. A comparative analysis with existing works highlights the methodology's advancements and contributions to IoT device recognition, setting a new benchmark for efficient and reliable device identification.

To gather data, the system requests network traffic information from IoT devices. For this purpose, two publicly available datasets comprising authentic device data were utilised. The datasets are 1) Aalto Dataset, Featuring 31 devices used to develop the modelling approach; 2) UNSW Dataset: This dataset contains comprehensive network traffic data, simulates IoT communication patterns and is used to assess the approach's broader applicability. IP addresses were only employed for network intrusion detection, not device identification. The larger UNSW dataset was used to evaluate the approach's generalizability, while the smaller Aalto dataset was utilised to refine the modelling approach.

The system processes the collected data, extracting features to create device fingerprints from network packets. Three methods are employed: 1) Individual Method: This method uses unique identifiers like MAC or IP addresses to differentiate devices, but it may struggle to distinguish between devices and behaviours; 2) Aggregated Method: This method groups packets with the same identifier attribute and individual packet ML labels, merging them to enhance accuracy and disregard mislabeled packets; 3) Mixed Method: Combines individual labels for exceptions, addressing issues with the aggregation process, such as MAC address transfer issues that may cause merging problems.

These fingerprinting strategies offer a novel approach to categorising network packets, but it's crucial to consider their implications in various network contexts to ensure accurate device recognition.

Feature Extraction. Features are extracted from packet headers in packet capture files, focusing on network traffic analysis. Before this, the PCAP files were separated into distinct training and testing sets to ensure complete isolation. With 20 sessions per device, the Aalto dataset is divided into two subsets: 16 for training and 4 for testing, following an 80% training and 20% testing distribution. Similarly, training and testing data for the UNSW dataset are chosen from several days' worth of captures. One hundred eleven features were taken from the packet headers and added to the payload entropy, source-destination port classes, and protocol information from the TCP-IP layers, all of which were shown to be helpful in earlier studies. Details about payload characteristics are provided by payload entropy, while summaries of source and destination ports are supplied by protocol and port class features. Since MAC and IP addresses uniquely identify devices but do not provide information about behaviour, they are purposefully removed as features.

Feature Selection. Once extracted features are extracted, they must be selected for the device fingerprint. This step identifies the most relevant and distinctive features significantly contributing to device identification. It also helps reduce computational overhead and enhance the efficiency of the following process. The initial feature pool obtained is refined using a voting method using the verse package. This method uses six scoring algorithms to evaluate the importance of features across devices and relies on user voting to decide which features should be included.

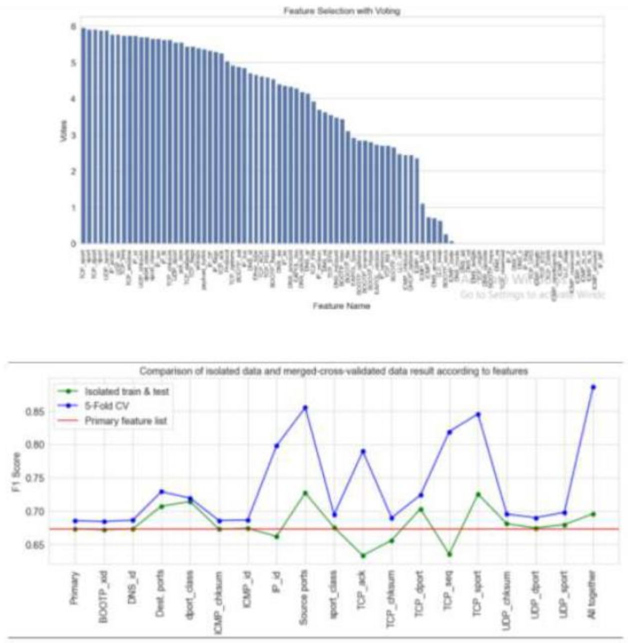
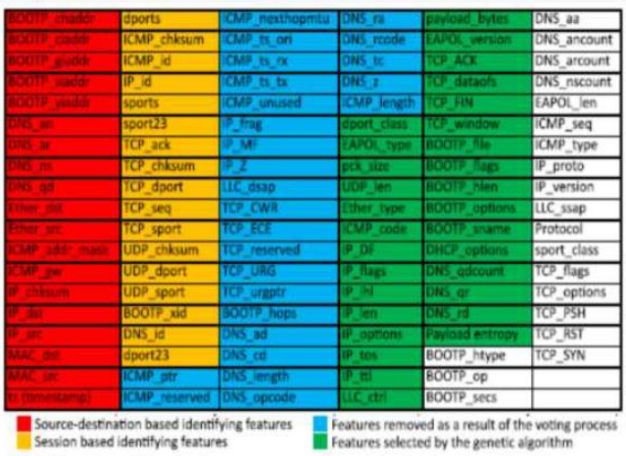


Figure 2 –Feature Selection

Additionally, we eliminated 26 features because none of the devices received any votes for them. Moreover, we discovered that characteristics based on ports showed potential for generalising models. While a few port numbers were generalised among several device instances, most appeared to be session-specific. Discrete values representing port classes and protocols were mapped to raw port-based feature values. After removing redundant and identifying characteristics, a Genetic Algorithm (GA) was employed to choose the best feature subset from the remaining pool. Acting as a wrapper method, the GA used a Decision Tree (DT) classifier to verify the utility of the feature set, which resulted in a refined feature subset that improved detection performance and decreased model complexity.



BOOTP_chaddr	dports	ICMP_nexthopmtu	DNS_ra	payload_bytes	DNS_aa
BOOTP_ciaddr	ICMP_chksum	ICMP_ts_ori	DNS_rcode	EAPOL_version	DNS_ancount
BOOTP_giaddr	ICMP_id	ICMP_ts_rx	DNS_tc	TCP_ACK	DNS_arcount
BOOTP_siaddr	IP_id	ICMP_ts_tx	DNS_z	TCP_dataofs	DNS_nscount
BOOTP_yiaddr	sports	ICMP_unused	ICMP_length	TCP_FIN	EAPOL_len
DNS_an	sport23	IP_frag	dport_class	TCP_window	ICMP_seq
DNS_ar	TCP_ack	IP_MF	EAPOL_type	BOOTP_file	ICMP_type
DNS_ns	TCP_chksum	IP_Z	pck_size	BOOTP_flags	IP_proto
DNS_qd	TCP_dport	LLC_dsap	UDP_len	BOOTP_hlen	IP_version
Ether_dst	TCP_seq	TCP_CWR	Ether_type	BOOTP_options	LLC_ssap
Ether_src	TCP_sport	TCP_ECE	ICMP_code	BOOTP_sname	Protocol
ICMP_addr_mask	UDP_chksum	TCP_reserved	IP_DF	DHCP_options	sport_class
ICMP_gw	UDP_dport	TCP_URG	IP_flags	DNS_qdcount	TCP_flags
IP_chksum	UDP_sport	TCP_urgptr	IP_ihl	DNS_qr	TCP_options
IP_dst	BOOTP_xid	BOOTP_hops	IP_len	DNS_rd	TCP_PSH
IP_src	DNS_id	DNS_ad	IP_options	Payload entropy	TCP_RST
MAC_dst	dport23	DNS_cd	IP_tos	BOOTP_htype	TCP_SYN
MAC_src	ICMP_ptr	DNS_length	IP_ttl	BOOTP_op	
ts (timestamp)	ICMP_reserved	DNS_opcode	LLC_ctrl	BOOTP_secs	

🟥 Source-destination based identifying features
🟨 Session based identifying features
🟦 Features removed as a result of the voting process
🟩 Features selected by the genetic algorithm

Figure 3 – Algorithm Selection

Algorithm selection. This work explores machine learning (ML) algorithms for predicting device types from extracted features. The Aalto data set evaluates six algorithms: RF, kNN, GB, DT, NB, and SVM. The study tunes hyperparameters for each algorithm using random search and nested cross-validation. RF, DT, and GB emerge as the top device identification (DI) performers. DT and NB are the fastest in inference time, but NB's accuracy is notably low. Despite kNN and GB's decent accuracy, their slow processing speeds render them impractical for real-time usage.

Similarly, SVM's speed and accuracy could be improved. DT is chosen as the most suitable algorithm for further investigation. Its blend of speed and accuracy makes it well-suited for real-time device detection systems operating in fast-paced network traffic scenarios.

RESULTS AND DISCUSSION

The results are shown for three different versions of the method – individual, aggregated, and mixed – depending on the chosen model. Initially, we looked into how the context of the aggregation algorithm affected the group size vs performance relationship. A positive correlation in Figure 1 indicates that larger groups are more effective. Larger group sizes may not always be feasible, though many IoT devices communicate occasionally. Considering this, a group size of 13 was selected, roughly at which performance begins to level off.

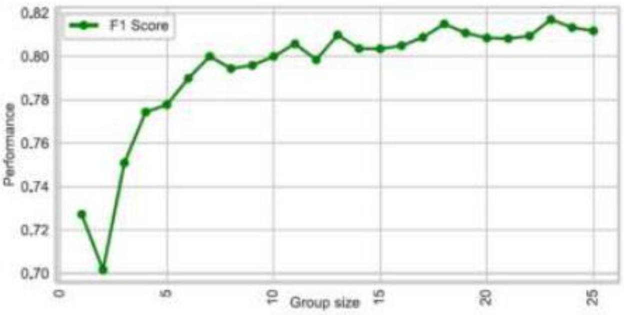
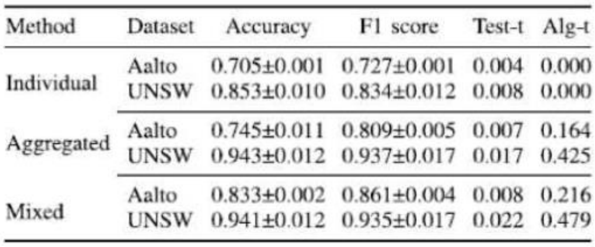


Figure 4 - Result Analysis

Within the UNSW and Aalto datasets, an F1 score of 73% and 83% was obtained using the individual packet technique. Aggregation significantly improved both datasets' capacity for accurately identifying devices. The overall F1 score increased to 81% on the Aalto dataset and nearly 94% on the UNSW dataset. Given that feature extraction on the UNSW dataset was limited to the Aalto dataset, it is highly intriguing to witness such a high degree of discrimination. This indicates that the selected feature set is highly adaptable to various IoT environments. The table shows the average device-level discrimination performance over the DT models for the Aalto dataset. Table 1 illustrates how highly unbalanced the data is.

Table 1 – Unbalanced dataset

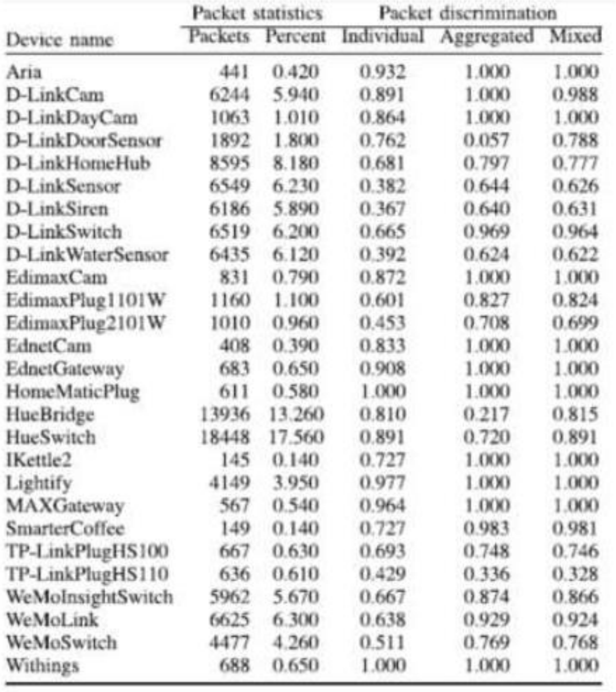


Method	Dataset	Accuracy	F1 score	Test-t	Alg-t
Individual	Aalto	0.705±0.001	0.727±0.001	0.004	0.000
	UNSW	0.853±0.010	0.834±0.012	0.008	0.000
Aggregated	Aalto	0.745±0.011	0.809±0.005	0.007	0.164
	UNSW	0.943±0.012	0.937±0.017	0.017	0.425
Mixed	Aalto	0.833±0.002	0.861±0.004	0.008	0.216
	UNSW	0.941±0.012	0.935±0.017	0.022	0.479

We must select a relatively insensitive metric to class size imbalance to provide a meaningful picture of the model's performance at the device level. This is why we use the F1 scores. This has previously been done with accuracy, a metric that needs to be more suitable for unbalanced datasets.

The device-based results of the Aalto data set show that almost all devices benefit from the aggregation algorithm, but four are negatively impacted.

Table 2 – Device proportions over the complete dataset



Device name	Packet statistics		Packet discrimination		
	Packets	Percent	Individual	Aggregated	Mixed
Aria	441	0.420	0.932	1.000	1.000
D-LinkCam	6244	5.940	0.891	1.000	0.988
D-LinkDayCam	1063	1.010	0.864	1.000	1.000
D-LinkDoorSensor	1892	1.800	0.762	0.057	0.788
D-LinkHomeHub	8595	8.180	0.681	0.797	0.777
D-LinkSensor	6549	6.230	0.382	0.644	0.626
D-LinkSiren	6186	5.890	0.367	0.640	0.631
D-LinkSwitch	6519	6.200	0.665	0.969	0.964
D-LinkWaterSensor	6435	6.120	0.392	0.624	0.622
EdimaxCam	831	0.790	0.872	1.000	1.000
EdimaxPlug1101W	1160	1.100	0.601	0.827	0.824
EdimaxPlug2101W	1010	0.960	0.453	0.708	0.699
EdnetCam	408	0.390	0.833	1.000	1.000
EdnetGateway	683	0.650	0.908	1.000	1.000
HomeMaticPlug	611	0.580	1.000	1.000	1.000
HueBridge	13936	13.260	0.810	0.217	0.815
HueSwitch	18448	17.560	0.891	0.720	0.891
IKettle2	145	0.140	0.727	1.000	1.000
Lightify	4149	3.950	0.977	1.000	1.000
MAXGateway	567	0.540	0.964	1.000	1.000
SmarterCoffee	149	0.140	0.727	0.983	0.981
TP-LinkPlugHS100	667	0.630	0.693	0.748	0.746
TP-LinkPlugHS110	636	0.610	0.429	0.336	0.328
WeMoInsightSwitch	5962	5.670	0.667	0.874	0.866
WeMoLink	6625	6.300	0.638	0.929	0.924
WeMoSwitch	4477	4.260	0.511	0.769	0.768
Withings	688	0.650	1.000	1.000	1.000

The transfer problem affects the pairing of these four devices (they share the same MAC address). A hybrid method of adding an exception to the aggregate mechanism was employed to address this. It can be observed from Table 2 that when the hybrid strategy was used, the total F1 score of the dataset increased from 81% to 86% on the Aalto dataset. Since the UNSW dataset has no transfer problem, this strategy has little effect on the outcome. It should be noted that the Aalto dataset outperforms the UNSW dataset significantly. It is the result of specific device groupings. A confusion matrix is shown in Figure 3, which contains only low-performance devices. In this case, the devices in the subgroup have certain things in common: either they are different models of the same product (like the red and green groups in Figure 3), or they are similar-purpose products manufactured by the same companies (like the yellow, blue, and orange groups in Figure 3). It doesn't seem possible to completely separate these devices based on their behaviour at the network level. However, they will likely employ remarkably similar hardware and software, exhibiting comparable behaviour and similarities in vulnerabilities and avoidance. Therefore, it would be possible to classify these devices together under a single label from a DI perspective. When the same is done, the accuracy of the Aalto dataset rises from 73% to 100%. In the context of phishing detection, the streamlined architecture adopted here facilitates efficient training and skip connections play a key role in preserving spatial information. This approach, which achieves good results with fewer parameters, suggests a promising avenue for future research in developing resource-efficient models for phishing detection. The exploration of modules capturing spatial and semantic features, alongside investigating alternative network architectures beyond standard CNNs, is encouraged to enhance overall detection performance. Emphasising resource-saving strategies will be crucial for scaling models and effectively countering evolving phishing threats.

CONCLUSIONS

Device fingerprinting is a powerful tool for identifying IoT devices. It can track devices, restrict unauthorised access, and enhance network security. Various methods for generating device fingerprints include statistical analysis, machine learning, and deep learning. The accuracy of device fingerprinting depends on the technique used and the data available. Device fingerprinting for IoT security has gained popularity in recent years. This is due to the proliferation of IoT devices and the sophistication of cyberattacks. Using device fingerprints ensures a robust and scalable recognition process, facilitating seamless integration and secure management of the interconnected IoT ecosystem. The system's real-time identification capabilities give customers quick access to device-specific data while strengthening security protocols and effectively allocating network resources. The methodology's adaptability and ability to handle class imbalances and evolving IoT environments highlight its potential for real-world implementation. By providing an autonomous and reliable device recognition solution, IoT Device Recognition using Device Fingerprinting strengthens the foundation for building smarter, interconnected environments in diverse domains, including industrial automation, healthcare, smart cities, etc. In summary, IoT device recognition utilising device fingerprinting has enormous potential to change the IoT landscape by building more interconnected, secure, and effective ecosystems.

REFERENCES

1. Taiwo, A. A., John, B. O., Sanusi, H., Inaolaji, F. A., Olasunkanmi, U. G., Azeez, A. I., Tajudeen, W. A., Akindele, A. E., Christian, Ch. N., Samuel, A. O., & Olaoluwa, J. A. (2024). Internet of things weather monitoring system. *World Journal of Advanced Research and Reviews, 22*(2), 2099–2110. doi: 10.30574/wjarr.2024.22.2.1647

2. Taiwo, A. A., Nzeanorue, C. C., Olanrewaju, S. A., Ajiboye, Q. O., Idowu, A. A. Hakeem, S., Nzeanorue, C. G., Agba, J. C., Dayo, F. P., Enabulele, E. C., Stephen, V. I., Oyesanya, A., Ogbe, M. I., & Olusola, R. A. (2024). Intelligent transportation system leveraging Internet of Things (IoT) Technology for optimised traffic flow and smart urban mobility management. *World Journal of Advanced Research and Reviews, 22*(3), 1509–1517. doi: 10.30574/wjarr.2024.22.3.1886

3. Industrial Internet of Things. (2017). In S. Jeschke, C. Brecher, H. Song, & D. B. Rawat (Eds.), *Springer Series in Wireless Technology*. Springer International Publishing. doi: 10.1007/978-3-319-42559-7

4. Sun, Y., Song, H., Jara, A. J., & Bie, R. (2016). Internet of Things and Big Data Analytics for Smart and Connected Communities. *IEEE Access, 4*, 766–773. doi: 10.1109/access.2016.2529723

5. Song, H., Srinivasan, R., Sookoor, T., & Jeschke, S. (2017). *Smart Cities: Foundations, Principles and Applications*. Hoboken: Wiley.

6. Zhang, Y., Sun, L., Song, H., & Cao, X. (2014). Ubiquitous WSN for Healthcare: Recent Advances and Future Prospects. *IEEE Internet of Things Journal, 1*(4), 311–318. doi: 10.1109/jiot.2014.2329462

7. Secure and Trustworthy Transportation Cyber-Physical Systems. (2017). In Y. Sun & H. Song (Eds.), *Springer Briefs in Computer Science*. Springer Singapore. doi: 10.1007/978-981-10-3892-1

8. Dartmann, G., Song, H., Schmeink. (2019). *Big Data Analytics for Cyber-Physical Systems*. doi: 10.1016/c2018-0-00208-x

9. Jiang, Y., Liu, Y., Liu, D., & Song, H. (2020). Applying Machine Learning to Aviation Big Data for Flight Delay Prediction. *2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*. doi: 10.1109/dasc-picom-cbdcom-cyberscitech49142.2020.00114

10. Song, H., Fink, G., & Jeschke, S. (2017). *Security and Privacy in Cyber-Physical Systems: Foundations, Principles and Applications*. Chichester: Wiley-IEEE Press.

11. Butun, I., Osterberg, P., & Song, H. (2020). Security of the Internet of Things: Vulnerabilities, Attacks, and Countermeasures. *IEEE Communications Surveys & Tutorials, 22*(1), 616–644. doi: 10.1109/comst.2019.2953364

12. Liu, Y., Wang, J., Li, J., Niu, S., & Song, H. (2022). Machine Learning for the Detection and Identification of Internet of Things Devices: A Survey. *IEEE Internet of Things Journal, 9*(1), 298–320. doi: 10.1109/jiot.2021.3099028

13. Oracle. (n. d.). What is IoT? Retrieved from https://www.oracle.com/internet-of-things/

14. Liu, Y., Wang, J., Li, J., Song, H., Yang, T., Niu, S., & Ming, Z. (2021). Zero-Bias Deep Learning for Accurate Identification of Internet-of-Things (IoT) Devices. *IEEE Internet of Things Journal, 8*(4), 2627–2634. doi: 10.1109/jiot.2020.3018677

15. Fan, L., Zhang, S., Wu, Y., Wang, Z., Duan, C., Li, J., & Yang, J. (2020). An IoT Device Identification Method based on Semi-supervised Learning. *2020 16th International Conference on Network and Service Management (CNSM)*. doi: 10.23919/cnsm50824.2020.9269044

16. Aneja, S., Aneja, N., & Islam, M. S. (2018). IoT Device Fingerprint using Deep Learning. *2018 IEEE International Conference on Internet of Things and Intelligence System (IOTAIS)*. doi: 10.1109/iotais.2018.8600824

17. Aksoy, A., & Gunes, M. H. (2019). Automated IoT Device Identification using Network Traffic. *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. doi: 10.1109/icc.2019.8761559

18. Marchal, S., Miettinen, M., Nguyen, T. D., Sadeghi, A.-R., & Asokan, N. (2019). AuDI: Toward Autonomous IoT Device-Type Identification Using Periodic Communication. *IEEE Journal on Selected Areas in Communications, 37*(6), 1402–1412. doi: 10.1109/jsac.2019.2904364

19. Duan, C., Gao, H., Song, G., Yang, J., & Wang, Z. (2022). ByteIoT: A Practical IoT Device Identification System Based on Packet Length Distribution. *IEEE Transactions on Network and Service Management, 19*(2), 1717–1728. doi: 10.1109/tnsm.2021.3130312

20. Adnan Ferman, V., & Ali Tawfeeq, M. (2022). Early Generation and Detection of Efficient IoT Device Fingerprints Using Machine Learning. *International Journal on Advanced Science, Engineering and Information Technology, 12*(1), 53. doi: 10.18517/ijaseit.12.1.14349