Designing Efficient model for ModelNet10 Database and Keras for 3D Object Recognition by PointNet Architecture

Aryaman Anand, Dr. Surjeet Dalal and Dr. Priyanka Dubey

Amity School of Engineering and Technology, Amity University Haryana, India; aryaman2023@gmail.com
Amity School of Engineering and Technology, Amity University Haryana, India; profsurjeetdalal@gmail.com
Amity School of Engineering and Technology, Amity University Haryana, India; pdubey@ggn.amity.edu

***Corresponding Author:** Aryaman Anand. *aryaman.anand53@gmail.com*

Abstract

This study examines the potential and usefulness of teaching robots to administer contactless medical care using the ModelNet10 database. Robotics and 3D object identification technology present intriguing solutions to safeguard patient and staff safety as the need for contactless care in medical environments increases. The two goals of this study are to first determine whether the ModelNet10 dataset can be used to train robots in contactless treatment scenarios, and then to develop and assess algorithms and methodologies that make use of the dataset for jobs like 3D object recognition, autonomous navigation, contactless patient monitoring, and medical supply delivery. The research technique includes a thorough examination of the available literature, algorithm creation, and experimental assessment. This study's conclusions have important ramifications for contactless medical treatment, including improved healthcare systems with precise 3D object identification, autonomous navigation, and contactless patient monitoring. In order to provide secure and efficient patient care in a post-pandemic world, our research makes use of cutting-edge technologies and addresses new obstacles.

Keywords

ModelNet10, 3D Object Recognition, MeshLab, PointNet, Separated by Commas

1. Introduction

This research paper emphasizes the importance of contactless treatment in medical settings, particularly during the COVID-19 pandemic. Robotics and 3D object recognition technologies offer promising solutions to address this need. The study aims to explore the practical implications of training robots in contactless medical environments using the ModelNet10 Princeton 3D database. The research has two objectives: to investigate the applicability of the ModelNet10 dataset for training robots in contactless medical treatment and to develop and evaluate algorithms for tasks such as 3D object recognition, autonomous navigation, contactless patient monitoring, and medical supplies delivery. The methodology includes a literature review, algorithm development, and implementation. The performance of the trained robot models will be evaluated using quantitative measures in real-world medical scenarios.

The findings have significant implications for contactless medical treatment. Successful utilization of the ModelNet10 dataset can enhance healthcare systems by providing accurate and efficient 3D object recognition, autonomous navigation, and contactless patient monitoring. This minimizes the risk of disease transmission and ensures the safety of patients and healthcare providers. The insights gained from this study will guide future research and contribute to the development of advanced robotic systems for contactless medical treatment, addressing emerging challenges in healthcare delivery.

2. Working

This project utilizes the PointNet deep learning model to categorize, locate, and segment 3D point clouds. The code begins by setting up dependencies and importing libraries. The ModelNet10 dataset is used, and its mesh files are read and visualized. The dataset is parsed, and the points and labels are stored separately. Data augmentation is applied to the training set, and the dataset is batched and shuffled. The PointNet model architecture is described using convolutional and fully connected layers, along with batch normalization and ReLU activation. The model is trained using the training dataset and monitored using validation data. After training, the predictions of the model are visualized using matplotlib. The code demonstrates the implementation of PointNet for point cloud analysis using the ModelNet10 dataset, covering data loading, preprocessing, model definition, training, and prediction visualization.

3. Results

3.1 20 Epoch Simulation

For the first simulation, this model was trained for 20 epochs on the ModelNet10 Database. The results are as follows.

Table 1 Epoch simultaion

Epoch	Training Loss	Training Accuracy	Validation Accuracy
1	3.5611	0.2836	0.2037
2	2.9869	0.4082	0.3855
3	2.717	0.4981	0.4813
4	2.6063	0.5512	0.4857
5	2.4069	0.5996	0.5661
6	2.369	0.6014	0.5275
7	2.2368	0.6487	0.6002
8	2.0797	0.6828	0.4174
9	2.056	0.6971	0.6388
10	1.8829	0.7407	0.2599
11	1.8964	0.7447	0.7445
12	1.7828	0.7745	0.7841
13	1.7929	0.7725	0.6773
14	1.6985	0.8033	0.5749
15	1.7114	0.8051	0.8128
16	1.6671	0.8213	0.7643
17	1.6954	0.8021	0.7731
18	1.6054	0.8384	0.8216
19	1.7019	0.8088	0.783
20	1.5749	0.8484	0.7930

Average Training Loss: 2.22006
Average Training Accuracy: 0.69071
Average Validation Accuracy: 0.61799

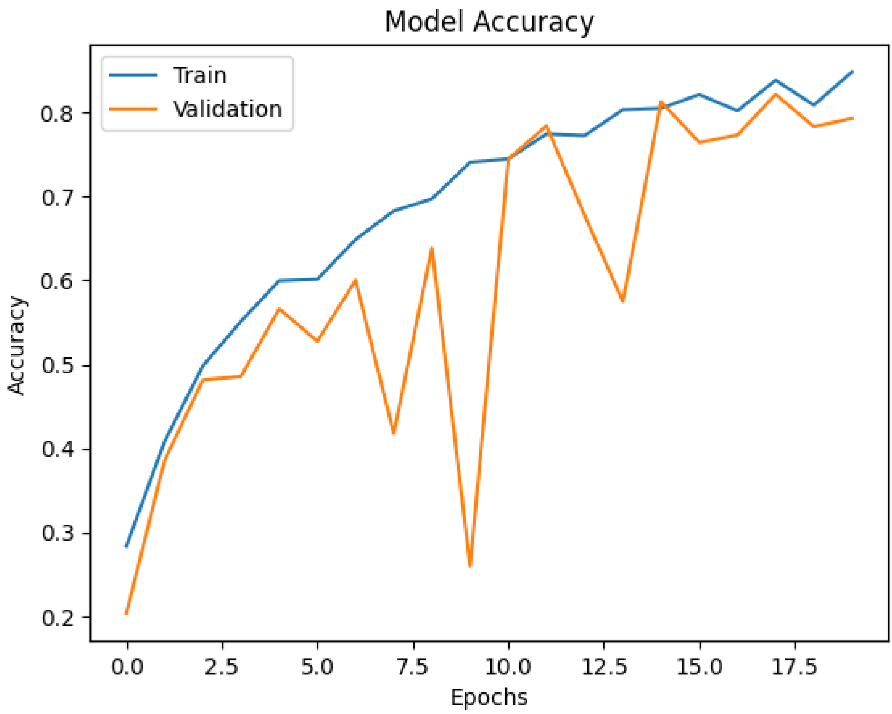


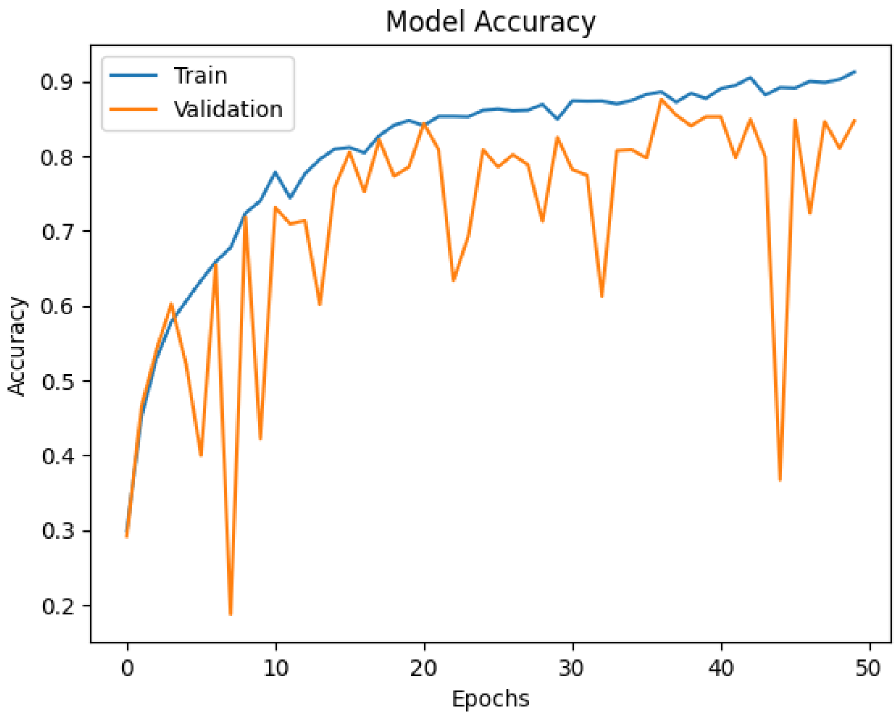
Fig 1. Model Accuracy

3.1 50 Epoch Simulation

For the second simulation, this model was trained for 50 epochs on the ModelNet10 Database. The results are as follows.

Epoch	Loss	Training Accuracy	Validation Accuracy
1	3.5365	0.2962	0.1872
2	3.0191	0.3786	0.2985
3	2.9364	0.4092	0.4075
4	2.8519	0.448	0.3645
5	2.6472	0.4841	0.4328
6	2.5089	0.5452	0.3612
7	2.4339	0.5783	0.489
8	2.2287	0.6412	0.3282
9	2.1663	0.6625	0.5573
10	2.0716	0.686	0.2467
11	2.0094	0.7106	0.739
12	1.9181	0.7286	0.7236
13	1.856	0.7557	0.7489
14	1.7884	0.7765	0.6685
15	1.7682	0.7933	0.7974

16	1.7779	0.7858	0.7952
17	1.7406	0.7883	0.7566
18	1.6468	0.8148	0.7941
19	1.6566	0.8173	0.7797
20	1.6082	0.8276	0.8271
21	1.5623	0.8419	0.8249
22	1.6283	0.8289	0.8073
23	1.6011	0.8286	0.6509
24	1.5753	0.8414	0.7225
25	1.5481	0.8467	0.8315
26	1.5056	0.8634	0.8304
27	1.5095	0.8604	0.8359
28	1.5222	0.8529	0.761
29	1.5002	0.8614	0.674
30	1.5294	0.8579	0.7577
31	1.4908	0.8677	0.7026
32	1.4764	0.8712	0.8051
33	1.4609	0.875	0.6346
34	1.4698	0.8715	0.8117
35	1.55	0.8547	0.7952
36	1.5016	0.8637	0.8557
37	1.4314	0.8882	0.8612
38	1.4563	0.8717	0.8458
39	1.4326	0.8792	0.8106
40	1.4669	0.8752	0.7963
41	1.4164	0.8812	0.8689
42	1.4006	0.8968	0.8304
43	1.4191	0.891	0.8447
44	1.392	0.896	0.7313
45	1.4188	0.886	0.7163
46	1.4131	0.892	0.8271
47	1.4023	0.8928	0.7192
48	1.374	0.9018	0.8106
49	1.4402	0.8835	0.7324
50	1.4164	0.8887	0.8161



Average Loss: 1.72509
Average Training Accuracy: 0.79576
Average Validation Accuracy: 0.70899

3.1 100 Epoch Simulation

For the third simulation, this model was trained for 100 epochs on the ModelNet10 Database. The table only contains results for the final 50 epochs. The results are as follows.

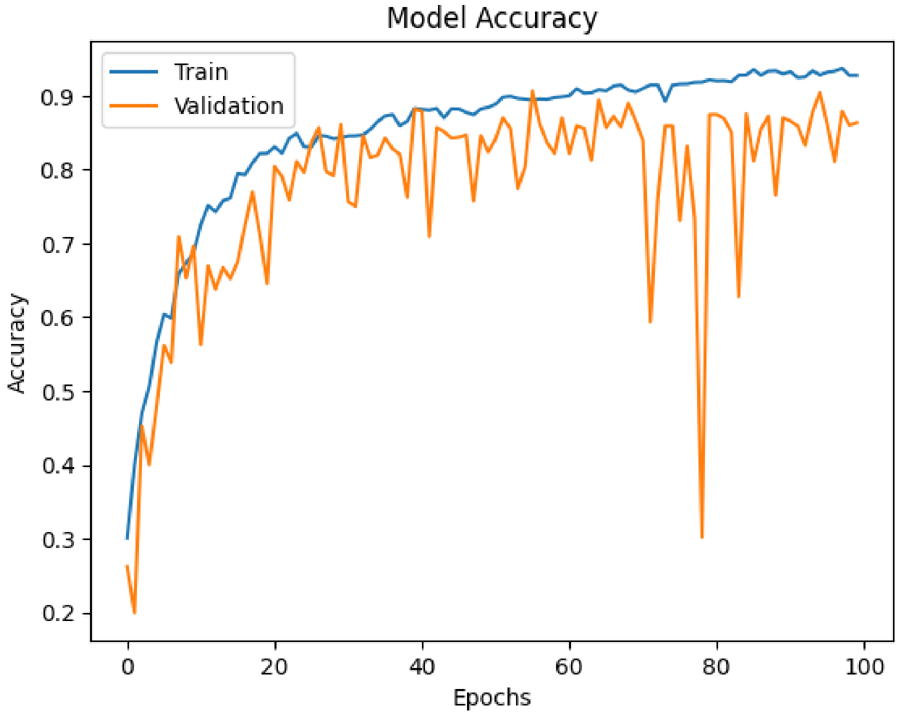
Epoch	Loss	Training Accuracy	Validation Accuracy
51	1.413	0.8885	0.8414
52	1.3817	0.898	0.87
53	1.3861	0.8993	0.8557
54	1.3717	0.8963	0.7742
55	1.3957	0.8953	0.804
56	1.3788	0.8943	0.9064
57	1.4067	0.8955	0.8612
58	1.395	0.895	0.8359
59	1.3668	0.8975	0.8216
60	1.3731	0.8985	0.87

61	1.3638	0.9	0.8216
62	1.3454	0.9093	0.859
63	1.352	0.9035	0.8557
64	1.358	0.904	0.8128
65	1.3507	0.908	0.8943
66	1.3558	0.9065	0.8568
67	1.3334	0.9131	0.8722
68	1.3386	0.9146	0.8579
69	1.3386	0.9075	0.8899
70	1.3467	0.905	0.8656
71	1.3327	0.9098	0.8392
72	1.3147	0.9146	0.5936
73	1.3263	0.9146	0.761
74	1.3877	0.8925	0.859
75	1.3229	0.9143	0.859
76	1.3125	0.9158	0.7313
77	1.3205	0.9161	0.8315
78	1.3274	0.9181	0.7335
79	1.3004	0.9183	0.3018
80	1.3151	0.9216	0.8744
81	1.3174	0.9201	0.8744
82	1.3047	0.9203	0.8689
83	1.309	0.9188	0.8502
84	1.2847	0.9278	0.6278
85	1.2928	0.9281	0.8756
86	1.2725	0.9354	0.8117
87	1.2741	0.9278	0.8546
88	1.2626	0.9334	0.8722
89	1.2722	0.9339	0.7654
90	1.2821	0.9296	0.87
91	1.2652	0.9326	0.8656
92	1.3027	0.9246	0.859
93	1.3009	0.9256	0.8326
94	1.2595	0.9339	0.8778
95	1.2834	0.9281	0.9042
96	1.2701	0.9318	0.8612
97	1.2594	0.9334	0.8106
98	1.2586	0.9371	0.8789
99	1.2866	0.9276	0.8601

100	1.2893	0.9276	0.8634

From Epoch 51-100,

Average Loss: 1.3269
Average Training Accuracy: 0.91437
Average Validation Accuracy: 0.82685



4.Discussion

In this work, a model was trained on the ModelNet10 Database, and its performance was examined across various epochs. There were three simulations run, each with a different number of epochs: 20, 50, and 100. The outcomes of these simulations are provided and discussed in the paragraphs that follow.

Similar characteristics can be seen in the 50-epoch simulation as well. The training accuracy increased from 0.2962 to 0.8887, while the training loss fell from 3.5365 to 1.4164. Although there were some variations, the validation accuracy typically increased from 0.1872 to 0.8161. This shows that across the additional epochs, the model was still able to learn and enhance its performance. It's important to note that the training accuracy fell short of 100%, indicating that the model may not have fully converged.

The outcomes for epochs 51–100 of the simulation across 100 epochs are shown. This period's average loss was 1.3269, which was less than the simulation's 50-epoch average loss. The average training accuracy was 0.91437, which shows that the training data were predicted with good precision. The model maintained a respectable performance on unobserved data, as evidenced by the average validation accuracy of 0.82685. These findings imply that lengthier training sessions for the model enhanced its performance.

It is clear from comparing the outcomes of the various simulations that adding more epochs often resulted in lower training loss and improved training accuracy. The effect on validation accuracy, however, differed. While the validation accuracy typically increased with the number of epochs, there were certain instances where overfitting and oscillations were seen.

It is crucial to take into account this study's constraints. The ModelNet10 Database was used for the simulations, hence the findings may not apply to other datasets. Additionally, the discussion ignored other elements like processing resources or model complexity in favour of concentrating only on the performance measures of loss and accuracy.

5. Conclusion

In summary, the goal of this study was to evaluate a model's performance over various numbers of epochs after being trained on the ModelNet10 Database. The findings showed that, in general, training loss and accuracy improved as the number of epochs increased. The influence on validation accuracy, however, varied, revealing overfitting tendencies in some circumstances.

These results are significant because of what they have to offer the fields of deep learning and machine learning. This work sheds light on the trade-off between model accuracy and training time by examining the impact of epoch numbers on model performance. It emphasises the need of striking the right balance between avoiding overfitting and enhancing generalisation.

The ramifications of this research go beyond academic settings to real-world settings where model training and validation are essential. Practitioners can choose appropriate training procedures and reduce excessive computational expenses by understanding the relationship between epoch numbers and performance measures. The results also highlight the need for methods to reduce overfitting and improve model generalisation, such as regularisation and early halting. Future study could go in a number of different areas as a result of these discoveries. First, research into various model designs and optimisation techniques may reveal how they interact with epoch numbers and affect model performance. Further enhancing model generalisation may be achieved by investigating additional regularisation strategies and hyperparameter tuning procedures.

To assess the generalizability of the results, it would be useful to expand this study to more datasets and domains. To gain a deeper grasp of the trade-offs involved, compare the effects of epoch numbers across various datasets. In summary, this study adds to the body of knowledge by showing the connection between model performance and epoch numbers. The results highlight the significance of careful epoch number selection and optimisation to strike the ideal balance between training accuracy and generalisation. The area of machine learning can continue to grow and enhance the performance of deep learning models in a variety of applications by taking the implications of these findings into account and researching potential future research directions.

References

[1]. Sethi, N. Jaglan, V. Bhaskar, S. (2021) Multi-Operator based Saliency Detection. 2021 International Conference on Computational Performance Evaluation (ComPE), 507-512.

[2]. Swati, Jaglan, V. Bhaskar, S. (2021) A NOVEL MULTI GRANULARITY LOCKING SCHEME BASED ON CONCURRENT MULTI-VERSION HIERARCHICAL STRUCTURE. INFORMATION TECHNOLOGY IN INDUSTRY, 9(1), 932-947.

[3]. Sethi, N. Verma, J.K. Shrivastava, U. Bhaskar, S. (2021) Google Stock Movement: A Time Series Study Using LSTM Network. Multidisciplinary Functions of Blockchain Technology in AI and IoT Applications, 70-87.

[4]. Chaudhary, S. Jatain, A. Nagpal, P. Bhaskar, S. (2021) Design and Development of Gesture Based Gaming Console. A & V Publications, 12(2), 51-56.

[5]. Nagpal, P. Jatain, A. Chaudhary, S. Bhaskar, S. (2021) Motion detection of webcam using frame differencing method. A & V Publications, 12(2), 32-38.
[6]. Jatain, A. Chaudhary, S. Batra, P. Bhaskar, S. (2021) Rest web services: An elementary learning. Research Journal of Engineering and Technology, 12(3), 75-78.
[7]. Mor, P. Bhaskar, S. (2021) Enabling Technologies and Architecture for 5G-Enabled IoT. Blockchain for 5G-Enabled IoT: The new wave for Industrial Automation, 223-259.
[8]. Jatain, A. Chaudhary, S. Nagpal, P. Bhaskar, S. (2021) Cloud Storage Architecture: Issues, Challenges and Opportunities. International Journal of Innovative Research in Computer Science & Technology (IJIRCST) ISSN, 2347-5552.
[9]. T Joy, D. Kaur, G. Chugh, A. Bhaskar, S. (2021) Computer Vision for Color Detection. International Journal of Innovative Research in Computer Science & Technology (IJIRCST) ISSN, 2347-5552.
[10]. Jaglan, V. Bhaskar, S. (2021) Locking Paradigm in Hierarchical Structure Environment. Advances in Mechanical Engineering: Select Proceedings of CAMSE 2020, , 653-661.
[11]. Nanda, A. Gupta, S. Bhaskar, S. (2020) A Comprehensive Survey of Machine Learning in Scheduling of Transactions. 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(2020), 740-745.
[12]. Bhaskar S. Bhaskar, S. (2020) Study of locking protocols in database management for increasing concurrency. 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(2020), 556-560.
[13]. Bhaskar S. Bhaskar, S. (2020) Reducing Complexity of Graph Isomorphism Problem. International Journal of Innovative Research in Computer Science & Technology (IJIRCST) ISSN, 2347-5552.
[14]. Bhaskar, S. (2020) CGVL:An Hierarchical Locking Mechanism . International Journal of Control and Automation, 12(6), 725-743
[15]. Dalal, S., Poongodi, M., Lilhore, U. K., Dahan, F., Vaiyapuri, T., Keshta, I., ... & Simaiya, S. Optimized LightGBM model for security and privacy issues in cyber-physical systems. Transactions on Emerging Telecommunications Technologies, e4771.
[16]. Dalal, S., Manoharan, P., Lilhore, U. K., Seth, B., Simaiya, S., Hamdi, M., & Raahemifar, K. (2023). Extremely boosted neural network for more accurate multi-stage Cyber attack prediction in cloud computing environment. Journal of Cloud Computing, 12(1), 1-22.
[17]. Shetty, S., & Dalal, S. (2022, December). Bi-Directional Long Short-Term Memory Neural Networks for Music Composition. In 2022 Fourth International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT) (pp. 1-6). IEEE.
[18]. Dalal, S., Seth, B., Radulescu, M., Cilan, T. F., & Serbanescu, L. (2023). Optimized Deep Learning with Learning without Forgetting (LwF) for Weather Classification for Sustainable Transportation and Traffic Safety. Sustainability, 15(7), 6070.
[19]. Onyema, E. M., Lilhore, U. K., Saurabh, P., Dalal, S., Nwaeze, A. S., Chijindu, A. T., ... & Simaiya, S. (2023). Evaluation of IoT-Enabled hybrid model for genome sequence analysis of patients in healthcare 4.0. Measurement: Sensors, 26, 100679.
[20]. Dalal, S., Manoharan, P., Lilhore, U. K., Seth, B., Simaiya, S., Hamdi, M., & Raahemifar, K. (2023). Extremely boosted neural network for more accurate multi-stage Cyber attack prediction in cloud computing environment. Journal of Cloud Computing, 12(1), 1-22.
[21]. Dalal, S., Goel, P., Onyema, E. M., Alharbi, A., Mahmoud, A., Algarni, M. A., & Awal, H. (2023). Application of Machine Learning for Cardiovascular Disease Risk Prediction. Computational Intelligence and Neuroscience, 2023.
[22]. Dalal, S., Seth, B., Radulescu, M., Secara, C., & Tolea, C. (2022). Predicting Fraud in Financial Payment Services through Optimized Hyper-Parameter-Tuned XGBoost Model. Mathematics, 10(24), 4679.
[23]. Dalal, S., Onyema, E. M., & Malik, A. (2022). Hybrid XGBoost model with hyperparameter tuning for prediction of liver disease with better accuracy. World Journal of Gastroenterology, 28(46), 6551-6563.
[24]. Edeh, M. O., Dalal, S., Obagbuwa, I. C., Prasad, B. V. V., Ninoria, S. Z., Wajid, M. A., & Adesina, A. O. (2022). Bootstrapping random forest and CHAID for prediction of white spot disease among

shrimp farmers. Scientific Reports, 12(1), 1-12.

[25]. Zaki, J., Nayyar, A., Dalal, S., & Ali, Z. H. (2022). House price prediction using hedonic pricing model and machine learning techniques. Concurrency and Computation: Practice and Experience, 34(27), e7342.

[26]. Dalal, S., Onyema, E., Romero, C., Ndufeiya-Kumasi, L., Maryann, D., Nnedimkpa, A. & Bhatia, T. (2022). Machine learning-based forecasting of potability of drinking water through adaptive boosting model. Open Chemistry, 20(1), 816-828. https://doi.org/10.1515/chem-2022-0187

[27]. Onyema, E. M., Dalal, S., Romero, C. A. T., Seth, B., Young, P., & Wajid, M. A. (2022). Design of Intrusion Detection System based on Cyborg intelligence for security of Cloud Network Traffic of Smart Cities. Journal of Cloud Computing, 11(1), 1-20.

[28]. Dalal, S., Onyema, E. M., Kumar, P., Maryann, D. C., Roselyn, A. O., & Obichili, M. I. (2022). A Hybrid machine learning model for timely prediction of breast cancer. International Journal of Modeling, Simulation, and Scientific Computing, 2023, 1-21.

[29]. Dalal, S., Seth, B., Jaglan, V., Malik, M., Dahiya, N., Rani, U., ... & Hu, Y. C. (2022). An adaptive traffic routing approach toward load balancing and congestion control in Cloud–MANET ad hoc networks. Soft Computing, 26(11), 5377-5388.

[30]. Edeh, M. O., Dalal, S., Dhaou, I. B., Agubosim, C. C., Umoke, C. C., Richard-Nnabu, N. E., & Dahiya, N. (2022). Artificial Intelligence-Based Ensemble Learning Model for Prediction of Hepatitis C Disease. Frontiers in Public Health, 847.

[31]. Seth, B., Dalal, S., Jaglan, V., Le, D. N., Mohan, S., & Srivastava, G. (2022). Integrating encryption techniques for secure data storage in the cloud. Transactions on Emerging Telecommunications Technologies, 33(4), e4108.

[32]. Malik, M., Nandal, R., Dalal, S., Maan, U., & Le, D. N. An efficient driver behavioral pattern analysis based on fuzzy logical feature selection and classification in big data analysis. Journal of Intelligent & Fuzzy Systems, 43(3), 3283-3292