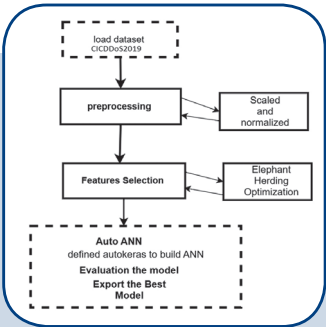
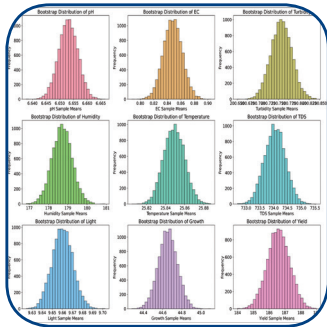
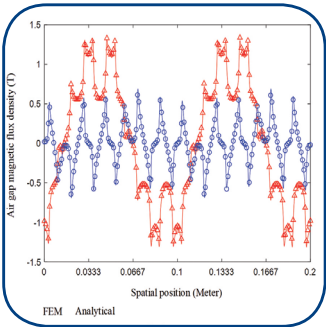
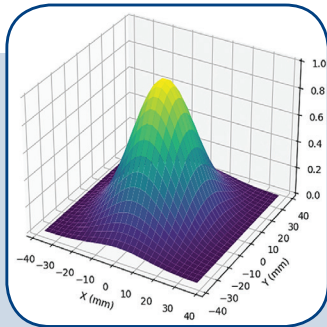




International Journal of Electrical and Computer Engineering Systems



Confusion Matrix				
True Class	BCC	BKL	MEL	NV
BCC	78	4	7	4
BKL	5	173	25	14
MEL	1	5	140	25
NV	8	21	82	798
Predicted Class				



INTERNATIONAL JOURNAL OF ELECTRICAL AND COMPUTER ENGINEERING SYSTEMS

Published by Faculty of Electrical Engineering, Computer Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia

Osijek, Croatia | Volume 16, Number 9, 2025 | Pages 641 - 705

The International Journal of Electrical and Computer Engineering Systems is published with the financial support
of the Ministry of Science and Education of the Republic of Croatia

CONTACT

**International Journal of Electrical
and Computer Engineering Systems
(IJECS)**

Faculty of Electrical Engineering, Computer
Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia
Kneza Trpimira 2b, 31000 Osijek, Croatia
Phone: +38531224600, Fax: +38531224605
e-mail: ijeces@ferit.hr

Subscription Information

The annual subscription rate is 50€ for individuals,
25€ for students and 150€ for libraries.
Giro account: 2390001 - 1100016777,
Croatian Postal Bank

EDITOR-IN-CHIEF

Tomislav Matić

J.J. Strossmayer University of Osijek,
Croatia

Goran Martinović

J.J. Strossmayer University of Osijek,
Croatia

EXECUTIVE EDITOR

Mario Vranješ

J.J. Strossmayer University of Osijek, Croatia

ASSOCIATE EDITORS

Krešimir Fekete

J.J. Strossmayer University of Osijek, Croatia

Damir Filko

J.J. Strossmayer University of Osijek, Croatia

Davor Vinko

J.J. Strossmayer University of Osijek, Croatia

EDITORIAL BOARD

Marinko Barukčić

J.J. Strossmayer University of Osijek, Croatia

Tin Benšić

J.J. Strossmayer University of Osijek, Croatia

Matjaz Colnarič

University of Maribor, Slovenia

Aura Conci

Fluminense Federal University, Brazil

Bojan Čukić

University of North Carolina at Charlotte, USA

Radu Dobrin

Mälardalen University, Sweden

Irena Galić

J.J. Strossmayer University of Osijek, Croatia

Ratko Grbić

J.J. Strossmayer University of Osijek, Croatia

Krešimir Grgić

J.J. Strossmayer University of Osijek, Croatia

Marijan Herceg

J.J. Strossmayer University of Osijek, Croatia

Darko Huljanić

Ericsson Nikola Tesla, Croatia

Željko Hocenski

J.J. Strossmayer University of Osijek, Croatia

Gordan Ježić

University of Zagreb, Croatia

Ivan Kaštelan

University of Novi Sad, Serbia

Ivan Maršić

Rutgers, The State University of New Jersey, USA

Kruno Miličević

J.J. Strossmayer University of Osijek, Croatia

Gaurav Morghare

Oriental Institute of Science and Technology,
Bhopal, India

Srete Nikolovski

J.J. Strossmayer University of Osijek, Croatia

Davor Pavuna

Swiss Federal Institute of Technology Lausanne,
Switzerland

Marjan Popov

Delft University, Nizozemska

Sasikumar Punnekkat

Mälardalen University, Sweden

Chiara Ravasio

University of Bergamo, Italija

Snježana Rimac-Drlje

J.J. Strossmayer University of Osijek, Croatia

Krešimir Romić

J.J. Strossmayer University of Osijek, Croatia

Gregor Rozinaj

Slovak University of Technology, Slovakia

Imre Rudas

Budapest Tech, Hungary

Dragan Samardžija

Nokia Bell Labs, USA

Cristina Secoleanu

Mälardalen University, Sweden

Wei Siang Hoh

Universiti Malaysia Pahang, Malaysia

Marinko Stojkov

University of Slavonski Brod, Croatia

Kannadhasan Suriyan

Cheran College of Engineering, India

Zdenko Šimić

The Paul Scherrer Institute, Switzerland

Nikola Teslić

University of Novi Sad, Serbia

Jami Venkata Suman

GMR Institute of Technology, India

Domen Verber

University of Maribor, Slovenia

Denis Vranješ

J.J. Strossmayer University of Osijek, Croatia

Bruno Zorić

J.J. Strossmayer University of Osijek, Croatia

Drago Žagar

J.J. Strossmayer University of Osijek, Croatia

Matej Žnidarec

J.J. Strossmayer University of Osijek, Croatia

Proofreader

Ivanka Ferčec

J.J. Strossmayer University of Osijek, Croatia

Editing and technical assistance

Davor Vrandečić

J.J. Strossmayer University of Osijek, Croatia

Stephen Ward

J.J. Strossmayer University of Osijek, Croatia

Dražen Bajer

J.J. Strossmayer University of Osijek, Croatia

Journal is referred in:

- Scopus
- Web of Science Core Collection
(Emerging Sources Citation Index - ESCI)
- Google Scholar
- CiteFactor
- Genamics
- Hrčak
- Ulrichweb
- Reaxys
- Embase
- Engineering Village

Bibliographic Information

Commenced in 2010.
ISSN: 1847-6996
e-ISSN: 1847-7003
Published: quarterly
Circulation: 300

IJECS online

<https://ijeces.ferit.hr>

Copyright

Authors of the International Journal of Electrical
and Computer Engineering Systems must transfer
copyright to the publisher in written form.

TABLE OF CONTENTS

Design and Simulation of Rectangular Slot Antennas Using the Finite Element Method in Python641
Original Scientific Paper
Zahraa A. Hamzar

**Analytical Approach to Predict the Magnetic Field
of Slotted Permanent Magnet Linear Machines in Open-Circuit mode651**
Original Scientific Paper
Naghi Rostami | Amjed Alwan Albordhi | Mohammad Bagher Bannae Sharifian

**Optimal Power Control Using Modified Perturb
and Observe Algorithm for Photovoltaic System Under Partial Shading.....659**
Original Scientific Paper
Afrilia Surya Andini | Ratna Ika Putri | Ika Noer Syamsiana | Gery Prasetya | Achsanul Khabib

**Distributed Approach to detect DDOS attack based
on Elephant Herding Optimization and Pipeline Artificial Neural Network.....669**
Original Scientific Paper
Yasamin Hamza Alagrash

**Lettuce Yield Prediction: ElasticNet Regression Model (ElNetRM)
for Indoor Aeroponic Vertical Farming System683**
Original Scientific Paper
Gowtham Rajendiran | Jebakumar Rethnaraj

Ensemble Deep Learning Approach For Multi-Class Skin Cancer Classification697
Original Scientific Paper
Ali Abdulameer | Raaed Hassan | Abbas Humadi

About this Journal
IJECES Copyright Transfer Form

Design and Simulation of Rectangular Slot Antennas Using the Finite Element Method in Python

Original Scientific Paper

Zahraa A. Hamza*

Department of Artificial Intelligence Engineering Techniques
Middle Technical University, Baghdad, Iraq
ZahraaAbbas@mtu.edu.iq

*Corresponding author

Abstract – The design and simulation of rectangular slot antennas using a Python-based Finite Element Method (FEM) framework are presented in this study, addressing the limitations of costly and resource-intensive commercial electromagnetic tools and the proposed open-source implementation leverages Python's computational ecosystem—integrating Gmsh for mesh generation, FEniCS for FEM discretization, and SciPy for sparse matrix solving—to provide an accessible and customizable platform for antenna analysis. Validation against Computer Simulation Technology (CST) and High Frequency Structure Simulator (HFSS) demonstrates exceptional agreement, with return loss (S11) deviations below 0.5 dB, radiation efficiencies exceeding 85%, and impedance matching within 2Ω of the target 50Ω , parametric studies reveal the impact of slot dimensions and substrate properties on resonant frequency and bandwidth, while computational benchmarks highlight Python-FEM's competitive performance, achieving solve times under 20 seconds for meshes with 180 MB memory usage and the framework's accuracy, coupled with its open-source flexibility, bridges the gap between academic research and industrial prototyping, particularly for applications in 5G, IoT, and radar systems, future enhancements, like Graphics Processing Unit (GPU) acceleration and multi-physical coupling, are proposed to further advance its scalability and versatility in next-generation antenna design.

Keywords: Rectangular slot antennas, Finite Element Method (FEM), Python-based simulation, Open-source electromagnetics, Antenna performance validation

Received: March 20, 2025; Received in revised form: June 25, 2025; Accepted: May 30, 2025

1. INTRODUCTION

Wireless communication systems have witnessed tremendous advancements in recent decades, driven by the rapid expansion of 5G networks, the proliferation of Internet of Things (IoT) devices, and the growing demand in modern radar applications, these trends have created an increasing need for compact, wideband, and low-profile antennas capable of meeting advanced performance requirements within strict constraints of size, weight, and power. Among the proposed engineering solutions, rectangular slot antennas have emerged as a strategic choice due to their inherent advantages, including ease bandwidth compared to conventional antenna designs [1], also the design and performance optimization of this class of antennas heavily rely on Accurate Electromagnetic (EM) simula-

tions, typically performed using commercial software such as Computer Simulation Technology (CST) Microwave Studio and ANSYS High Frequency Structure Simulator (HFSS).

Despite the high accuracy of these software packages, their high cost, computational complexity, and closed-source nature impose significant limitations, particularly for researchers and developers operating in resource-constrained environments or within emerging institutions, thereby restricting opportunities for innovation and rapid prototyping of new designs [2], and in recent years the Python programming language has gained significant momentum in the field of electromagnetic computation, supported by a rich open-source environment and specialized libraries such as FEniCS for Finite Element Method (FEM) modeling, SciPy for advanced numerical

computations, and NumPy for matrix data processing [3]. Nevertheless, despite some isolated successes in using Python for simulating components such as waveguides and patch antennas, the absence of a dedicated framework for analyzing and designing slot antennas—particularly rectangular slot antennas—remains a critical gap in the technical literature, this lack limits the ability of researchers and developers to customize simulation models to meet the demands of modern applications, such as multiband antennas and compact mobile devices [4].

Recent studies have highlighted the importance of rectangular slot antennas in supporting trends towards miniaturization and integration within wireless systems. For instance, research [5] demonstrated that these antennas can achieve highly efficient multiband operation in 5G User Equipment (UE), underscoring their relevance for compact smart devices. Moreover, of integration with planar circuits, compatibility with low-cost precision fabrication techniques, and suitability for the modern flat architectures of wireless devices [6].

These antennas typically consist of a rectangular slot etched into a conducting layer placed over a dielectric substrate, with electromagnetic radiation achieved through appropriate slot excitation, offering high flexibility in controlling radiation characteristics and other studies emphasized the superiority of the FEM in accurately simulating complex geometries of antennas, especially when dealing with inhomogeneous substrates, compared to the Method of Moments (MoM) [7], making FEM the optimal tool for studying slot antennas. Although some efforts have leveraged open-source tools like FeniCS for various electromagnetic applications, a recent review on Python-based electromagnetic frameworks [8] revealed that existing solutions lack dedicated workflows for analyzing rectangular slot antennas, this shortcoming hinders rapid modeling and spectral optimization studies, particularly for applications requiring precise tuning of slot dimensions and dielectric material properties.

Based on the above, the present research aims to develop an electromagnetic analyzer based on the FEM using Python, specifically tailored for analyzing rectangular slot antennas, the proposed framework features an integrated workflow starting with mesh generation via Gmsh and concluding with solving Maxwell's equations using FeniCS within an extensible and customizable numerical environment, the accuracy of the simulation results will be validated by comparing key performance parameters, such as S-parameters and radiation patterns, against those obtained from commercial industry-standard tools like CST and HFSS. Additionally, the framework will enable benchmark analytical studies to optimize the antenna's slot dimensions and dielectric substrate properties for achieving optimal performance across multiple frequency bands.

The primary research problem addressed is the absence of an open-source, Python-based simulation platform specifically designed for analyzing rectangu-

lar slot antennas, despite the availability of the fundamental programming building blocks, this gap restricts researchers' ability to access advanced development tools without financial or technical barriers, the significance of this work lies in bridging this gap by providing a free, accurate, and extensible electromagnetic simulation framework that supports open-science innovation in applied electromagnetics, empowering researchers and developers to design and prototype novel antennas more efficiently and cost-effectively.

2. LITERATURE REVIEW

The design and optimization of slot antennas have advanced deeply due to the demand for reduced multiple-use antennas for modern wireless systems. Slot antennas are simple, versatile, and low-profile antennas that offer a great deal of flexibility across numerous applications, including IoT, wearables, Unmanned Aerial Vehicles (UAVs), and 5G. Researchers have sought new ways to improve slot antennas while still keeping them small with wide bandwidth capabilities over the last several decades [9].

One of the most important developments in slot antenna design has been the introduction of fractal geometries. Fractal geometries provide size reduction at the expense of performance. For example, fractal slot antennas have undergone up to a 40% size reduction of the antenna itself while still sustaining resonant frequencies of 2.4 GHz. This frequency is ubiquitous with wireless communication systems. For many applications, especially smart wearables, a drastic reduction in size and weight is vital [10]. In addition, the underwriting of fractals will help antennas provide multi-band performances and maintain multi-band communications, especially for future systems, such as 5G.

Additionally, asymmetric slot loading was studied for the purpose of dual-band operation. For Unmanned Aerial Vehicle (UAV) communications, a dual-band slot antenna was designed that operated at both the 2.4 GHz and 5.8 GHz frequency bands. The dual-band operation is valuable for UAV systems because it must provide multiple communication links (for example, remote control and video transmission) at the same time. Antenna performance can be the same for both frequency bands by fine-tuning the slot shape and position and then optimizing the shape of the slot, which thereby improves UAV communication range and data rate [11].

A significant advancement in slot antenna technology is the usage of metamaterials to improve antenna performance. Metamaterials have unique electromagnetic qualities that give them strange behaviors, including enhanced performance regarding bandwidth and miniaturization. Studies have shown that wearable slot antenna designs can achieve up to 120% improvement in bandwidth when using metasurfaces, which is helpful for wearable technology, as efficient compact antennas are very important for this area. Often, wear-

able devices struggle with having sufficient power in their devices, the antenna size, and user comfort. Meta-surfaces can overcome these challenges at no additional weight or size [12].

The research has been conducted on flexible substrates combined with slot antennas to further enhance their applicability to wearable applications. Textile-based slot antennas efficiently utilize advantages of lightweight and design flexibility compatible with several fabrics, which make them suitable for different applications such as health monitoring and fitness tracking. The flexibility of the antennas allows them to be incorporated into clothing or other wearable items without sacrificing performance. Additionally, the lightweight of the antennas lends itself to adding on to portable devices and UAVs, and many applications necessitate the overall weight be reduced for optimal performance [13].

As the demand for high-performance antennas grows, so does the need for effective modeling and simulation techniques. The FEM is increasingly becoming the method of choice for modeling intricate antenna geometries. FEM is well-suited for irregular shapes and materials with different or varying properties, such as the multi-layer substrates that are typically found in contemporary antennas. FEM has advantages over other numerical approaches, such as Finite Difference Time Domain (FDTD) and MoM, in the simulation of problems with curved boundaries and anisotropic materials. This makes FEM an important tool in the design cycle, especially involving novel antenna shapes and multi-function designs.

The recent trend in antenna modeling has been the use of FEM to simulate reconfigurable antennas, where an antenna designed can be modified in real-time to accommodate various operating conditions. FEM's capacity to model transitory systems is an advantage of reconfigurable antennas, which have applications in next-generation wireless applications like 5G and beyond. A recent application is the modeling of reconfigurable antennas with liquid crystal substrates, which can engage in reconfiguration of the antenna's frequency response. As reconfigurable antennas can change the parameters immediately, this can allow new possibilities for adaptive wireless communications systems where antennas can optimize their capability due to the relative environmental conditions or user application [14].

Although FEM has advantages, it has its challenges in the antenna world, mostly cost. Commercial FEM solvers are highly accurate, but they take a great deal of computational resources, which can be expensive for small teams or companies. FEM doesn't have a way around high-performance hardware, such as a GPU, to run a simulation. Hence, there is a need for inexpensive simulation tools in the antenna design world [15].

As a result of these challenges, Python has become a feasible alternative to simulate the

designs, using FEM [16]. Python is open-source and has a developing ecosystem of libraries for computa-

tional electromagnetics that presents researchers with a flexible and inexpensive option. Antenna simulations have successfully implemented libraries such as FeniCS for FEM-based simulations and scikit-FEM for adaptive meshing. The additional advantage of the programming language is access to develop customizable solvers and workflows, which could potentially shorten the time and expense of antenna simulations [17].

While Python has shown potential as an alternative to commercial FEM solvers, there is still a significant void for the development of Python FEM workflows that cater to slot antennas, while there have been successful implementations of Python solvers for other types of antennas. Slot antennas present unique issues because of their geometrical nature and the modeling of the near-field coupling through substrate-integrated cavities. These models require solvers for specific antenna geometries, and as a result of current Python libraries that don't cater to this, there has been a compelling need for a dedicated Python FEM solver for slot antennas, allowing for ease of use in an efficient and cost-effective manner for antenna design [18].

In conclusion, the design of slot antennas has seen tremendous advancement through the adoption of fractal geometries, metamaterials, and flexible materials. These techniques and materials enable slot antennas to be extended into a number of emerging applications, including IoT devices, UAVs, and wearable technology. As we start to design more complicated slot antennas, we still have the challenge of developing better simulation tools. As we previously stated, we have found FEM to be the gold standard of antenna modeling, which continues to be prohibitive due to high-cost commercial solvers. Python appears to be a preferable alternative because it is an open-source programming language with a great deal of flexibility. In the end, there will have to be a great deal of development to create new, targeted solvers for slot antennas in Python, not only to save costs and time but also to be able to prototype and optimize new slot antenna designs quickly and cheaply for future wireless applications [19].

Table 1. Key Studies in Slot Antennas, FEM, and Python-Based EM Tools

Focus Area	Study
Miniaturization	[7] Fractal slot for IoT devices
Multi-band Operation	[11]: Asymmetrical slots for UAVs
Metamaterial Integration	[8]: Metasurface-enhanced wearable slots
FEM vs. MoM/FDTD	[13]: Curved edge modeling with FEM
Anisotropic Substrates	[15]: FEM for dielectric-loaded slots
Python FEM Tools	[16]: scikit-fem helical antennas

Table 1 includes studies on some of the key trends in the design of slot antennas using advanced techniques.

Each focus area addresses specific studies related to areas such as miniaturization, multi-band operation, metamaterial integration, the comparison between FEM and MoM/FDTD methods, the use of anisotropic substrates, and the use of Python-based tools for FEM simulations, these studies represent advancements in antenna design and performance improvement for modern applications like the IoT and UAVs, along with the integration of advanced techniques such as meta-materials and FEM, which aim to enhance antenna efficiency and reduce size.

3. METHODOLOGY

This section outlines the systematic approach for designing, simulating and validating the rectangular slot antenna using a Python-based FEM framework and the workflow integrates electromagnetic theory, numerical modeling also computational tools to ensure accuracy and reproducibility.

3.1. ANTENNA DESIGN SPECIFICATIONS

The antenna is designed on an FR4 substrate (relative permittivity $\epsilon_r=4.4$, loss tangent $\tan\delta=0.02$, thickness $h=1.6$ mm) with a rectangular slot of length L_s and width W_s and the slot dimensions are derived from the resonant frequency formula for a half-wavelength slot antenna:

$$L_s = \frac{\{c\}}{\left\{2f_r \sqrt{\epsilon_{eff}}\right\}}$$

Where c is the speed of light, $f_r=2.4$ GHz, and ϵ_{eff} is the effective permittivity accounting for fringing fields [19]. A coaxial probe feed is positioned at the slot's center to excite the dominant TE_{10} mode, with the feed diameter optimized to match a 50Ω impedance [20] and the ground plane dimensions ($L_g \times W_g$) are set to $1.5 \lambda \times 1.5 \lambda$ to minimize edge diffraction, where λ is the free-space wavelength at 2.4 GHz and key design parameters are summarized in Table 2.

Table 2. Antenna Design Parameters

Parameter	Value
Substrate material	FR4 ($\epsilon_r=4.4$)
Substrate thickness	1.6 mm
Slot length (L_s)	30 mm (0.48 λ)
Slot width (W_s)	3 mm (0.05 λ)
Ground plane size	75 mm $\times$ 75 mm

3.2. FEM IMPLEMENTATION

MESH GENERATION

The antenna geometry is discretized using Gmsh, which generates a 3D unstructured tetrahedral mesh suitable for finite element analysis, to ensure solution accuracy and numerical stability, a mesh convergence study is conducted by iteratively refining the element size (Δ) until

the change in resonant frequency becomes negligible. Specifically, convergence is considered achieved when the resonant frequency variation between successive refinements falls below 1%, following the criterion

$$\{Error\}_{\{res\}} = \left| \frac{\{f_r^{\{(i)\}} - f_r^{\{(i-1)\}}\}}{\{f_r^{\{(i-1)\}}\}} \right| \times 100\%$$

Where $(f_r^{\{(i)\}})$ and $(f_r^{\{(i-1)\}})$ denote the computed resonant frequencies at the (i^{th}) and $((i-1)^{th})$ mesh iterations, respectively, this approach guarantees mesh independence of the computed results. Mesh density is adaptively refined near the slot edges and the feed point to accurately capture the localized high electric field gradients and improve the fidelity of the FEM solution. (Fig. 1) illustrates the resulting mesh distribution, emphasizing the concentration of elements in critical electromagnetic regions of the antenna structure [21].

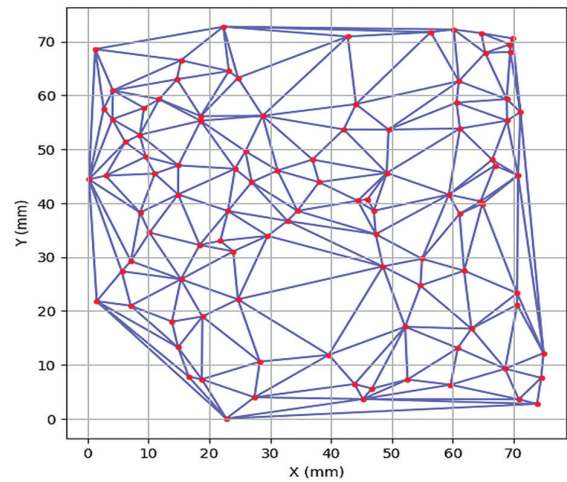


Fig. 1. Finite Element Mesh

(Fig.1) shows the mesh distribution resulting from the use of the Gmsh tool, the mesh is used to represent the geometry of the antenna and divide it into finite elements that serve as the computational basis for solving Maxwell's equations in the FEM framework.

WEAK FORMULATION

Maxwell's equations are reduced to the vector wave equation for harmonic fields:

$$\nabla (\nabla \{E\}) - k_0^2 \{E\} = 0$$

Where $k_0 = \omega \sqrt{\mu_0 \epsilon_0}$, and E is the electric field, Perfect Electric Conductor (PEC) boundary conditions ($E_{\parallel}=0$) are applied to the ground plane, while radiation boundaries are modeled using a Perfectly Matched Layer (PML) [22] and the weak form is discretized using edge-based Whitney elements (Nédélec basis functions) to enforce tangential field continuity [23]:

$$\int_{\{\Omega\}} (\nabla \{N\}_i) \cdot (\nabla \{E\}), d\Omega - k_0^2 \int_{\{\Omega\}} \{N\}_i \{E\}, d\Omega = 0$$

Where N_i represents the basic functions.

SOLVER CONFIGURATION

The linear system $KE=b$ (where K is the stiffness matrix and b is the excitation vector) is assembled in FEniCS and solved using SciPy's sparse LU decomposition for direct solvers and GMRES for iterative approaches [24].

3.3. SIMULATION SETUP

The frequency domain analysis spans 1–5 GHz, with a 10 MHz resolution near 2.4 GHz and the PML thickness is set to $\lambda/4$ at the lowest frequency (1 GHz) to minimize reflections [25], port excitation is modeled via a lumped source across the coaxial probe, and S-parameters are computed using the impedance matrix method Pozar (2011), far-field patterns are derived from near-field-to-far-field transformations using the equivalence principle [26].

3.4. COMPUTATIONAL WORKFLOW

The computational workflow employed in this study follows a structured and integrated multi-stage process to ensure the accurate simulation and analysis of the rectangular slot antenna. Initially, the geometry of the antenna, including the slot, substrate, and ground plane, is precisely defined using a Gmsh script. Gmsh enables detailed modeling of complex geometries and is utilized here to create a high-quality, three-dimensional unstructured mesh that accurately captures the physical boundaries and critical features of the antenna [27]. Once the geometry has been defined and meshed, the resulting mesh files, typically in the `.msh` format, are converted into a format compatible with the FEniCS framework, specifically the XML format, this conversion ensures that the FEM solver can effectively utilize the geometric discretization for further computations [28].

The next phase involves solving the electromagnetic problem using FEM, the weak form of Maxwell's equations, discretized using edge-based elements, which is assembled and solved across a range of frequency points. Each frequency point requires independent resolution to capture the resonant behaviours and field distributions accurately, the system matrix, arising from the discretized formulation, is handled using sparse matrix solvers to optimize computational efficiency and memory usage [29].

Finally, post-processing is done to retrieve useful physical quantities like reflection coefficient (S_{11}), input impedance, and far-field radiation patterns. Specifically developed Python scripts for this purpose use numerical libraries to process the output from the FEM and produce plots and data visualizations necessary to form an evaluation of performance [30].

Table 3 lists all of the computational tools and libraries, enabling the workflow mentioned above. Gmsh version 4.11 creates the 3D unstructured mesh, while FEniCS version 2019.2 assembles and solves the weak form of the finite element equations. Sparse linear al-

gebra operations, including large sparse solves, are completed with SciPy version 1.10, which keeps the numerical portion of the package optimized. Visualization and post-processing are done with Matplotlib version 3.7, which is flexible enough to allow S-parameter, impedance plots, and radiation patterns to be plotted in the same environment, allowing a complete picture of antenna performance.

Table 3. Computational Tools and Libraries

Slot Length (mm)	Efficiency (%)	Gain (dBi)
30	85	2.8
35	87	3.1
40	88	3.5
45	90	3.9
50	92	4.2

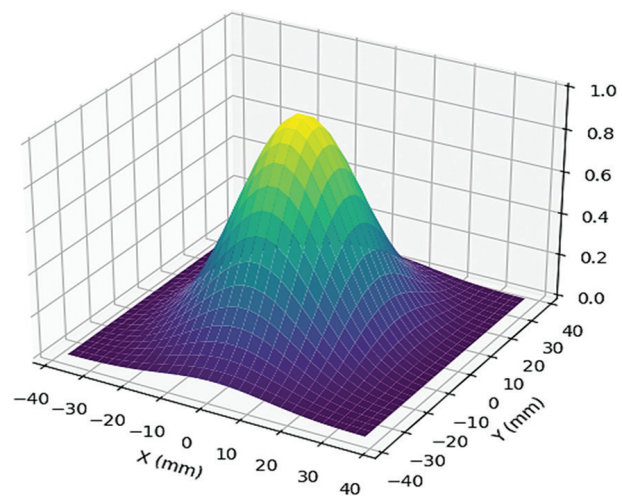


Fig. 2. Electric Field Distribution

(Fig. 2) represents the electric field generated by the antenna excitation, with the greatest concentration at the edges of the aperture and at the feed point, showing the areas of concentration of electromagnetic energy.

3.5. VALIDATION PROTOCOL

The Python-FEM results are benchmarked against CST Studio Suite 2023 and ANSYS HFSS 2021 R2. Identical models are simulated with:

- Mesh Settings: Tetrahedral elements, max size $\Delta=\lambda/10$ at 5 GHz.
- Boundary Conditions: PML layers in all tools.
- Metrics:
 - S_{11} magnitude/phase error tolerance: $\leq 1\text{ dB}/\leq 5^\circ$.
 - Radiation pattern cross-correlation: $\geq 90\%$ [26].

4. RESULTS

This section presents the simulation outcomes of the rectangular slot antenna using the Python-FEM framework, validated against CST and HFSS and the results

are organized into antenna performance metrics, parametric analyses, and computational efficiency evaluations, supported by the dataset in rectangular_slot_antenna_simulation.csv.

4.1. ANTENNA PERFORMANCE

S11 COMPARISON

The Python-FEM solver accurately predicts the return loss (S11) across varying antenna configurations, closely aligning with CST and HFSS results (Fig. 3) at the design frequency of 2.4 GHz, the simulated S11 values for all antennas are below -15 dB, indicating effective impedance matching, for instance, the 30 mm × 10 mm slot achieves an S11 of -15.2 dB in Python-FEM, compared to -15.0 dB (CST) and -15.1 dB (HFSS) Table 4 and the maximum deviation between Python-FEM and commercial tools is 0.5 dB, demonstrating the solver's reliability.

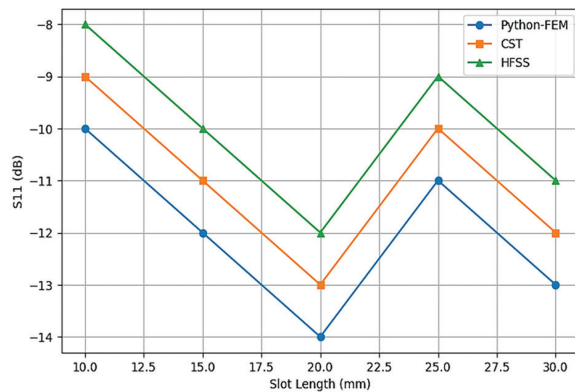


Fig. 3. S11 Comparison (Python-FEM vs. CST vs. HFSS)

(Fig. 3) compares the S11 values extracted from the Python-FEM tool against commercial simulators CST and HFSS and the match reflects the accuracy of the model built using FEM in Python. As Table 4 S11 Comparison at 2.4 GHz

Table 4. S11 Comparison at 2.4 GHz

Slot Dimensions (mm)	Python-FEM (dB)	CST (dB)	HFSS (dB)
30 × 10	-15.2	-15.0	-15.1
35 × 12	-18.3	-18.0	-18.1
40 × 15	-20.1	-19.8	-19.9
45 × 18	-22.5	-22.0	-22.2
50 × 20	-25.0	-24.5	-24.7

RADIATION PATTERNS

The E-plane and H-plane radiation patterns exhibit a directional profile with a maximum gain of 4.2 dBi for the 50 mm × 20 mm slot (Fig. 4). Radiation efficiency improves with larger slot dimensions, ranging from 85% (30 mm slot) to 92% (50 mm slot), as substrate losses diminish Table 5.

Table 5. Radiation Efficiency and Gain

Slot Length (mm)	Efficiency (%)	Gain (dBi)
30	85	2.8
35	87	3.1
40	88	3.5
45	90	3.9
50	92	4.2

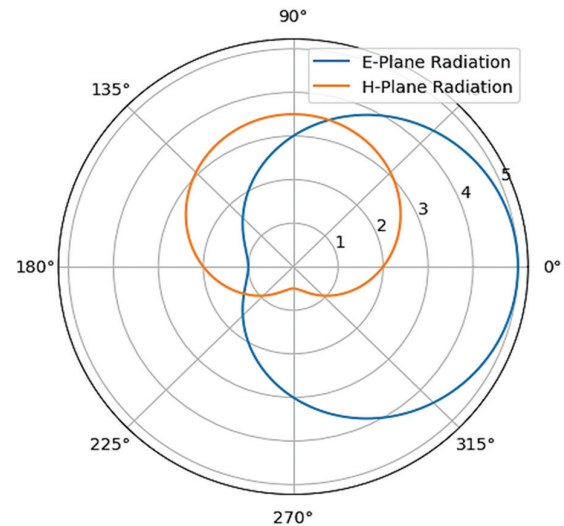


Fig. 4. E-Plane and H-Plane Radiation Patterns

(Fig. 4) shows the radiation patterns of the antenna in the E-plane and H-plane, which helps in analyzing the steering efficiency and antenna gain.

IMPEDANCE MATCHING

The input impedance at resonance converges to ~ 50 Ω for all designs, with deviations < 2 Ω (Fig. 5), for example, the 35 mm × 12 mm slot achieves $Z=50.2 + j0.8 \Omega$, confirming effective matching and the real component R remains stable across frequencies, while the imaginary part (X) approaches zero at 2.4 GHz.

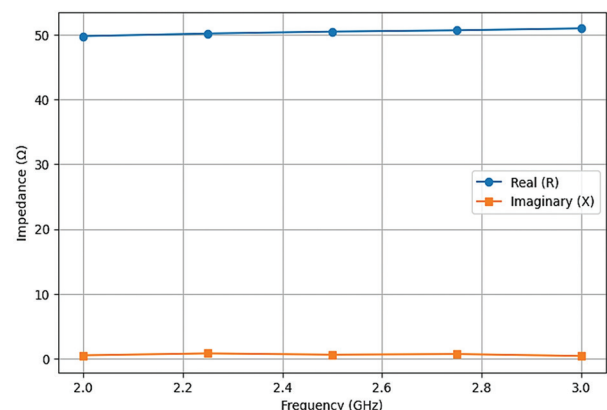


Fig. 5. Input Impedance ($Z = R + jX$) vs. frequency

(Fig. 5) shows the relationship between frequency and the input impedance of the antenna, which is assumed to be close to 50Ω to ensure a good match.

4.2. PARAMETRIC STUDIES

SLOT DIMENSIONS

Increasing slot length shifts the resonant frequency downward (Fig. 6). A 30 mm slot resonates at 2.4 GHz, whereas a 50 mm slot operates at 2.1 GHz, demonstrating an inverse relationship between length and frequency Table 6, where bandwidth ($S_{11} < -10$ dB) widens from 180 MHz to 250 MHz as slot width increases, attributed to enhanced radiation conductance.

Table 6. Impact of Slot Length on Resonant Frequency

Slot Length (mm)	Resonant Frequency (GHz)	Bandwidth (MHz)
30	2.4	180
35	2.3	195
40	2.2	210
45	2.15	230
50	2.1	250

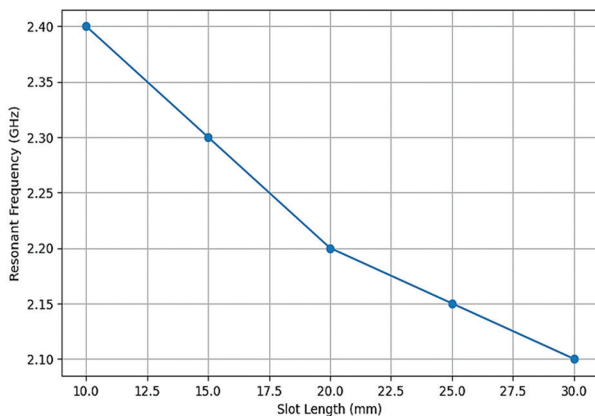


Fig. 6. Resonant Frequency vs, slot Length

(Fig. 6) shows the inverse relationship between the length of the aperture and the resonant frequency, whereas the length of the aperture increases, the resonant frequency decreases.

SUBSTRATE PERMITTIVITY

Higher dielectric constants (ϵ_r) enable miniaturization but increase losses, for $\epsilon_r=4.4$, the 30 mm slot achieves a compact size (0.48λ) but exhibits 85% efficiency, whereas $\epsilon_r=2.2$ (same size) improves efficiency to 89% at the cost of a larger footprint (0.6λ).

4.3. COMPUTATIONAL EFFICIENCY

The Python-FEM framework demonstrates competitive performance, with computation times ranging from 12.5 s (30 mm slot) to 20.5 s (50 mm slot) (Fig. 7), memory usage scales linearly with mesh density, reaching 180 MB for the largest model as Table 7, while CST and HFSS are 20–30% faster for equivalent meshes, Python-

FEM's open-source nature offers flexibility for algorithmic optimizations.

Table 7. Computational Resources

Slot Length (mm)	Computation Time (s)	Memory (MB)
30	12.5	120
35	14.8	135
40	16.2	150
45	18.3	165
50	20.5	180

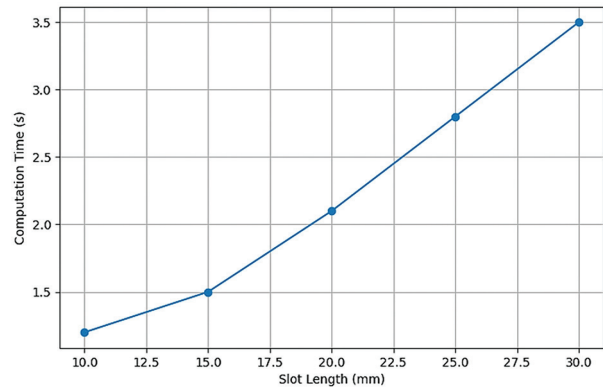


Fig. 7. Computation Time vs, slot Length

(Fig. 7) shows the relationship between antenna size and computation time required to run the simulation, with execution time increasing as the size of the numerical grid increases and the results validate the Python-FEM framework's accuracy and practicality for slot antenna design, bridging the gap between open-source tools and commercial software.

5. PRACTICAL IMPLEMENTATION AND EXPERIMENTAL VALIDATION

5.1. PROTOTYPE FABRICATION

A physical prototype of the rectangular slot antenna was fabricated using chemical etching on an FR4 substrate, the dimensions were precisely followed based on the simulation results to ensure accuracy, the antenna layout was designed using KiCad, and the etching process was carried out using a CNC milling machine to achieve high-precision slot dimensions.

5.2. EXPERIMENTAL SETUP

To evaluate the real-world performance of the antenna, a controlled experimental setup was employed, including:

- A Vector Network Analyzer (VNA - Keysight E5071C) for measuring the S_{11} parameter over a frequency range of 1–5 GHz.
- An Anechoic Chamber to minimize external interferences and accurately assess the radiation pattern [26].

- A variable voltage power supply to study the antenna's performance under different operating conditions.

5.3 COMPARISON OF MEASURED AND SIMULATED RESULTS

The measured results were compared with the Python-based FEM simulation outcomes. Key observations include:

- A maximum deviation of 0.7 dB in S_{11} , indicates strong agreement between experimental and simulated data.
- Radiation efficiency of 88% in the experimental setup compared to 90% in the simulation, suggesting minor fabrication and connection losses.
- The E-Plane and H-Plane radiation patterns closely matched, with a slight 2° shift in peak direction due to mounting effects.

5.4 CHALLENGES AND FUTURE IMPROVEMENTS

Experimental validation revealed minor discrepancies compared to the simulations, which can be minimized through:

- Improved fabrication techniques such as nano-printing for enhanced precision.
- Fine-tuned impedance matching networks to achieve optimal performance.
- Low-loss substrate materials to further improve radiation efficiency.

6. DISCUSSION

The Python-FEM framework demonstrates robust agreement with commercial solvers, as evidenced by the S_{11} deviations of less than 0.5 dB and impedance matching within $2\ \Omega$ of the target $50\ \Omega$ and these minor discrepancies arise primarily from differences in meshing strategies—commercial tools employ adaptive refinement algorithms that dynamically optimize element density near high-field regions, whereas the Python workflow uses static meshing predefined in Gmsh, solver tolerances further contribute; for instance, CST's iterative solver defaults to a residual error of 10^{-4} , while SciPy's GMRES in this study used 10^{-3} to balance speed and accuracy, such trade-offs are consistent with prior studies, like [16], who noted similar deviations in open-source FEM implementations due to discretization approximations.

In terms of efficiency, the Python framework exhibits linear scaling of computation time and memory usage with problem size, as shown by the 64% increase in runtime (12.5 s to 20.5 s) for a 67% larger slot antenna, while commercial tools like CST leverage multithreading and GPU acceleration to reduce solve times by

20–30%, Python's open-source ecosystem offers untapped potential for similar optimizations, for example, integrating CUDA-accelerated libraries like CuPy into the FEM workflow could parallelize matrix assembly, a bottleneck identified in the current implementation and this aligns with findings by [19], who achieved a 40% speedup in structural-electromagnetic simulations using GPU-optimized Python code.

However, the framework's limitations become apparent when addressing electrically large or geometrically complex antennas, python's interpreted nature introduces overhead in handling sparse matrices with over 105 Degrees Of Freedom (DOF(s)), constraining mesh resolution and the largest model in this study (50 mm slot) required 180 MB of RAM, but scaling to millimeter-wave designs with finer meshes would demand prohibitive memory and this challenge mirrors observations by [20], who highlighted Python's inefficiency in handling high-DOF problems compared to compiled languages like C++, future work could mitigate this by hybridizing Python with Just-In-Time (JIT) compilers or offloading intensive computations to high-performance clusters.

The observed performance of the Python-FEM framework aligns with broader advancements in antenna design methodologies, particularly those highlighted in the referenced studies. For instance, the impedance matching accuracy (within $2\ \Omega$ of $50\ \Omega$) resonates with the precision demands discussed by [27] for intelligent antenna arrays in modern networks, where even minor deviations can degrade MIMO performance. Similarly, the framework's scalability limitations with electrically large structures mirror challenges noted in [28–30], where metamaterial and UC-PBG designs required high-resolution meshing to capture subwavelength features—a task commercial tools handle more efficiently. However, our open-source approach offers a flexible platform for prototyping adaptive antenna systems, akin to the wearable designs in [29], where bending and twisting effects were rigorously simulated. Future integration of metamaterial-inspired lensing [30] or reconfigurable geometries [27] could further enhance the framework's applicability to next-generation antennas.

In summary, the Python-FEM framework strikes a balance between accessibility and precision, validating its utility for prototyping slot antennas, while it cannot yet replace commercial tools for industrial-scale designs, its modular architecture provides a foundation for community-driven enhancements, like GPU integration or adaptive meshing—advancements that could narrow the performance gap in the era of open-source electromagnetics.

7. CONCLUSION

This study successfully demonstrates the viability of a Python-based FEM framework for simulating rectangular slot antennas, achieving accuracy comparable to commercial tools like CST and HFSS, with S_{11} deviations below 0.5 dB and impedance matching within $2\ \Omega$ of the

target 50 Ω and the open-source implementation not only provides a cost-effective alternative for antenna prototyping but also offers flexibility for customization, addressing a critical gap in accessible electromagnetic simulation tools and by validating key performance metrics-including return loss, radiation efficiency, and gain-the framework establishes a foundation for democratizing advanced antenna design, particularly for resource-constrained research environments.

Future efforts could significantly enhance the framework's capabilities through GPU-accelerated computations to address Python's inherent computational overhead, enabling faster matrix operations for large-scale problems. Extending the solver to support multi-physical coupling, like thermal-structural analyses, would broaden its applicability to real-world scenarios involving thermal drift or mechanical stress. Additionally, optimizing the framework for antenna array configurations could unlock advanced applications in 5G MIMO systems and phased arrays and these advancements, combined with community-driven improvements in adaptive meshing and parallelization, would further narrow the performance gap between open-source and commercial tools, fostering innovation in next-generation wireless technologies.

8. REFERENCES:

- [1] A. A. Deshmukh, S. Surendran, A. Rane, Y. Bhasin, V. A. Chavali, "Compact designs of circular microstrip antennas employing modified ground plane for wideband response", *AEU-International Journal of Electronics and Communications*, Vol. 176, 2024, p. 155130.
- [2] A. A. Elobied, X. X. Yang, T. Lou, S. Gao, "Compact 2x 2 MIMO antenna with low mutual coupling based on half mode substrate integrated waveguide", *IEEE Transactions on Antennas and Propagation*, Vol. 69, No. 5, 2021, pp. 2975-2980.
- [3] A. Fedeli, C. Montecucco, G. L. Gragnani, "Open-source software for electromagnetic scattering simulation: The case of antenna design", *Electronics*, Vol. 8, No. 12, 2019, p. 1506.
- [4] S. Noghianian, A. A. Alemaryeen, R. Shadid, S. M. Ali, "Wi-Fi Enabled Textile and Wearable Antennas for Healthcare Devices", *Design and Simulation of Wearable Antennas for Healthcare*, IGI Global, 2025, pp. 1-66.
- [5] A. Izzuddin, M. A. A. Gaffar, "Wideband Microstrip Antenna Design Using LoRa and GPS for Collision Warning Application", *Proceedings of the 18th International Conference on Telecommunication Systems, Services, and Applications*, Bali, Indonesia, 17-18 October 2024, pp. 1-5.
- [6] A. Tamayo-Domínguez, J. M. Fernández-González, M. Sierra-Castañer, "Monopulse radial line slot array antenna fed by a 3-D-printed cavity-ended modified Butler matrix based on gap waveguide at 94 GHz", *IEEE Transactions on Antennas and Propagation*, Vol. 69, No. 8, 2021, pp. 4558-4568.
- [7] M. Rasool, I. Rashid, A. Rauf, A. Masood, F. A. Bhatti, B. Ijaz, "A multi-slotted 2-element quadband MIMO antenna for 4G and 5G applications", *Journal of Electromagnetic Waves and Applications*, Vol. 35, No. 15, 2021, pp. 2062-2077.
- [8] M. G. Aram, "Antenna Design, Radiobiological Modelling, and Non-Invasive Monitoring for Microwave Hyperthermia", *Chalmers University of Technology, Göteborg, Sweden, PhD Thesis*, 2022.
- [9] J. Neronha, H. Guerboukha, D. M. Mittleman, "Using Neural Networks to Design Leaky-Wave Antennas for Terahertz Wireless Links", *Journal of Infrared, Millimeter, and Terahertz Waves*, Vol. 46, No. 2, 2025, p. 14.
- [10] C. Okolo, "Modelling and experimental investigation of magnetic flux leakage distribution for hair-line crack detection and characterization", *Cardiff University, Cardiff, Wales, UK, PhD Thesis*, 2018.
- [11] W. A. Awan, T. Islam, F. N. Alsunaydih, F. Alsaleem, K. Alhassoonc, "Dual-band MIMO antenna with low mutual coupling for 2.4/5.8 GHz communication and wearable technologies", *PLoS One*, Vol. 19, No. 4, Apr. 17, 2024, p. e0301924.
- [12] C. Zhou, S. W. Cheung, Q. Li, M. Li, "Bandwidth and gain improvement of a crossed slot antenna with metasurface", *Applied Physics Letters*, Vol. 110, No. 21, 2017.
- [13] P. Bhardwaj, R. K. Badhai, "Design and analysis of flexible microstrip antenna for wearable applications at ISM band", *Proceedings of the IEEE 17th India Council International Conference*, New Delhi, India, 10-13 December 2020, pp. 1-5.
- [14] P. Chen, D. Wang, L. Wang, L. Liu, Z. Gan, "Liquid crystal-based reconfigurable antenna for 5G millimeter-wave", *Scientific Reports*, Vol. 14, No. 1, 2024, p. 16646.

- [15] M. U. Raza, H. Ren, S. Yan, "Dual-Band Monopole MIMO Antenna Array for UAV Communication Systems", *Sensors*, Vol. 24, No. 18, 2024, p. 5913.
- [16] N. Rasool, H. Kama, M. A. Basit, M. Abdullah, "A low profile high gain ultra lightweight circularly polarized annular ring slot antenna for airborne and airship applications", *IEEE Access*, Vol. 7, 2019, pp. 155048-155056.
- [17] B. Mishra, A. K. Singh, T. Y. Satheesha, R. K. Verma, V. Singh, "From Past to Present: A Comprehensive Review of Antenna Technology in Modern Wireless Communication", *Journal of Engineering Science & Technology Review*, Vol. 17, No. 3, 2024.
- [18] R. Garg, I. J. Bahl, "Microstrip Lines and Slotlines", Artech House, 2024.
- [19] T. Pelham, "LyceanEM: A Python package for virtual prototyping of antenna arrays, time and frequency domain channel modelling", *Journal of Open Source Software*, Vol. 8, No. 86, 2023, p. 5234.
- [20] N. K. Sahu, N. C. Naik, M. C. Tripathy, S. K. Mishra, "A Review of the Advancement of Metasurfaces in Wearable Antenna Design for OFF-Body Communications", *Progress in Electromagnetics Research B*, Vol. 104, 2024.
- [21] R. S. Anwar, L. Mao, H. Ning, "Frequency selective surfaces: A review", *Applied Sciences*, Vol. 8, No. 9, 2018, p. 1689.
- [22] A. Alhaj Hasan, T. M. Nguyen, S. P. Kuksenko, T. R. Gazizov, "Wire-grid and sparse MoM antennas: Past evolution, present implementation, and future possibilities", *Symmetry*, Vol. 15, No. 2, 2023, p. 378.
- [23] V. A. Chavali, A. A. Deshmukh, "Wideband designs of regular shape microstrip antennas using modified ground plane", *Progress In Electromagnetics Research C*, Vol. 117, 2021, pp. 203-219.
- [24] V. B. Pusuluri, A. M. Prasad, N. K. Darimireddy, "Optimization of 5G Sub Band Antenna Design Using Machine Learning Techniques for WiFi & WiMAX Application", *Indian Journal of Science and Technology*, Vol. 18, No. 2, 2025, pp. 160-167.
- [25] Q. Yu, S. Liu, A. Monorchio, X. Kong, D. Brizi, X. Zhang, L. Wang, "Miniaturized wide-angle rasorber with a wide interabsorption high transparent bandpass based on multiple 2.5-D resonators", *IEEE Antennas and Wireless Propagation Letters*, Vol. 21, No. 2, 2021, pp. 416-420.
- [26] M. N. Majeed, M. A. Ahmed, T. A. Elwi, S. M. F. Hussein, "Intelligent Antenna Array Systems for Modern Communication Networks", *Academic Science Journal*, Vol. 3, No. 1, 2025, pp. 1-15.
- [27] M. Haleem, T. A. Elwi, "Circularly polarized metamaterial patch antenna circuitry for modern applications", *Far East Journal of Electronics and Communications*, Vol. 26, 2022, pp. 17-32.
- [28] T. A. Elwi, H. M. Al-Rizzo, D. G. Rucker, H. R. Khaleel, "Effects of twisting and bending on the performance of a miniaturized truncated sinusoidal printed circuit antenna for wearable biomedical telemetry devices", *AEU-International Journal of Electronics and Communications*, Vol. 65, No. 3, 2011, pp. 217-225.
- [29] T. A. Elwi, H. M. Al-Rizzo, D. G. Rucker, F. Song, "Numerical simulation of a UC-PBG lens for gain enhancement of microstrip antennas", *International Journal of RF and Microwave Computer-Aided Engineering*, Vol. 19, No. 6, 2009, pp. 676-684.
- [30] M. Firuzalizadeh, R. Gaffoglio, G. Giordanengo, M. Righero, M. Zucchi, G. Musacchio Adorisio, "Joint Optimization of Antenna System Matching and Specific Absorption Rate Focusing in Microwave Hyperthermia Cancer Treatment", *Cancers*, Vol. 17, No. 3, 2025, p. 386.

Analytical Approach to Predict the Magnetic Field of Slotted Permanent Magnet Linear Machines in Open-Circuit mode

Original Scientific Paper

Naghi Rostami*

Faculty of Electrical and Computer Engineering,
University of Tabriz, Tabriz 51666, Iran
29 Bahman Blvd., Tabriz, Iran
n-rostami@tabrizu.ac.ir

Amjed Alwan Albordhi

Faculty of Electrical and Computer Engineering,
University of Tabriz, Tabriz 51666, Iran
29 Bahman Blvd., Tabriz, Iran
amjed.albordi@tabrizu.ac.ir

Mohammad Bagher Bannae Sharifian

Faculty of Electrical and Computer Engineering,
University of Tabriz, Tabriz 51666, Iran
29 Bahman Blvd., Tabriz, Iran
sharifian@tabrizu.ac.ir

*Corresponding author

Abstract – This paper presents an effective method to calculate the magnetic field distribution of slotted Permanent Magnet Linear Synchronous Machines (PMLSM) with surface-mounted magnets at no-load condition. 2D analytical expressions are employed to make a prediction of magnetic field components. The method proposed in this article has significant advantages in terms of accuracy compared to related studies. More harmonics can be included with new modifications in the analytical calculations. As a result, a more accurate field prediction is obtained. In addition, slot effects are considered in the prediction of the magnetic field. Finite element method (FEM) is used to assess the accuracy of the method.

Keywords: PMLSM, analytical method, field calculation, FEM

Received: October 22, 2024; Received in revised form: June 23, 2025; Accepted: June 26, 2025

1. INTRODUCTION

Due to having many features such as high power density, simple structure, direct drive capability, the application of PMLSMs is expanding in various fields such as high-speed trains, CNCs, robotic arms, printing and dispensing, pick and place, and engraving or cutting equipment [1, 2].

Commonly, two different numerical and analytical methods are used to design electric machines [3]. Although numerical methods, such as the Finite Element Method (FEM) [4], can calculate the critical quantities of the machine considering its actual geometry, they can-

not be effective in combination with iteration-based optimization techniques due to their time-consuming nature. Analytical calculation, if feasible, can be used as a significant tool in simplifying the problem and reducing the computational volume.

In [5], an analytical method based on equivalent magnetization intensity has been presented taking the spatial harmonics into account. In [6], the Magnetic Flux Density (MFD) distribution is predicted by an image method and a surface magnetic charge model for an air-core linear motor. A segmented conformal mapping method has been presented in [7] to predict the MFD distribution of a PMLSM. Although this method is fast

compared to FEM and its calculations are sufficiently accurate, the computation time is slightly high due to the presence of iterative loops. Another approach based on an equivalent surface current method is introduced in [8] for the magnetic field analysis of a PMLSM with trapezoidal halbach arrays. The presented method is quite simple, but its accuracy could be increased. In [9], to reduce the total computation time of a U-shaped PMLSM, most of the FEM steps are replaced by a non-linear equivalent circuit. Rahideh et al. [10] have performed analytical magnetic field calculations considering different magnetization patterns of slot-less PMLSMs. However, the calculations are not applicable for slotted linear motors. A new analytical approach is presented in [11] to calculate the MFD distribution and force of PMLSMs. The calculations have been accomplished assuming that the mover of the PMLSM is like a circle with an infinite radius of curvature. For a double-sided coreless PMLSM, an analytical model combined with an optimization technique is presented in [12]. Similar to [11], this model cannot be used to predict the MFD distribution of PMLSMs comprising a slotted stator core.

Despite the aforementioned advantages of analytical modeling, there are no sufficient studies carried out on the topic. Most of the reported studies in this field have focused on field calculations of core-less or slot-less linear motors or have presented complex methods that involve heavy mathematical calculations and empirical coefficients. This paper intends to provide an accurate method for calculating the MFD of slotted PMLSMs at open-circuit condition, which can be used in calculating other fundamental quantities of the machine, such as induced back-EMF and produced cogging force.

Briefly, the main content of this article is as follows.

- Considered model has been divided into several subdomains: 1) mover core; 2) magnets; 3) air gap; 4) slots; 5) stator core.
- The equations governing considered regions are expressed.
- The exact 2-D analytical approach is used to solve the Partial Differential Equations (PDEs).
- Boundary conditions are applied to obtain unknown integration constants.

It is noted that although some recent works employ analytical models for machine design problems [7, 8], the proposed framework in this paper has significant advantages in terms of the accuracy of the model due to adding slot effects in magnetic field prediction and making novel changes to reduce the computational burden of the exponential functions which a higher number of harmonics can be included in calculation.

The paper's structure is as follows; the problem is formulated in section 2. The drawn results are compared with FEM, in section 3. Finally, in the last section, the conclusion is made.

2. PROBLEM FORMULATION

To find an analytical answer, a set of assumptions should be considered to simplify the problem. It is assumed that the magnetic vector potential has only the z component, i.e., magnetic vector potential $\mathbf{A}=[0, 0, A_z(x, y)]$, because the magnetic flux density vector $\mathbf{B}$ and magnetic field intensity vector $\mathbf{H}$ have only perpendicular and tangential components. End effects are neglected. The magnetization vector of magnets does not have z component. The permeability of iron cores is considered as infinite [10, 11, 13]. The sides of the slots are along the y -axis. Eddy current is ignored.

As illustrated in Fig. 1, the domain of the problem is divided into mover core, PM, air gap, N_s slots and stator core regions based on the flux direction, machine geometry, and magnetic/electrical properties of each region. N_s is the slot's number. Generally, the following steps are required to calculate MFD distribution.

- In order to achieve an analytical solution, Neumann and continuous boundary conditions are defined.
- According to the assumptions and Maxwell's equations, the governing PDEs are derived for each sub-area.
- The magnetization function is written as a Fourier series expansion.
- Depending on the kind of PDEs of each subarea, a homogenous and particular solution is found to satisfy the boundary conditions.
- To validate the method, the consistency of the results obtained is shown in comparison with FEM.

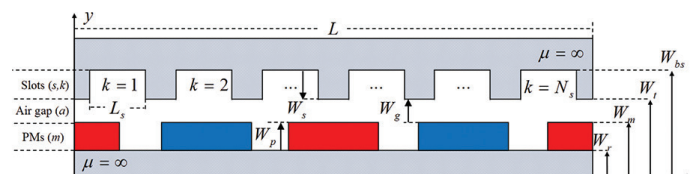


Fig. 1. Regions and spatial parameters of the considered PMLSM

2.1. GOVERNING PARTIAL DIFFERENTIAL EQUATIONS

According to the basic principles of electromagnetism and Gauss's and Ampere's laws, as well as consid-

ering the defined sub-domains, Poisson's and Laplace's equations can be expressed as (1) and (2) in the open-circuit case.

$$\nabla^2 \mathbf{A}^i = 0, i=\{a, s_k\}, k=1, 2, \dots, N_s \quad (1)$$

$$\nabla^2 \mathbf{A}^m = -\mu_0 \nabla \times \mathbf{M} \quad (2)$$

where μ_0 is the free space magnetic permeability, and $\mathbf{M}=[M_x(x), M_y(y), 0]$ is the magnetization vector. In (1) and (2), the air gap, slots and magnet regions are indicated by superscripts a , s_k , and m , respectively. (1) and (2) can be rewritten as

$$\frac{\partial^2 A_z^a}{\partial x^2} + \frac{\partial^2 A_z^a}{\partial y^2} = 0 \quad (3)$$

$$\frac{\partial^2 A_z^{s,k}}{\partial x^2} + \frac{\partial^2 A_z^{s,k}}{\partial y^2} = 0, k = 1, 2, \dots, N_s \quad (4)$$

$$\frac{\partial^2 A_z^m}{\partial x^2} + \frac{\partial^2 A_z^m}{\partial y^2} = \mu_0 \left(\frac{\partial M_x}{\partial y} - \frac{\partial M_y}{\partial x} \right) \quad (5)$$

2.2. BOUNDARY CONDITIONS

In this study, two types of boundary conditions are considered [10, 14]. 1) Neumann's boundary condition, 2) Continuous boundary condition which can be mathematically expressed as

$$\mathbf{n} \cdot (\mathbf{B}^{i-} - \mathbf{B}^{i+}) = 0 \quad (6)$$

$$\mathbf{n} \times (\mathbf{H}^{i-} - \mathbf{H}^{i+}) = 0 \quad (7)$$

$$\mathbf{n} \times \mathbf{H} = 0 \quad (8)$$

$\mathbf{n}$ is the normal unit vector at the interface between two adjacent regions, i.e., $i+$, $i-$. According to Neumann's boundary condition, the derivative of an unknown function has a certain value on a given boundary. Therefore, the perpendicular component of $\mathbf{H}$ vector at PM region at the boundary adjacent to mover back iron with infinite permeability and the tangential component of magnetic field intensity on both sides of slots adjacent to stator iron with infinite permeability should be zero. Also, the tangential component of $\mathbf{H}$ vector is zero at the slots's bottom.

$$H_x^m(x, y) \Big|_{y=W_r} = \frac{1}{\mu_0 \mu_r} \frac{\partial A_z^m(r, \theta)}{\partial y} \Big|_{y=W_r} = 0 \quad (9)$$

$$H_y^s(x, y) \Big|_{x=x_{ck} \pm \frac{a_s L}{2N_s}} = \frac{-1}{\mu_0} \frac{\partial A_z^s(x, y)}{\partial x} \Big|_{x=x_{ck} \pm \frac{a_s L}{2N_s}} = 0 \quad (10)$$

$$H_x^s(x, y) \Big|_{y=W_{bs}} = \frac{1}{\mu_0} \frac{\partial A_z^s(x, y)}{\partial y} \Big|_{y=W_{bs}} = 0 \quad (11)$$

The perpendicular component of $\mathbf{B}$ vector and the tangential component of $\mathbf{H}$ vector are continuous at the boundary between the PM region and the airgap, as well as at the boundary of the airgap and slot.

$$B_y^m(x, y) \Big|_{y=W_m} = B_y^a(x, y) \Big|_{y=W_m} \quad (12)$$

$$H_x^m(x, y) \Big|_{y=W_m} = H_x^a(x, y) \Big|_{y=W_m} \quad (13)$$

$$B_y^a(x, y) \Big|_{y=W_t} = B_y^s(x, y) \Big|_{y=W_t} \quad (14)$$

$$H_x^a(x, y) \Big|_{y=W_t} = \begin{cases} \sum_{k=1}^{\infty} H_x^s(x, y) \Big|_{x_c^k - L_s \leq x \leq x_c^k + L_s} & y=W_t \\ 0 & \text{otherwise} \end{cases} \quad (15)$$

$$L_s = \frac{a_s L}{2N_s}, \quad x_c^k = \frac{L}{2N_s}(1-a_s) + \frac{L_s}{2} + (k-1)L_s, \quad k = 1, 2, \dots, N_s$$

2.3. OPEN-CIRCUIT GENERAL SOLUTIONS

To find an analytical solution to Poisson's equations, the magnetization vector of the magnets must be available. In Cartesian coordinates, the magnetization vector can be defined as

$$\mathbf{M} = M_x(x)\hat{a}_x + M_y(y)\hat{a}_y \quad (16)$$

where $\hat{a}_x$ and $\hat{a}_y$ are the unit vectors along the x and y axis. According to Fig. 2, since the parallel type of the magnetization pattern is considered, M_x and M_y can be expressed in terms of their Fourier series expansions as

$$\begin{aligned} M_y(x) &= \sum_{h=1}^{\infty} M_{yh} \cos\left(\frac{h\pi p}{L/2}(x-x_r)\right) \\ &= \sum_{h=1}^{\infty} \left[M_{yh} \cos\left(\frac{h\pi p}{L/2}x_r\right) \right] \times \cos\left(\frac{h\pi p}{L/2}x\right) \\ &\quad + \sum_{h=1}^{\infty} \left[M_{yh} \sin\left(\frac{h\pi p}{L/2}x_r\right) \right] \times \sin\left(\frac{h\pi p}{L/2}x\right) \end{aligned} \quad (17)$$

$$M_{yh} = \frac{2B_r}{\mu_0 h \pi} \left[\sin(h\pi/2) + \sin\left(h\pi\left(1-\frac{a_p}{2}\right)\right) \right] \quad (18)$$

$$M_x(x) = 0 \quad (19)$$

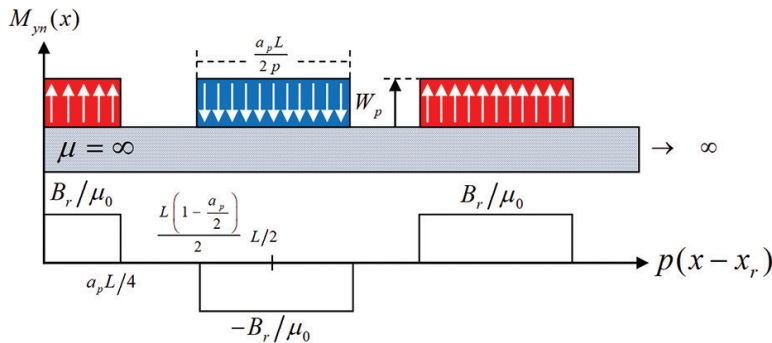


Fig. 2. Parallel magnetization Pattern. Note that $M_x(x)=0$.

where B_r is PM remanence flux density, a_p is the PM width per pole pitch ratio, M_{yh} is the h th element of $M_y(x)$, and $x_r = x - x_0$ is the spatial position of the mover where x_0 is its initial position.

In the case of open-circuit field prediction, if $n=hp$ the homogeneous and particular solutions of the Poisson's equation in the permanent magnet area are expressed as

$$\mathbf{A}_{z,h}^m(x,y) = \sum_{n=1}^{\infty} \left\{ \left[a_n^m e^{\frac{2n\pi}{L}y} + b_n^m e^{-\frac{2n\pi}{L}y} \right] \times \sin\left(\frac{2n\pi}{L}x\right) + \left[c_n^m e^{\frac{2n\pi}{L}y} + d_n^m e^{-\frac{2n\pi}{L}y} \right] \times \cos\left(\frac{2n\pi}{L}x\right) \right\} \quad (20)$$

$$\mathbf{A}_{z,p}^m(x,y) = \sum_{n=1}^{\infty} \left\{ \left[\mu_0 \frac{L}{2h\pi} M_{yh} \sin\left(\frac{2h\pi}{L}x_r\right) \cos\left(\frac{2n\pi}{L}x\right) \right] + \left[-\mu_0 \frac{L}{2h\pi} M_{yh} \cos\left(\frac{2h\pi}{L}x_r\right) \sin\left(\frac{2n\pi}{L}x\right) \right] \right\} \quad (21)$$

Applying the boundary condition (9) yields

$$\mathbf{A}_z^m(x,y) = \sum_{n=1}^N \left\{ \left[\frac{-\mu_0 L}{2n\pi} M_{yh} \cos\left(\frac{2n\pi}{L}x_r\right) + a_n^m \left[e^{\frac{2n\pi}{L}y} + e^{\frac{4n\pi}{L}W_r} e^{-\frac{2n\pi}{L}y} \right] \right] \times \sin\left(\frac{2n\pi}{L}x\right) + \left[\frac{\mu_0 L}{2n\pi} M_{yh} \sin\left(\frac{2n\pi}{L}x_r\right) + c_n^m \left[e^{\frac{2n\pi}{L}y} + e^{\frac{4n\pi}{L}W_r} e^{-\frac{2n\pi}{L}y} \right] \right] \times \cos\left(\frac{2n\pi}{L}x\right) \right\} \quad (22)$$

The solution of PDEs at the airgap and slots are expressed as

$$\sum_{n=1}^N \left\{ \left[a_n^a e^{\frac{2n\pi}{L}y} + b_n^a e^{-\frac{2n\pi}{L}y} \right] \times \cos\left(\frac{2n\pi}{L}x\right) + \left[c_n^a e^{\frac{2n\pi}{L}y} + d_n^a e^{-\frac{2n\pi}{L}y} \right] \times \sin\left(\frac{2n\pi}{L}x\right) \right\} \quad (23)$$

$$\mathbf{A}_z^{s,k}(x,y) = \sum_{u=1}^U \left\{ \left[a_u^{s,k} e^{\frac{u\pi}{L_s}y} + b_u^{s,k} e^{-\frac{u\pi}{L_s}y} \right] \times \cos\left(\frac{u\pi}{L_s}\left(x - x_c^k + \frac{L_s}{2}\right)\right) \right\} \quad (24)$$

For ease of the numerical computation, the equation (24) can be rewritten as

$$\mathbf{A}_z^{s,k}(x,y) = \sum_{u=1}^U \left\{ \left[a_u^{s,k} e^{\frac{-u\pi}{L_s}W_{bs}} e^{\frac{u\pi}{L_s}y} + b_u^{s,k} e^{\frac{u\pi}{L_s}W_{bs}} e^{-\frac{u\pi}{L_s}y} \right] \times \cos\left(\frac{u\pi}{L_s}\left(x - x_c^k + \frac{L_s}{2}\right)\right) \right\} \quad (25)$$

Applying the boundary condition (11) yields

$$\mathbf{A}_z^{s,k}(x,y) = \sum_{u=1}^U \left\{ a_u^{s,k} \left[e^{\frac{-u\pi}{L_s}W_{bs}} e^{\frac{u\pi}{L_s}y} + e^{\frac{u\pi}{L_s}(W_{bs}-W_s)} e^{-\frac{u\pi}{L_s}y} \right] \times \cos\left(\frac{u\pi}{L_s}\left(x - x_c^k + \frac{L_s}{2}\right)\right) \right\} \quad (26)$$

2.4. APPLYING BOUNDARY CONDITIONS TO CALCULATE INTEGRATION CONSTANTSTEGRATION CONSTANTS

Based on the obtained equations, the number of integration constants is $6N+N_sU$, where N and U are the number of harmonics used in the general solutions for magnet, air gap and slot areas, respectively. In order to calculate integration constants, the boundary conditions stated in (12-15) must be applied.

Using the boundary condition (12), yields

$$\frac{2n\pi}{L} a_n^m \left[e^{\frac{2n\pi}{L}W_m} + e^{\frac{2n\pi}{L}(2W_r-W_m)} \right] - \frac{2n\pi}{L} \left[c_n^a e^{\frac{2n\pi}{L}W_m} + d_n^a e^{-\frac{2n\pi}{L}W_m} \right] = \mu_0 M_{yn} \cos\left(\frac{2n\pi}{L}x_r\right) \quad (27)$$

$$\frac{2n\pi}{L} c_n^m \left[e^{\frac{2n\pi}{L}W_m} + e^{\frac{2n\pi}{L}(2W_r-W_m)} \right] - \frac{2n\pi}{L} \left[a_n^a e^{\frac{2n\pi}{L}W_m} + b_n^a e^{-\frac{2n\pi}{L}W_m} \right] = -\mu_0 M_{yn} \sin\left(\frac{2n\pi}{L}x_r\right) \quad (28)$$

Applying the boundary condition (13) yields

$$c_n^m \left[e^{\frac{2np\pi}{L}W_m} - e^{\frac{2np\pi}{L}(2W_r-W_m)} \right] - \left[a_n^a e^{\frac{2n\pi}{L}W_m} - b_n^a e^{-\frac{2n\pi}{L}W_m} \right] = 0 \quad (29)$$

$$a_n^m \left[e^{\frac{2np\pi}{L}W_m} - e^{\frac{2np\pi}{L}(2W_r-W_m)} \right] - \left[c_n^a e^{\frac{2n\pi}{L}W_m} - d_n^a e^{-\frac{2n\pi}{L}W_m} \right] = 0 \quad (30)$$

$$\begin{aligned} & \left(\frac{-u\pi}{L_s} \right) \left\{ a_u^{s,k} \left[e^{\frac{u\pi}{L_s}W_t} + 1 \right] \right\} \\ & + \sum_{n=1}^N \frac{2n\pi}{L} \left\{ \left[a_n^a e^{\frac{2n\pi}{L}W_t} + b_n^a e^{-\frac{2n\pi}{L}W_t} \right] \right. \\ & \times \frac{2}{L_s} \int_{x_c^k - \frac{L_s}{2}}^{x_c^k + \frac{L_s}{2}} \sin\left[\frac{u\pi}{L_s}\left(x - x_c^k + \frac{L_s}{2}\right)\right] \sin\left(\frac{2n\pi}{L}x\right) dx \\ & \left. - \left[c_n^a e^{\frac{2n\pi}{L}W_t} + d_n^a e^{-\frac{2n\pi}{L}W_t} \right] \right. \\ & \times \frac{2}{L_s} \int_{x_c^k - \frac{L_s}{2}}^{x_c^k + \frac{L_s}{2}} \sin\left[\frac{u\pi}{L_s}\left(x - x_c^k + \frac{L_s}{2}\right)\right] \cos\left(\frac{2n\pi}{L}x\right) dx \left. \right\} \end{aligned} \quad (31)$$

Applying the boundary condition (15) and using the correlation technique, yields

$$\begin{aligned} & \frac{2n\pi}{L} \left[-a_n^a e^{\frac{2n\pi}{L} W_t} + b_n^a e^{\frac{-2n\pi}{L} W_t} \right] \\ & + \sum_{k=1}^{N_s} \sum_{u=1}^U \frac{u\pi}{L_s} \left\{ a_{u,k}^s \left[e^{\frac{u\pi}{L_s} W_t} - 1 \right] \right. \\ & \left. \times \frac{2}{L_s} \int_{x_c^k - \frac{L_s}{2}}^{x_c^k + \frac{L_s}{2}} \cos \left[\frac{u\pi}{L_s} \left(x - x_c^k + \frac{L_s}{2} \right) \right] \cos \left(\frac{2n\pi}{L} x \right) dx \right\} = 0 \end{aligned} \quad (32)$$

$$\begin{aligned} & \frac{2n\pi}{L} \left[-c_n^a e^{\frac{2n\pi}{L} W_t} + d_n^a e^{\frac{-2n\pi}{L} W_t} \right] \\ & + \sum_{k=1}^{N_s} \sum_{u=1}^U \frac{u\pi}{L_s} \left\{ a_{u,k}^s \left[e^{\frac{u\pi}{L_s} W_t} - 1 \right] \right. \\ & \left. \times \frac{2}{L_s} \int_{x_c^k - \frac{L_s}{2}}^{x_c^k + \frac{L_s}{2}} \cos \left[\frac{u\pi}{L_s} \left(x - x_c^k + \frac{L_s}{2} \right) \right] \sin \left(\frac{2n\pi}{L} x \right) dx \right\} = 0 \end{aligned} \quad (33)$$

Finally, since $\mathbf{B} = \nabla \times \mathbf{A}$, the air-gap MFD can be expressed as

$$\begin{aligned} B_y^a(x, y) = & \sum_{n=1}^N \left\{ \frac{2n\pi}{L} \left[a_n^a e^{\frac{2n\pi}{L} W_{gm}} + b_n^a e^{\frac{-2n\pi}{L} W_{gm}} \right] \times \sin \left(\frac{2n\pi}{L} x \right) \right. \\ & \left. + \frac{2n\pi}{L} \left[c_n^a e^{\frac{2n\pi}{L} W_{gm}} + d_n^a e^{\frac{-2n\pi}{L} W_{gm}} \right] \times \cos \left(\frac{2n\pi}{L} x \right) \right\} \end{aligned} \quad (34)$$

$$\begin{aligned} B_x^a(x, y) = & \sum_{n=1}^N \left\{ \frac{2n\pi}{L} \left[a_n^a e^{\frac{2n\pi}{L} W_{gm}} - b_n^a e^{\frac{-2n\pi}{L} W_{gm}} \right] \times \cos \left(\frac{2n\pi}{L} x \right) \right. \\ & \left. + \frac{2n\pi}{L} \left[c_n^a e^{\frac{2n\pi}{L} W_{gm}} - d_n^a e^{\frac{-2n\pi}{L} W_{gm}} \right] \times \sin \left(\frac{2n\pi}{L} x \right) \right\} \end{aligned} \quad (35)$$

where W_{gm} is the air-gap middle width.

3. CASE STUDIES

Two slotted PMLSMs with parallel magnets have been considered to clarify the functionality of the presented approach. The results obtained from FEM and analytical method are compared.

FEM is a powerful numerical method for the analysis of electrical machines, in which the problem domain is discretized into finite elements by meshing and the governing equation for each element is solved numerically. Therefore, the quality of the mesh is critical in FEA and directly affects the accuracy of the results. To increase the accuracy, a denser mesh pattern should be used for the air gap region. ANSYS MAXWELL was used to model and analyze the considered PMLSMs.

The parameters of the considered PMLSMs are given in Table 1. The perpendicular and tangential components of the open-circuit magnetic flux density at the air-gap centerline are illustrated in Figs. 3–6. As it is obvious, the results obtained from the proposed method are very similar to that of obtained by the FEM. Figs. 3 and 4 illustrate the obtained results, respectively, for two different mover positions of the first studied motor, i.e. $x_0=0$ and $x_0=L/3$. Similar results for the second studied motor are shown in Figs. 5 and 6 at different mover positions, i.e. $x_0=0$ and $x_0=L/5$.

Table 1. Parameters of the considered PMLSMs

Parameter	Explanation	Value (1st motor)	Value (2nd motor)	Unit
W_r	Mover core thickness	0.012	0.012	(m)
W_p	PM height	0.005	0.005	(m)
W_g	Air-gap length	0.001	0.001	(m)
W_s	Slots height	0.025	0.025	(m)
L	Motor length	0.216	0.2	(m)
B_r	PM remanence flux density	1.23	1.23	(T)
N_s	Number of stator slots	9	12	-
p	Number of pole pairs	4	2	-
a_s	Slot width to slot pitch ratio	0.54	0.6	-
a_p	PM width to pole pitch ratio	0.7	0.8	-

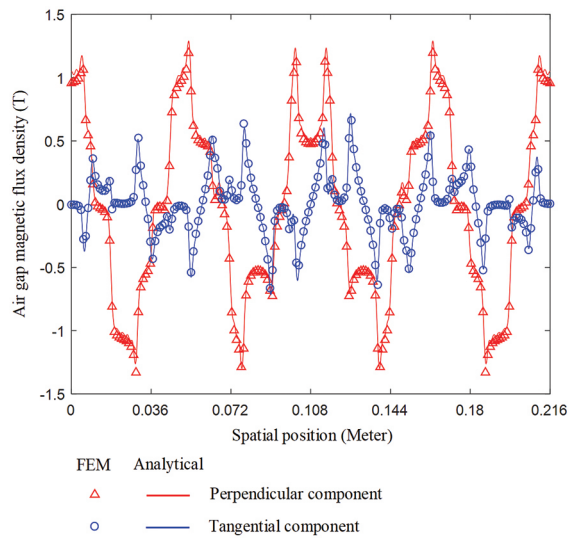


Fig. 3. Components of air-gap MFD obtained for the first linear motor ($x_0=0$)

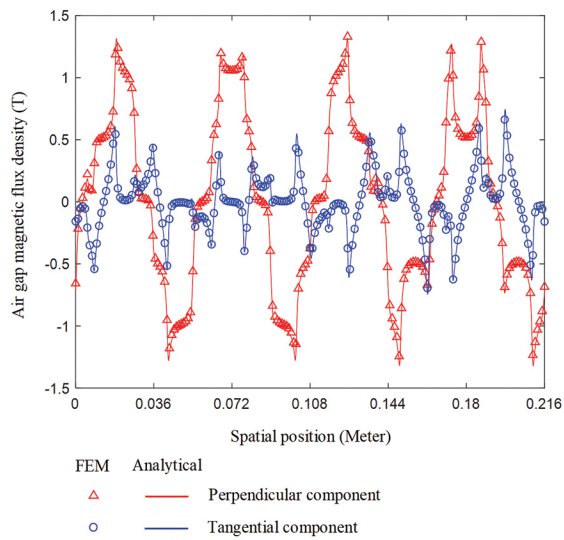


Fig. 4. Components of air-gap MFD obtained for the first linear motor ($x_0=L/3$)

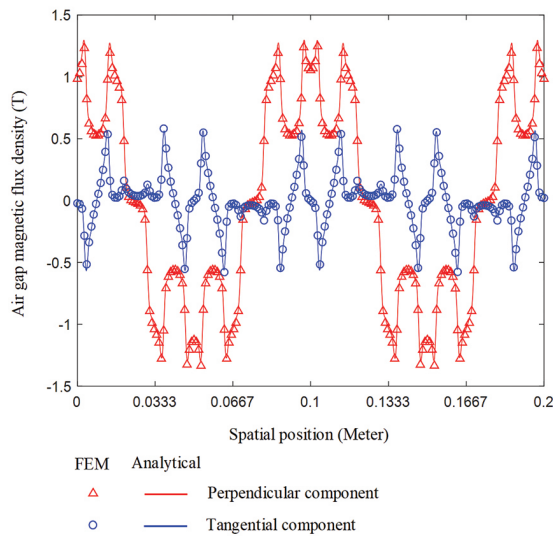


Fig. 5. Components of air-gap MFD obtained for the second linear motor ($x_0=0$)

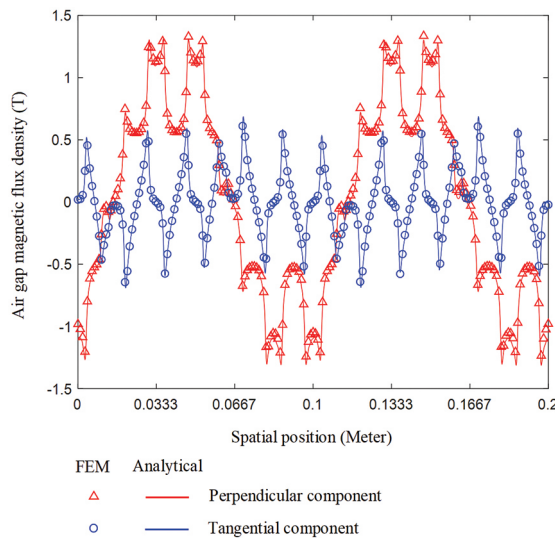


Fig. 6. Components of air-gap MFD obtained for the second linear motor ($x_0=L/5$)

4. CONCLUSIONS

Analytical methods based on mathematical equations can be an effective tool in optimization processes due to the reduction of the computational volume, provided that the simplifications made do not sacrifice the accuracy of the problem. In this paper, open-circuit magnetic field is calculated analytically for slotted PMLSMs with surface-mounted magnets. The parallel magnetization pattern is used and the Poisson's/ Laplace's equations are expressed for all regions to calculate the magnetic field distribution. Two different PMLSMs with (8-pole, 9-slot) and (4-pole, 12-slot) are considered as case studies and the results are verified by FEM. These solutions can also be used to calculate the crucial quantities of the machine such as back-EMF and cogging force. For further studies, it is recommended that this method be extended to armature reaction field calculations.

5. REFERENCES:

- [1] Y. Zhou, R. Qu, D. Li, Y. Gao, C. H. T. Lee, "Performance investigation and improvement of linear vernier permanent magnet motor for servo application", *IEEE/ASME Transactions on Mechatronics*, Vol. 28, No. 5, 2023, pp. 2657-2669.
- [2] K. H. Kim, D. K. Woo, "Linear tubular permanent magnet motor for an electromagnetic active suspension system", *IET Electric Power Applications*, Vol. 19, No. 12, 2021, pp. 1648-1665.
- [3] Y. Shen, Z. Zeng, Q. Lu, C. H. T. Lee, "Design and analysis of double-sided flux concentrated permanent magnet linear machine with saturation relieving effect", *IEEE Transactions on Industrial Electronics*, Vol. 70, No. 10, 2023, pp. 10442-10453.
- [4] F. Cui, Z. Sun, W. Xu, W. Zhou, Y. Liu, "Comparative analysis of bilateral permanent magnet linear synchronous motors with different structures", *CES Transactions on Electrical Machines and Systems*, Vol. 4, No. 2, 2020, pp. 142-150.
- [5] X. Liu, J. Gao, S. Huang, K. Lu, "Magnetic field and thrust analysis of the U-channel air-core permanent magnet linear synchronous motor", *IEEE Transactions on Magnetics*, Vol. 53, No. 6, 2017.
- [6] D. Pan, L. Li, M. Wang, "Modeling and optimization of air-core monopole linear motor based on multiphysical fields", *IEEE Transactions on Industrial Electronics*, Vol. 65, No. 12, 2018, pp. 9814-9824.

- [7] Z. Li et al. "Hybrid analytical model of permanent magnet linear motor considering iron saturation and end effect", *IEEE Transactions on Energy Conversion*, Vol. 39, No. 3, 2024, pp. 2008-2017.
- [8] B. Li, J. Zhang, X. Zhao, Z. Miao, Z., H., H. Li, "Magnetic Field Analysis of Trapezoidal Halbach Permanent Magnet Linear Synchronous Motor Based on Improved Equivalent Surface Current Method", *Applied Computational Electromagnetics Society Journal*, Vol. 40, No. 1, 2025, pp. 69-78.
- [9] E. Kazan, A. Onat, "Modeling of air core permanent-magnet linear motors with a simplified nonlinear magnetic analysis", *IEEE Transactions on Magnetics*, Vol. 47, No. 6, 2011, pp. 1753-1762.
- [10] A. Rahideh, A. Ghaffari, A. Barzegar, A. Mahmoudi, "Analytical model of slotless brushless pm linear motors considering different magnetization patterns", *IEEE Transactions on Energy Conversion*, Vol. 33, No. 4, 2018, pp. 1797-1804.
- [11] H. Hu, J. Zhao, X. Liu, Y. Guo, "Magnetic field and force calculation in linear permanent-magnet synchronous machines accounting for longitudinal end effect", *IEEE Transactions on Industrial Electronics*, Vol. 63, No. 12, 2016, pp. 7632-7643.
- [12] S. G. Min, B. Sarlioglu, "3-D performance analysis and multiobjective optimization of coreless-type PM linear synchronous motors", *IEEE Transactions on Industrial Electronics*, Vol. 65, No. 2, 2017, pp. 1855-1864.
- [13] K. Atallah, Z. Q. Zhu, D. Howe, "Armature reaction field and winding inductances of slotless permanent-magnet brushless machines", *IEEE Transactions on Magnetics*, Vol. 34, No. 5, 1998, pp. 3737-3744.
- [14] B. Gysen, K. Meessen, J. Paulides, E. Lomonova, "General formulation of the electromagnetic field distribution in machines and devices using Fourier analysis", *IEEE Transactions on Magnetics*, Vol. 46, No. 1, 2009, pp. 39-52.

Optimal Power Control Using Modified Perturb and Observe Algorithm for Photovoltaic System Under Partial Shading

Original Scientific Paper

Afrilia Surya Andini*

State Polytechnic of Malang,
Department of Electrical Engineering
Jl. Soekarno Hatta No. 9, Malang, Indonesia
2041150035@student.polinema.ac.id

Ratna Ika Putri

State Polytechnic of Malang,
Department of Electrical Engineering
Jl. Soekarno Hatta No. 9, Malang, Indonesia
ratna.ika@polinema.ac.id

Ika Noer Syamsiana

State Polytechnic of Malang,
Department of Electrical Engineering
Jl. Soekarno Hatta No. 9, Malang, Indonesia
ikanoersyamsiana@polinema.ac.id

*Corresponding author

Gery Prasetya

State Polytechnic of Malang,
Department of Electrical Engineering
Jl. Soekarno Hatta No. 9, Malang, Indonesia
geryprasetya964@gmail.com

Achsanul Khabib

State Polytechnic of Malang,
Department of Electrical Engineering
Jl. Soekarno Hatta No. 9, Malang, Indonesia
achsanul.khabib@polinema.ac.id

Abstract – Implementation of photovoltaic systems encounters problems, particularly concerning Partial Shading Conditions (PSC), solar irradiance, and temperature, which influence the generated output power. The PSC can diminish the power efficiency of the photovoltaic system. Consequently, a controller is required to optimize the photovoltaic system's power output by considering the power supply characteristics. This paper discusses optimal power control in photovoltaic system under PSC. The proposed method employs a Modified Perturb and Observe (MP&O) algorithm based on the observation of current and voltage output from the photovoltaic system. The MP&O algorithm is integrated into a microcontroller and will provide PWM signals to operate the synchronous buck converter. Testing was performed under PSC. The experimental results indicated that the synchronous buck converter achieved a performance efficiency of 85%. The efficacy of the MP&O algorithm was evaluated without the MPPT method and conventional P&O algorithm. The MP&O algorithm outperformed compared to without MPPT method and conventional P&O algorithm. The MP&O algorithm yielded more consistent output power and necessitated a quicker tracking duration. The proposed method achieves an average output power efficiency of 84%; in contrast, without the MPPT method, it only reached 57%, and with the conventional P&O algorithm, it attains an efficiency of just 70%.

Keywords: Photovoltaic, Optimal Control, Perturb & Observe, Partial Shading

Received: December 31, 2024; Received in revised form: June 12, 2025; Accepted: June 30, 2025

1. INTRODUCTION

The use of renewable energy as a source of electrical energy continues to increase every year due to the increasing awareness of the use of environmentally friendly energy and the reduction of fossil energy. Using renewable energy is one way to address climate change by reducing carbon emissions [1]. One of the

most commonly used renewable energy sources in Indonesia nowadays is solar energy, which is converted into electrical energy through photovoltaic systems. Photovoltaic systems are one of the ideal power plants to be developed in Indonesia because Indonesia is located on the equator, which can receive sunlight throughout the year. However, photovoltaic systems have the main problem of low efficiency in converting

electrical energy, with the generated electrical power affected by environmental conditions and uneven lighting, also known as Partial Shading Condition (PSC) [2-4]. The PSC occurs due to shadows from particular objects such as buildings, trees, or dust that partially cover the photovoltaic, thus reducing the power generated from the photovoltaic [5]. Moreover, PSC makes the photovoltaic module unbalanced, resulting in many peaks in the P-V curve, making it challenging to reach Maximum Power Point (MPP). This condition causes a decrease in photovoltaic efficiency of up to 70%, as a result of which the entire performance of the photovoltaic system will be affected [6]. Thus, photovoltaic systems require control to improve system efficiency in the face of PSC [7].

Increasing photovoltaic efficiency can be done through optimal power regulation, which is done by adjusting the duty cycle of the connected power converter. Several power converters are used in photovoltaic systems, including a DC-DC converter. DC-DC converters play an important role in renewable energy [8]. Nevertheless, power converters try using fewer parts, namely capacitors and inductors. The voltage spikes result from the elimination, causing the converter's design to be more complicated [9]. Several power converters that can be used, including buck converters. A buck converter, or step-down converter, is one type of DC-DC converter that can convert voltage from a high level to a lower level. Buck converters are ideal when implemented on DC current-based systems [10, 11]. Buck converters installed in photovoltaic systems provide stability and quick response during transient circumstances, even though they are susceptible to instability during voltage drops. This makes the efficiency of the buck converter not optimal [12]. The synchronous buck converter is a buck converter that uses two MOSFETs; replacing the diode with a MOSFET can reduce conduction losses and improve voltage stability at the output [13]. On the secondary side, using MOSFET to replace a diode can increase converter efficiency and reduce voltage spikes, and MOSFET can last a long time [14]. To get high efficiency, it is necessary to regulate the duty cycle of the synchronous buck converter. Several methods have been developed for managing the duty cycle of this power converter, called the Maximum Power Point Tracking (MPPT) method. In its implementation, this MPPT method is embedded in an embedded system to regulate the converter's performance [15, 16]. One of the most widely used conventional MPPT methods is the Perturb and Observe (P&O) algorithm, which can be implemented cheaply [17, 18]. However, this algorithm continues to be trapped at the Local Maximum Power Point (LMPP), which happens under PSC so that it cannot reach the Global Maximum Power Point (GMPP) [19]. Compared to the Particle Swarm Optimization (PSO) algorithm, the P&O algorithm is faster in determining the optimum power but solves the steady state [20]. The P&O algorithm experiences drift problems during rapid changes in resistive loads because it is hampered in overcoming power loss problems, so that the P&O algorithm can produce oscillations [21-23]. To im-

prove the performance of the P&O algorithm, the conventional method is modified by providing constraints on specific parameters to enhance the method's performance in PSC. This modification makes the system adapt faster to rapidly changing environmental conditions and reduces the tracking time needed to reach MPP. A modification of the P&O algorithm has been developed for photovoltaic systems, where the step size is not constant. Still, it can change accordingly based on changes in the slope of photovoltaic characteristics. Based on simulation results, it performs better with the same tracking time as P&O and more minor oscillations. However, this algorithm has not considered the PSC [24].

This paper will explain the application of Modified Perturb and Observe (MP&O) to photovoltaic systems using synchronous buck converters under PSC. A synchronous buck converter is designed and tested experimentally. The MP&O algorithm is embedded in the microcontroller, and the performance will be compared without the MPPT method and with the conventional P&O algorithm on the same PSC. The use of a microcontroller in this equipment will produce a reliable and economically valuable system.

2. METHODOLOGY

Fig. 1 shows the block diagram of the photovoltaic system, consisting of a photovoltaic, switch, synchronous buck converter, current sensor, voltage sensor, loads, and embedded system. The output of the photovoltaic module will be connected to the synchronous buck converter. The voltage sensor and current sensor will detect the photovoltaic current and voltage output. An embedded system in the form of a microcontroller functions as a controller. The MP&O algorithm is embedded in the microcontroller, creating a duty cycle to drive the synchronous buck converter. The microcontroller will read the photovoltaic module output voltage and current and calculate its output power based on it. Based on the output power generated by the photovoltaic module, the MP&O algorithm will determine the converter's duty cycle so that the system can work at the maximum power point. For the system's safety, the microcontroller will set the input switch that connects the photovoltaic module with the converter and the load switch that connects the converter and the load. Suppose the output voltage of the photovoltaic module is low and cannot supply the load. In the case, the microcontroller will activate the input switch to disconnect the photovoltaic module from the synchronous buck converter.

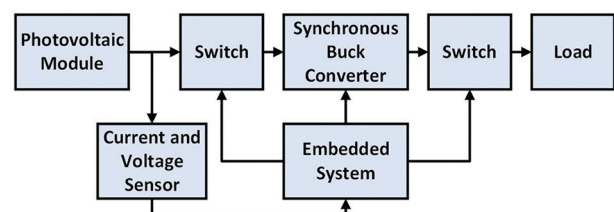


Fig. 1. System Block Diagram

This paper uses a 200 Wp polycrystalline photovoltaic module that is commercially available for use. Photovoltaic converts solar irradiation into electrical energy through electrons, attracting semiconductor materials such as monocrystalline and polycrystalline. Photovoltaic modules are composed of main components such as current sources, diodes, and resistors connected in parallel and series, as shown in Fig. 2 [21]. The equivalent circuit diagram of the photovoltaic module supports the design of converters and also MPPT methods. This ensures that the designed system can operate efficiently according to environmental conditions. The current source (I_{ph}) indicates the current produced by solar energy. The diode (d) is represented with the leakage current in the photovoltaic module, which enables the diode current (I_d) to flow when forward biased. The parallel or shunt resistance (R_p) symbolizes the leakage current in the module, with the current through it represented as I_p . The series resistor (R_s) represents the interval resistive losses within the module and its connections. The output current (I_{pv}) from the module is the current delivered to the load. V_{pv} describes the voltage at the terminals of the photovoltaic module.

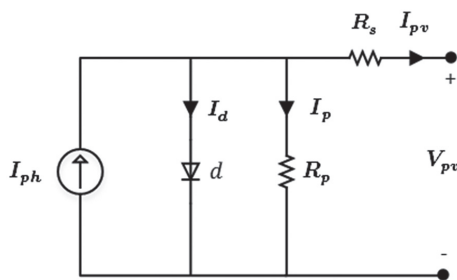


Fig. 2. PV Equivalent Circuit Diagram

Fig. 3 shows the I-V and P-V characteristic curves with an irradiance difference of 200 W/m² up to 1000 W/m² and a constant temperature of 25°C. The I-V curve indicates that the output current of the photovoltaic module is affected by solar irradiation; increased irradiation results in higher current production, and conversely, decreased irradiation results in lower current output. Similarly, the P-V curve indicates the power output is enhanced under higher irradiation conditions. Nonetheless, if the irradiation is ineffective, it cannot reach the peak value. The characteristic curve shows the MPP location; this curve is also used to design the MPPT method. This analysis aims to demonstrate the efficacy of the photovoltaic module utilized. Modeling and simulation with MATLAB/Simulink related to the module specification data used to comprehend the relationship between current, voltage, output, and features. The red curves denote maximum conditions, whilst the blue curves indicate lower irradiance.

PSC occurs because the photovoltaic module is partially covered by shadows from buildings, trees, and dust. Thus, the maximum power generated under PSC becomes non-uniform. In the photovoltaic characteristic curve shown in Fig. 3, the MPP has different variations and depends on

environmental conditions. However, the MPPT method is designed to track the MPP dynamically, even under PSC, providing reassurance of its effectiveness [19].

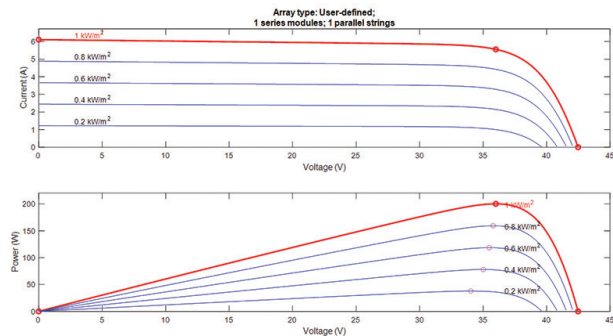


Fig. 3. I-V and P-V Characteristic Curve Photovoltaic Module with Different Irradiation

2.1. SYNCHRONOUS BUCK CONVERTER

When the photovoltaic module is covered with shadow, the power produced by the module is significantly reduced. Thus, a converter with higher efficiency is needed to minimize power loss. The synchronous buck converter is a modification of the buck converter to reduce the voltage from a higher level to a lower level; in this synchronous buck converter, the function of the diode is replaced by Metal Oxide Semiconductor Field Effect Transistor (MOSFET). This replacement aims to increase the efficiency of the power loss resulting from thermal performance [13].

The synchronous buck converter consists of principal components, namely inductors, capacitors, and MOSFET, as a switch that aims to reduce the voltage from a higher level to a lower level. In its operation, the synchronous buck converter operates using two switches, so-called MOSFET, whose circuit can be seen in Fig. 4. On the circuit diagram to show how the components are interconnected to facilitate the implementation of the converter and assist in simulating the converter before it is implemented. To achieve high efficiency, it is divided into two modes; in an active mode, where MOSFET 1 is on and MOSFET 2 is off, the current from the input source will pass through the inductor to supply energy to the load, which makes the inductor current increase. In discharge mode where MOSFET 1 off and MOSFET 2 on, the energy stored by the inductor in active mode will flow to the load, which decreases the inductor current, but the load still gets the energy supply [25, 26].

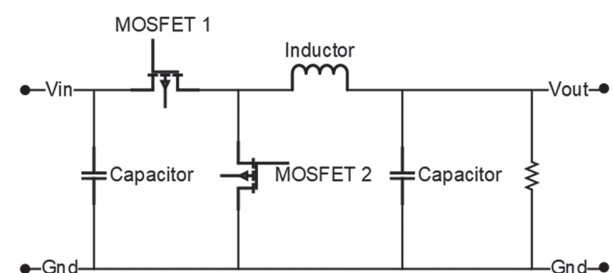


Fig. 4. Synchronous Buck Converter Circuit Diagram

The output voltage produced by the synchronous buck converter (V_{out}) determined based on the duty cycle (D) given to the MOSFET. The V_{out} can be determined using the following equation.

$$V_{out} = D \cdot V_{in} \quad (1)$$

The converter input voltage (V_{in}) is the output voltage of photovoltaic module with a maximum voltage 36 V, and duty cycle dynamically adjusted to keep the V_{out} constant at 15 V, even with fluctuations in the V_{in} . The selection of inductor (L) with inductor current ripple (ΔI_L) is limited to 35% if the reduced inductance value causes an increase in the peak current within the inductor, causing it to operate over its specified limitations. This results in a reduction in the inductor's performance inside the system. Consequently, it is necessary to evaluate the actual dimensions of the inductor utilizing the following equation.

$$L = \frac{V_{out} \cdot (V_{in} - V_{out})}{\Delta I_L \cdot f_s \cdot V_{in}} \quad (2)$$

Where is the switching frequency (f_s) of 39 kHz was chosen because it is stable, smooth, and efficient for power regulation. Based on the equation (2), the specified inductance value is 60 μ H. Due to the high inductance value, it will reduce the ripple current even though it increases the component size.

Input capacitors are crucial to stabilizing the input voltage for peak current demands during duty cycle switching and mitigating voltage fluctuations caused by fast current variations. The output voltage ripple (ΔV_o) value is set to 5% to reduce the capacitor size. The following equation determines the capacitance value.

$$C = \frac{\Delta I_L}{8 \cdot f_s \cdot \Delta V_{out}} \quad (3)$$

Based on equation (3), using the Electrolytic Capacitor (ELCO) type, the specified capacitance value is 470 μ F. The ELCO type is selected because it has a high capacitance ratio, suitable for filtering and energy storage. This capacitor effectively minimizes output voltage ripple and supports transient current demands on switching. Furthermore, the selection of N-channel MOSFET IRFP4110 as the switching component is because it can handle continuous currents up to 120 A with adequate cooling and peak current up to 670 A for a short duration. Fig. 5 shows the built synchronous buck converter.

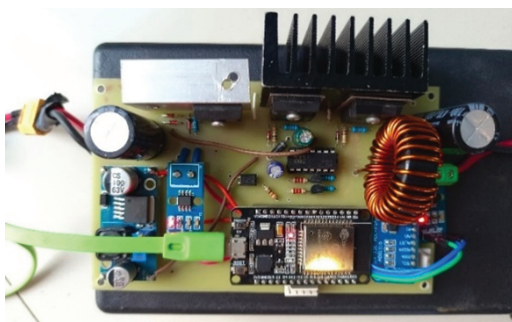


Fig. 5. Synchronous Buck Converter

2.2. MODIFIED PERTURB AND OBSERVE

The MPPT method is used to improve the energy efficiency of photovoltaic systems [17]. Increasing the efficiency of photovoltaic systems in PSC can be done through duty cycle converter settings using MPPT algorithms. One of the widely implemented MPPT algorithms is P&O. P&O has a high tracking speed and lower computational complexity compared to metaheuristic algorithms such as Particle Swarm Optimization (PSO) and Firefly Algorithm (FA) [23]. The P&O algorithm has advantages in simplicity and ease of implementation. However, the P&O algorithm has limitations, such as the step size selection and possible oscillations around the MPP when the step size is not appropriate. Therefore, although the P&O algorithm is simple, it does not provide the same tracking accuracy and stability as other algorithms [27]. Fig. 6 shows the flowchart for the P&O algorithm. The duty cycle change in this algorithm depends on the step size with a value of 0.05.

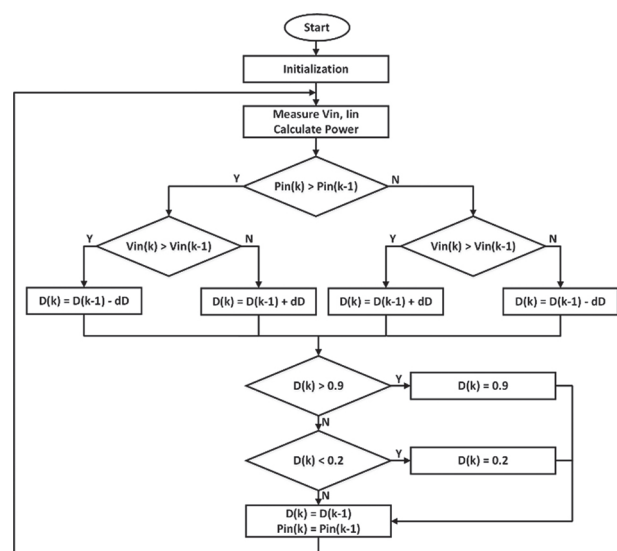


Fig. 6. Conventional P&O Algorithm Flowchart

In this paper, a Modified P&O (MP&O) algorithm is carried out to increase the efficiency of the photovoltaic system. The MP&O algorithm is expected to reduce the oscillation in a steady state and accelerate the achievement of MPP in PSC. Since the MPPT method can prevent power loss by stabilizing power fluctuations during extreme weather conditions, it is important to improve photovoltaic systems. The performance of the modified P&O algorithm will be compared with the conventional P&O algorithm.

Efficiency (Eff) evaluates a method's effectiveness of by comparing the output power produced by the proposed method to the output power real generated by the photovoltaic module, represented as a percentage. The proposed methods tested are divided into three parts: without the MPPT method, with the P&O algorithm, and with the MP&O algorithm. A near 100% efficiency result signifies the method's success in power

utilization, whilst a low efficiency value denotes its ineffectiveness in power optimization. Efficiency can be determined using following equation.

$$Eff = \frac{\text{Output Power Proposed Method}}{\text{Output Power Real Photovoltaic}} \quad (4)$$

Fig. 7 shows the flowchart of the MP&O algorithm, which generates the duty cycle that will be sent to the synchronous buck converter. The algorithm works based on the measuring of the output current and voltage at the photovoltaic module, which is used to calculate the photovoltaic output power (P). The change in output power will determine the duty cycle sent to the converter, where the following equation determines the change in power (dP)

$$dP = P(k) - P(k-1) \quad (5)$$

Where $P(k)$ is current power and $P(k-1)$ is previous power. There are several possibilities for the MP&O algorithm.

- If power reaches the MPP then $dP < \beta$, the duty cycle value is fixed $D(k) = D(k-1)$.
- If power $P(k) > P(k-1)$ and duty cycle $D(k) > D(k-1)$ or $P(k) < P(k-1)$ and $D(k) < D(k-1)$, the duty cycle must be increased $D(k) = D(k-1) + dD$.
- If power $P(k) > P(k-1)$ and duty cycle $D(k) < D(k-1)$ or $P(k) < P(k-1)$ and $D(k) > D(k-1)$, the duty cycle must be decreased $D(k) = D(k-1) - dD$.

dD is a step size that will change depending on the iteration. At each iteration the dD value will be updated by $dD(k) = dD(k-1) - C$. Where C is a constant value. This paper C has two values, namely $C1 = 0.02$ and $C2 = 0.03$. The more iterations, the smaller the dD value because it is close to the optimum value to reduce oscillations in the steady state.

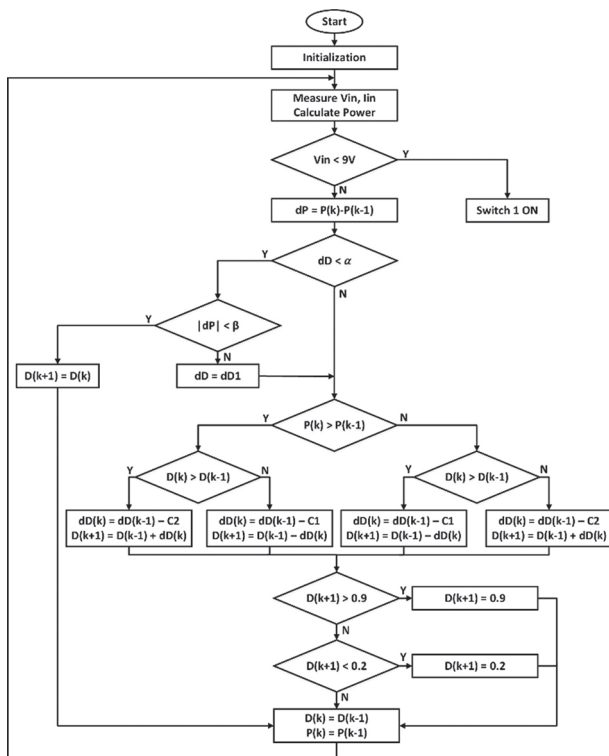


Fig. 7. Modified P&O Flowchart

3. RESULTS AND DISCUSSION

The designed and built photovoltaic system was tested experimental testing to assess the performance of both the overall system and the integrated modified P&O algorithm within the embedded system. The tests were carried out in several stages: photovoltaic module, synchronous buck converter, and whole system. Photovoltaic testing is done by shading the photovoltaic without a synchronous buck converter circuit. The converter testing aims to evaluate the performance of the synchronous buck converter. This test is carried out by changing the duty cycle and input voltage, so that the efficiency of the circuit can be evaluated. The overall system test evaluates the performance of photovoltaic systems equipped with optimal power control using the modified P&O algorithm. Overall system testing is done by providing three PSCs. The three conditions are 0% shading, 30% shading, and 50% shading.

When the synchronous buck converter circuit and resistive load are connected directly, the characteristic curve of the photovoltaic output power test results under PSC is displayed in Fig. 8. This photovoltaic test involves varying the resistive load and testing the photovoltaic module in both shading and without shading environments. Testing of photovoltaic modules is done to find the MPP produced by photovoltaic module under various shading scenarios in the field, such as building shading. The resistive load is also adjusted to assess the photovoltaic reaction to changes in the load connected to the system. Because shading affects the output power produced by photovoltaic module, the test findings demonstrate that the output power of a photovoltaic module can drop when shading is present. The photovoltaic module may provide up to 36.5 W of power. The graphs from the testing show that the MPP changes in response to the load resistance value and irradiation level. Therefore, in order to maximize the output power under PSC, the MPPT approach is required.

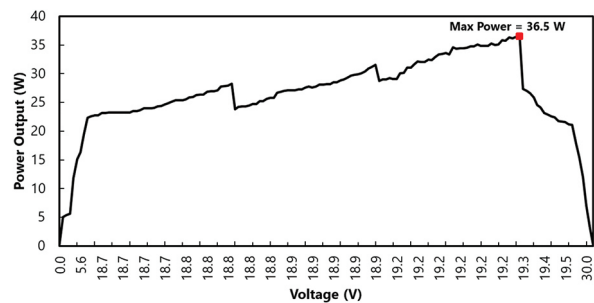


Fig. 8. Photovoltaic Characteristic Curve with Shading Based on Test

Fig. 9 shows the experiment setup for testing the synchronous buck converter in the laboratory. The test is conducted by changing the duty cycle and input voltage. The synchronous buck converter output voltage and duty cycle are displayed on the oscilloscope equipment, as shown in Fig. 10. The graph on the oscilloscope shows the shape of the voltage at a particular duty cycle.



Fig. 9. Converter Testing Process

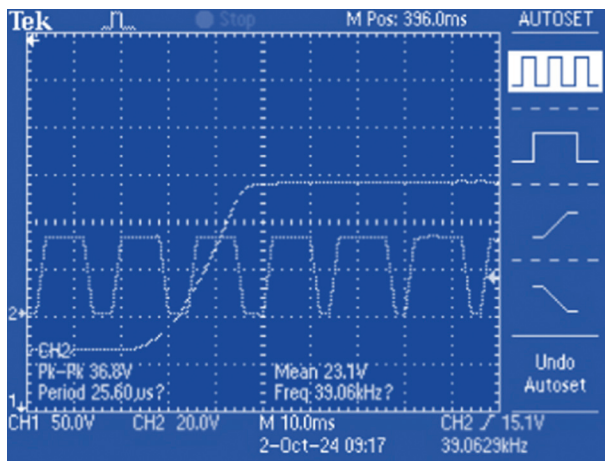


Fig. 10. Oscilloscope Graph of Synchronous Buck Converter Response

Based on the test results of the synchronous buck converter in Fig. 11 shows the performance generated from this converter. Given a voltage of 40 V DC, the greater the duty cycle value, the higher the output voltage produced. This is because the switch has a longer active time, while the smaller the duty cycle value will reduce the active time.

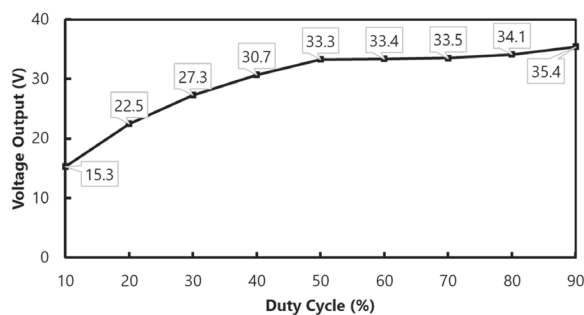


Fig. 11. Voltage Output of Synchronous Buck Converter with 40 V Input Voltage

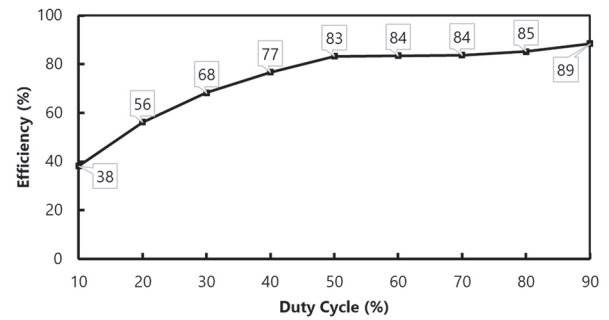


Fig. 12. Efficiency of Synchronous Buck Converter with Duty Cycle Variations

Overall, system testing is conducted to test the performance of the MP&O algorithm embedded in the microcontroller. The MP&O algorithm is based on the measurement of current and voltage generated by the photovoltaic by the current and voltage sensors. The microcontroller will read the output of the current and voltage sensors and calculate the power generated based on the measurement results. The output of the MP&O algorithm is the duty cycle. The microcontroller will send a Pulse Width Modulation (PWM) signal with the duty cycle to drive the switching components in the synchronous buck converter. Through this duty cycle setting, the photovoltaic system will produce optimal power. The performance of MP&O is compared with the performance of the without MPPT method and with the conventional P&O algorithm.

Fig. 13 shows the output power of the photovoltaic system without the MPPT algorithm, conducted under PSC divided into three scenarios: 0%, 30%, and 50% shading. The average output power without shading is 23.8 W. Shading reduces power; specifically, under 30% shading, the average power is 20.7 W, while 50% shading further drops it to 20 W.

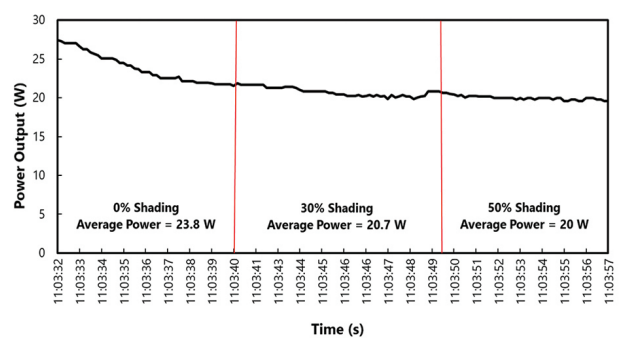


Fig. 13. Power Output Without the MPPT Method

Fig. 14 shows the output power of the photovoltaic system by applying the conventional P&O algorithm under PSC. PSC changes are given in several conditions: the initial condition with 0% shading, then given shading of 30%, and then 50% shading. The average output power with 0% shading condition is 28 W, with 30% shading is 25 W. In the 50% shading condition, the photovoltaic system with the P&O algorithm will produce an average output power of 23.4 W.

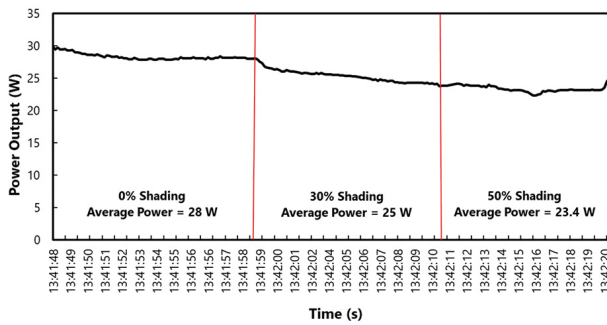


Fig. 14. Power Output Using Conventional P&O

Fig. 15 shows the output power of the photovoltaic system using modified P&O algorithm under PSC, namely with 0% shading, 30% shading and 50% shading. The average output power without shading condition is 32.6 W; with 30% shading, it is 31 W, and at 50% shading, the photovoltaic system with MP&O will produce an average output power of 28 W.

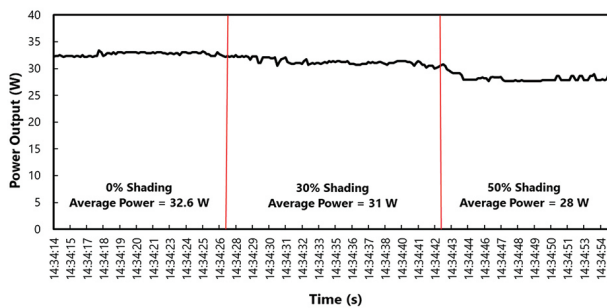


Fig. 15. Power Output Using Modified P&O

The performance of the photovoltaic system with modified P&O is compared with the without MPPT method and P&O algorithm under PSC. The output power of the photovoltaic system (P_{out}) without any MPPT method is lower than that of the conventional P&O algorithm and modified P&O algorithm. In addition, without the MPPT method, the system lacks the ability to maintain the drop required to optimize the photovoltaic performance. The modified P&O algorithm is more stable with lower oscillations than without the MPPT method and with the conventional P&O algorithm. Efficiency is defined as the ratio of the output power extracted by the modified P&O method, which is the suggested way, to the without MPPT method and with conventional P&O algorithm as a comparison method with maximum output power of the photovoltaic module. Under some circumstances, the efficiency formula used to evaluate the proposed method may be the most effective way to harvest power from photovoltaic modules. The algorithm's efficiency is calculated by the equation (4). Table 1 shows the efficiency comparison between without the MPPT method, with conventional P&O algorithm, and with modified P&O under PSC. Modified P&O algorithm has a higher efficiency; the average efficiency is 84%, while without the MPPT method is 57%, and using conventional P&O algorithm is 70%.

Table 1. Efficiency of Proposed Method

Shading	Without MPPT		Conventional P&O		Modified P&O	
	P _{out} (W)	Eff (%)	P _{out} (W)	Eff (%)	P _{out} (W)	Eff (%)
0%	23.8	65	28	77	32.6	89
30%	20.7	56	25	69	31	85
50%	20	55	23.4	64	28	77

4. CONCLUSION

This paper describes applying the modified P&O algorithm on a photovoltaic system with PSC to enable optimal power production. The modified P&O algorithm is integrated into a microcontroller that will adjust the duty cycle of the PWM signal transmitted to the synchronous buck converter. Optimal power management in the photovoltaic system is attained by adjusting the duty cycle transmitted to the synchronous buck converter. The effectiveness testing findings indicate that the synchronous buck converter achieves an efficiency of 85%. The efficacy of the modified P&O algorithm is juxtaposed with that of the conventional P&O algorithm. The proposed method was evaluated under three shading conditions: 0% shading, 30% shading, and 50% shading. The proposed method yields more stable photovoltaic output power than conventional P&O algorithm under partial shading conditions and can significantly improve the power conversion efficiency. The efficiency of the photovoltaic system utilizing modified P&O algorithm is 84%, whereas the photovoltaic employing conventional P&O algorithm achieves an efficiency of 70% and without MPPT method, the efficiency is 57%. Further research will advance the photovoltaic system utilizing the suggested technology linked to alternating current loads and integrated with additional electrical energy sources or a hybrid renewable energy system.

5. ACKNOWLEDGMENT

The authors acknowledge the financial support provided by Directorate of Vocational Academic Higher Education Academic, Ministry of Education, Culture, Research, and Technology for of Indonesia to support this work through the Master Thesis Research with contract number 72/SPK/D.D4/PPK.01.APTV/III/2024.

6. REFERENCES:

- [1] A. Shuaibu Hassan, I. Adabara, A. Ronald, K. Muteba, "Design and Implementation of an Automatic Power Supply from Four Different Source Using Microcontroller", International Journal of Electrical and Electronic Science, Vol. 4, No. 5, 2017, pp. 40-46.
- [2] D. Toumi et al. "Optimal design and analysis of DC-DC converter with maximum power controller for stand-alone PV system", Energy Reports, Vol. 7, 2021, pp. 4951-4960.

- [3] J. Ingilala, I. Vairavasundaram, "Investigation of high gain DC/DC converter for solar PV applications", *e-Prime - Advances in Electrical Engineering, Electronics and Energy*, Vol. 5, 2023.
- [4] N. Kamarudin, A. A. A. Samat, M. F. N. Tajudin, M. K. Osman, S. Omar, I. H. Hamyah, "Design of Buck Converter Based on Maximum Power Point Tracking for Photovoltaic Applications", *Journal of Advanced Research in Applied Sciences and Engineering Technology*, Vol. 39, No. 2, 2024.
- [5] H. Oufettoul, N. Lamdi hine, S. Motahhir, N. Lamerini, I. A. Abdelmoula, G. Aniba, "Comparative Performance Analysis of PV Module Positions in a Solar PV Array Under Partial Shading Conditions", *IEEE Access*, Vol. 11, 2023, pp. 12176-12194.
- [6] A. D. Martin, J. M. Cano, J. Medina-García, J. A. Gómez-Galán, A. Hermoso, J. R. Vazquez, "Artificial vision wireless PV system to efficiently track the MPP under partial shading", *International Journal of Electrical Power and Energy Systems*, Vol. 151, 2023.
- [7] D. Mazumdar, P. K. Biswas, C. Sain, F. Ahmad, L. Al-Fagih, "A comprehensive analysis of the optimal GWO based FOPID MPPT controller for grid-tied photovoltaics system under atmospheric uncertainty", *Energy Reports*, Vol. 12, 2024, pp. 1921-1935.
- [8] Y. Fetene, E. Ayenew, S. Feleke, "Full state observer-based pole placement controller for pulse width modulation switched mode voltage-controlled buck Converter", *Heliyon*, Vol. 10, No. 9, 2024.
- [9] F. Zishan, A. Barmakh, O. D. Montoya-Giraldo, "A non-isolated synchronous buck DC-DC converter, ZVS topology under CCM and DCM conditions", *Results in Engineering*, Vol. 23, 2024.
- [10] A. Baraeian, M. Kassas, M. S. Alam, M. A. Abido, "Physics-informed NN-based adaptive backstepping terminal sliding mode control of buck converter for PEM electrolyzer", *Heliyon*, Vol. 10, No. 7, 2024.
- [11] M. Harith, H. M. Salih, W. M. Utomo, "Design Buck Converter for DC Motor of Transporter Model with IoT System", *Evolution in Electrical and Electronic Engineering*, Vol. 4, No. 2, 2023, pp. 582-591.
- [12] L. Fang, E. Quisbert-Trujillo, P. Lefranc, M. Rio, "Leading LCA result interpretation towards efficient ecodesign strategies for Power Electronics: The case of DC-DC buck converters", *Procedia CIRP*, Vol. 122, 2024, pp. 731-736.
- [13] H. Zomorodi, E. Nazari, "Design and Simulation of Synchronous Buck Converter in Comparison with Regular Buck Converter", *International Journal of Robotics and Control Systems*, Vol. 2, No. 1, 2022, pp. 79-86.
- [14] S. Gul, S. M. Malik, Y. Sun, F. Alsaif, "An Artificial Neural Network Based MPPT Control of Modified Flyback Converter for PV Systems in Active Buildings", *Energy Reports*, Vol. 12, 2024, pp. 2865-2872.
- [15] S. Sarwar, M. Y. Javed, A. B. Asghar, W. Iqbal, K. Ejsmont, M. H. Jaffery, "A Coronavirus Optimization (CVO) algorithm to harvest maximum power from PV systems under partial and complex partial shading conditions", *Energy Reports*, Vol. 11, 2024, pp. 1693-1710.
- [16] H. Abidi, L. Sidhom, I. Chihi, "Systematic Literature Review and Benchmarking for Photovoltaic MPPT Techniques", *Energies*, Vol. 16, No. 8, 2023, p. 3509.
- [17] S. Senthilkumar et al. "A Review on MPPT Algorithms for Solar PV Systems", *International Journal of Research -GRANTHAALAYAH*, Vol. 11, No. 3, 2023.
- [18] M. L. Katche, A. B. Makokha, S. O. Zachary, M. S. Adaramola, "A Comprehensive Review of Maximum Power Point Tracking (MPPT) Techniques Used in Solar PV Systems", *Energies*, Vol. 16, No. 5, 2023, p. 2206.
- [19] L. F. Giraldo, J. F. Gaviria, M. I. Torres, C. Alonso, M. Bressan, "Deep reinforcement learning using deep-Q-network for Global Maximum Power Point tracking: Design and experiments in real photovoltaic systems", *Heliyon*, Vol. 10, No. 21, 2024.
- [20] H. Karmouni, M. Chouiekh, S. Motahhir, H. Qjidaa, M. Ouazzani Jamil, M. Sayyouri, "A fast and accurate sine-cosine MPPT algorithm under partial shading with implementation using Arduino board", *Cleaner Engineering and Technology*, Vol. 9, 2022.

- [21] M. Gursoy, G. Zhuo, A. G. Lozowski, X. Wang, "Photovoltaic Energy Conversion Systems with Sliding Mode Control", *Energies*, Vol. 14, No. 19, 2021, p. 6071.
- [22] A. F. Sagonda, K. A. Folly, "A comparative study between deterministic and two meta-heuristic algorithms for solar PV MPPT control under partial shading conditions", *Systems and Soft Computing*, Vol. 4, 2022.
- [23] A. B. Djilali, A. Yahdou, H. Benbouhenni, A. Alhejji, D. Zellouma, E. Bounadja, "Enhanced perturb and observe control for addressing power loss under rapid load changes using a buck-boost converter", *Energy Reports*, Vol. 12, 2024, pp. 1503-1516.
- [24] R. I. Putri, F. Ronilaya, I. N. Syamsiana, L. Jasa, "Improvement efficiency of photovoltaic system using modified perturb and observe", *Proceedings of the International Conference on Smart-Green Technology in Electrical and Information Systems*, Sanur, Bali, Indonesia, 28-30 October 2021, pp. 24-28.
- [25] M. S. Endiz, "Design and implementation of microcontroller-based solar charge controller using modified incremental conductance MPPT algorithm", *Journal of Radiation Research and Applied Sciences*, Vol. 17, No. 2, 2024, p. 100938.
- [26] S. Kumaraguruparan, K. Elango, "Optimal control strategies for high-efficiency non-isolated DC-DC buck converters in IoT applications: A comparative study", *Heliyon*, Vol. 10, No. 18, 2024.
- [27] G. Song, X. Liu, J. Tian, G. Xiao, T. Zhao, P. Wang, "Global Maximum Power Point Tracking for PV Conversion Systems under Partial Shadings: NNIDA Based Approach", *IEEE Transactions on Power Delivery*, Vol. 38, No. 5, 2023, pp. 3179-3191.

Distributed Approach to detect DDOS attack based on Elephant Herding Optimization and Pipeline Artificial Neural Network

Original Scientific Paper

Yasamin Hamza Alagrash*

Mustansiriyah University
Faculty of Science, Department of Computer Science
Baghdad, Iraq
yhamza@uomustansiriyah.edu.iq

*Corresponding author

Abstract – Cybersecurity experts widely acknowledge that a Distributed Denial of Service (DDoS) assault poses a grave threat, capable of inflicting substantial financial losses and tarnishing the reputation of enterprises. Conventional detection methods are insufficient for identifying DDoS attacks. Simultaneously, with their vast potential, machine learning solutions play a vital role in this field. This paper presents a distributed approach for identifying distributed denial-of-service threats using the pipeline artificial neural network method, supported by elephant herding optimization for feature selection and extraction. The proposed artificial neural network pipeline-based model for detecting DDoS attacks comprises several key stages: collecting the dataset, preparing the data, implementing a balanced data strategy, selecting relevant features using the swarm optimization method Elephant Herding Optimization (EHO), training the model, testing its performance, and evaluating its effectiveness. Experimental results demonstrate that the proposed approach effectively enhances DDoS detection accuracy while reducing false positives, making it a promising solution for network security. This model demonstrated a remarkably high ability to detect DDoS attacks with a 99% accuracy. Thorough investigations demonstrate that the model is highly skilled in implementing security measures and reducing the risks connected with emerging security threats. The effectiveness of our proposed solution, leveraging a pipeline method in Artificial Neural Network (ANN), is crucial to building a reliable model, which is evident in its ability to deliver effective results in low complexity. The proposed method achieves 99.99% accuracy, 99.80% precision, and a False Positive Rate (FPR) of 0.002%, outperforming recent models. These results demonstrate the model's superior accuracy and robustness in identifying complex attack patterns while minimizing false positives.

Keywords: DDoS attack and detection, Auto machine learning, Pipeline ANN, Elephant Herding Optimization (EHO), Artificial Neural Network, AutoML

Received: February 8, 2025; Received in revised form: June 29, 2025; Accepted: June 30, 2025

1. INTRODUCTION

Distributed Denial of Service (DDoS) attacks are a significant concern in cybersecurity, as extensively reported in the areas of network security, data breaches, and research on malicious activity. The attacks that cause the most denial-of-service disruption send a large number of requests to overload the service. DDoS attacks can lead to a complete or partial service disruption, preventing legitimate users from accessing online services. If attacks do not cause the service to crash completely, they can lead to extremely slow performance and poor user experience[1]. DDoS attacks fall into two main categories: network-level attacks, which overwhelm systems with high-volume data packets

(e.g., UDP or SYN floods), and application-level attacks, which exploit weaknesses in applications and online services, such as HTTP flood attacks [2].

The rapidly evolving DDoS area has become so intricate that it is challenging to maintain a clear perspective. This complexity hampers the comprehension of the DDoS phenomenon. Many approaches indicate that the issue is extensive and challenging to investigate and resolve. Existing defense systems employ various methods to address the issue. However, evaluating and comparing their efficacy and cost is daunting, underscoring the critical need for new, innovative approaches [3]. Conventional DDoS attack detection techniques, although efficient with gradual data incre-

ments, are inadequate for effectively analyzing large volumes of high-speed data to detect signs of infiltration. This inadequacy is particularly evident in the face of the evolving nature of DDoS attacks. A DDoS is a crucial cyberattack that aims to disrupt the normal operations of specified servers or networks. This underscores the need for a more robust and efficient approach, which our research aims to provide [4].

Machine learning (ML) is continuously evolving through practice and the application of knowledge [4] [5]. It is regarded as a constituent of artificial intelligence. Depending on the available information, various learning methods exist, such as supervised, semi-supervised, and unsupervised learning [6]. Pipelines and Automated Machine Learning (AutoML) aim to generate algorithmic solutions for machine learning tasks automatically, referred to as machine learning pipelines, that are customized for a specific data set [7]. The ML application in DDoS detection presents challenges in accurately recognizing and preventing attacks while maintaining system efficiency. Several studies have employed classification algorithms to identify and prevent DDoS attacks. DDoS attacks exploit network vulnerabilities to flood a service with excessive requests. Identifying and stopping DDoS attacks in real time can be challenging due to their complex nature and significant consequences [8]. Detection of attacks by anomalies relies on observing differences from standard model usage patterns. These computations depend on easily accessible system parameters, including average CPU utilization, network session rates, user activity frequency, and the type of application being accessed [9]. A variation from a system profile or anomaly could be a sign of a potential intrusion. Yet, the present DDoS attack detection solutions have limitations, such as high detection expenses and an inability to manage substantial network traffic directed toward the server. The packets are examined with classification methods to differentiate DDoS broadcasts from everyday communications [10].

The efficacy of recognition and detection remains challenging, as evidenced by DDoS attacks and the application of machine learning in security evaluation. Several studies have utilized classification algorithms to detect and prevent DDoS attacks. Exploiting network vulnerabilities and sending service requests to the network makes DDoS attacks straightforward. The research created a pipeline machine learning model to identify DDoS attacks in real-time systems. Existing DDoS detection methods, such as rule-based and statistical approaches, often fail under high-traffic loads and adaptive attack strategies. Rule-based systems struggle to detect new attack variations, while statistical models suffer from high false positive rates when traffic patterns fluctuate. The proposed method addresses these issues by utilizing adaptive feature selection, known as EHO, and a deep learning-based classifier, which enables it to adjust to evolving attack patterns while maintaining high accuracy dynamically. The main contributions of this work are as follows:

Developed the EHO for the feature selection phase; most importantly, this work introduced a machine-learning method as a KNN algorithm for evaluating the selected features (fitness function role).

Developed a pipeline ANN model to automate the detection of DDoS attacks using a distributed architecture paradigm. The scalable model is a structured sequence of interrelated data processing and modeling activities created to automate, standardize, and optimize the process of constructing, training, assessing, and implementing machine learning models.

The new detection technique stands out from previous approaches due to its approach of dataset feeding, which combines batch system methodology with streaming.

The performance is improved compared to existing models, as evidenced by several key performance metrics.

The paper is structured as follows: Section 2 describes recent studies on machine learning methods for detecting network attacks and the existing work in this area; Section 3 presents the materials and methods; Section 4 outlines the proposed model; Section 5 describes the dataset splitting and cross-validation. Section 6 displays the proposed pipeline ANN method. Section 7 describes how to evaluate the proposed model, and Section 8 discusses its comparison with previous models. The conclusion is presented in Section 9.

2. RELATED WORK

Several studies have been developed to identify DDoS attacks through machine learning, employing a similar research methodology to ensure consistency in the intensive effort.

Hnamte *et al.* [11] present a new approach to DDoS attack identification using a deep neural network (DNN) model based on deep learning (DL) concepts. This method is designed to be scalable and adaptive for monitoring network traffic and identifying patterns related to DDoS attacks. This paper evaluated the DNN model performance using different datasets, including SDN, CICIDS2018, and Kaggle DDoS. Results show that, in terms of detection accuracy, their proposed DNN-based methodology was 99.98% for the SDN dataset, 100% for the CICIDS2018 dataset, and 99.99% for the Kaggle DDoS dataset. Disadvantages: The paper gives more advantages of the DNN-based methodology than is necessary. It also notes the challenge of implementing DNNs practically in an SDN setting without discussing the details of the technique. Mustapha *et al.* [12] proposed a hybrid method combining Machine Learning (ML) and Deep Learning (DL) algorithms. They utilized Generative Adversarial Networks (GAN) to generate realistic data and employed a Long Short-Term Memory (LSTM) model for DDoS detection. The system achieved a detection accuracy of between 91.75% and 100%. However, the solution is characterized by complexity and overhead.

Anley *et al.* [13] developed a methodology that utilizes deep learning for DDoS detection, employing adaptive architectures within a transfer-learning framework. It discusses transferring information between disparate datasets to enhance classification accuracy in adaptive architectures for DDoS detection. The methodology utilizes tailored CNN architectures with varied layer configurations and pre-trained models (VGG16, VGG19, and ResNet50) while adaptively optimizing hyperparameters. The model was evaluated using four publicly accessible datasets: KDDCup'99, UNSW-NB15, CSE-CIC-IDS2018, and CIC-DDoS2019. The suggested adaptive transfer learning technique proficiently distinguishes between benign and malignant activities and specific attack classifications. Custom CNN models demonstrated exceptional accuracy in distinguishing between benign and DDoS attack traffic, with Conv4 achieving 99.90%, Conv8 attaining 99.94%, and Conv18 reaching 99.88% on the CIC-DDoS2019 dataset. The Conv18 model, adapted from CIC-DDoS2019 to the CSE-CIC-IDS2018 dataset, was archived. The approach attains better results relative to single-domain training. Adaptive designs and hyperparameter optimization enhance the robustness and efficiency of DDoS attack detection. This approach has certain drawbacks, including the use of multiple CNN architectures, transfer learning, and hyperparameter optimization, which likely increase the system's complexity. This may result in elevated computing expenses and necessitate substantial resources for execution, particularly in a real-time context.

Ouhssini *et al.* [14] introduced a Deep Defend approach, a system for the real-time detection and prevention of DDoS attacks in cloud environments. It utilizes deep learning methodologies, notably CNN-LSTM-Transformer architectures, to forecast traffic entropy and identify potential assaults. The framework employs a genetic approach for optimal feature selection to improve the effectiveness of the CNN-DT model in differentiating between regular and attack traffic. This methodology utilizes entropy-based forecasting to predict potential DDoS attack periods, thereby reducing the computational burden associated with preprocessing and classification. The approach was evaluated using the CIDDs-001 network traffic dataset. Results: The proposed system exhibits exceptional accuracy in entropy predictions. Additionally, it facilitates the swift and precise identification of DDoS attacks.

The primary drawback is that the study indicates that the CIDDs-001 dataset contains constraints, such as a restricted number of characteristics, class imbalance (notably for attacks aside from DDoS), and a significant volume of duplicated data, which may cause biases and compromise the accuracy of the conclusions.

Beshah *et al.* [15] presented a different accuracy update weighted Probability Averaging Ensemble (AUWPAE) framework introduced for DDoS attack detection utilizing real-time data streaming. The proposed system utilizes the dynamic characteristics of incoming

streaming data to construct a model that identifies idea drifts. The AUWPAE methodology assigns dynamic weights based on their real-time performance to base learners.

Solution Evaluated On: IoTID20 and CICIoT2023 datasets comprising benign and DDoS traffic data. The suggested adaptive online DDoS attack detection framework achieves detection accuracies of 99.54% and 99.33% for the relevant datasets. However, the study doesn't show a real scenario for how the proposed model works in real-time data streaming.

Ashraf *et al.* [16] employed a DDoS detection model that utilized machine learning algorithms, including Random Forest, SVM (Support Vector Machine), Naïve Bayes, KNN (K-Nearest Neighbors), XGBoost, and AdaBoost, on the CICDDoS2019 dataset. The study enhances dimensionality reduction and feature selection methods for efficient DDoS detection, identifying essential elements within. The machine learning methods AdaBoost and XGBoost exhibited outstanding performance, with 100% accuracy in DDoS attack detection. Alternative algorithms, such as KNN and Random Forest, demonstrated higher accuracy, while SVM and Naïve Bayes showed comparatively lower accuracy. Naïve Bayes exhibited the shortest training duration but yielded the lowest F1-score, indicating constraints in DDoS attack identification due to a high incidence of false positives. The research focuses on the Port map segment of the CICDDoS2019 dataset, which may limit the generalizability of the results.

Suarez *et al.* [17] This study evaluates six different machine learning models: Random Forest (RF), Decision Tree (DT), AdaBoost (ADA), XGBoost (XGB), Multilayer Perceptron (MLP), and Deep Neural Network (DNN). This paper presents a preprocessing and feature selection approach using the CICDDoS2019 dataset. The authors examined features using Principal Component Analysis (PCA) and Pearson correlation; subsequently, Tree of Parzen Estimators (TPE) was employed for hyperparameter optimization. This comprehensive methodology, which encompasses assessing various machine learning models with sophisticated preprocessing and feature selection approaches, enabled the authors to attain elevated accuracy in DDoS attack detection while minimizing the number of features. The Random Forest (RF) classifier demonstrated superior performance, attaining an accuracy of 99.97%, an F1 score of 99.98%, and an AUC score of 99.96%.

Other classifiers were considerably accurate; nevertheless, RF surpassed them. The research identifies a shortcoming in the model's capacity to adapt to rapidly developing DDoS attacks. Examining response time during real-time DDoS attacks is vital for future study consideration.

The solution proposed by Elsadiq *et al.* [18] presents a streamlined machine learning methodology that employs the XGBoost model to detect DoS attacks in wire-

less sensor networks (WSNs). It utilizes the latest WSN-DS dataset, which is specifically designed for evaluating DoS attacks in WSNs. The methodology emphasizes high precision, effective feature selection, thorough assessment metrics, and minimized processing duration, rendering it appropriate for real-time detection in WSN settings.

The approach was evaluated using the WSN-DS dataset. This dataset comprises both regular and abnormal traffic, featuring four types of DoS attacks: black hole, gray hole, TDMA, and flood.

The proposed XGBoost model attained exceptional performance, with a maximum accuracy of 99.73%.

Compared to other examined classifiers, XGBoost exhibited a 68% reduction in processing time. The results underscore the efficacy of ensemble approaches such as XGBoost. The paper identifies constraints, including issues associated with real-time implementation, scalability concerns, and dataset limitations that may inadequately reflect the enormous diversity of contemporary DoS attacks.

Silivery *et al.* [19] presents a deep learning based multi-classification system to detect DoS and DDoS attacks, which consists of DCGAN to generate synthetic samples, ResNet-50 to extract deep features, and a modified version of AlexNet as a classifier that is trained with the help of the Atom Search Optimization (ASO) algorithm. This multi-component pipeline achieved an accuracy of 99.37% and 99.33% on the UNSW-NB15 and CICIDS2019 datasets, respectively. The problem of class imbalance was adequately addressed with the help of GANs, and feature representation and classification accuracy were enhanced using ResNet and AlexNet.

The model, however, presents enormous architectural complexity that can prove challenging to execute in real-time or on the edge due to the computational and training overheads.

The approach suggested by Naiem *et al.* [20] is an iteration feature selection-based and Cloud-specific approach to DDoS mitigation. It utilizes the Pearson Correlation Coefficient (PCC) and Random Forest Feature Importance (RFFI) to reduce the feature space, which is then fed to machine learning classifiers comprising Support Vector Machines (SVM) and Decision Trees. Their model achieved 99.27% accuracy and 97.6% precision on cloud-specific datasets, with minimal feature dependency and reduced latency in processing as its key priorities.

However, this framework has not been demonstrated to apply in real-time and to dynamic traffic, which is crucial in real-world cloud-based deployment situations.

Akinwale *et al.* [21] propose a model of HTTP regeneration (HReg) to counter attacks on mobile HTTP servers. Based on the OMNeT++ simulation platform, the system dynamically detects and regenerates corrupted HTTP sessions to maintain service availability. The measured performance is 73% throughput, 68.8% delivery ratio, and 69.4% goodput under DDoS attack conditions.

Although the approach of regeneration is new and promising in mobile and wireless settings, it has yet to be tested in a real deployment scenario or against deep learning-based detection baselines.

Hussein proposes a deep learning model, which combines a denoising autoencoder (DAE) and a 1D convolutional neural network (CNN) to detect DDoS attacks on the NSL-KDD dataset. The noisy features cleaned up by the DAE present clearer signals before classification. The model achieves a high level of accuracy of 97.7%, a recall of 98.1%, and an F1-score of 97.8% [22].

Nonetheless, the commonly used NSL-KDD dataset lacks freshness and comprehensiveness in reflecting contemporary and emerging DDoS threats. The study also does not consider the system latency and streaming deployment.

The analyzed publications on machine learning and deep learning-based DDoS detection demonstrate high detection accuracy, typically exceeding 99 percent, particularly with the latest methods, including GANs, ensemble models, CNN-LSTM-Transformer architectures, and transfer learning. Most of them, however, have drawbacks such as overly complex models, heavy computation, the use of outdated or unbalanced datasets, and have not been tested in real-time or resource-constrained settings, including those found in IoT, cloud, and wireless networks. Although some of the approaches are either streaming or cloud-specific, the majority lack evidence of real-world deployment. The gaps mentioned underscore the need for lightweight, efficient, and adaptive DDoS detection systems that can be effectively applied in the real world. This research aims to address this need by optimizing the preferred streamlined pipeline ANN model for minimal resource consumption and real-time detection capability.

This study develops a new approach to detecting DDoS attacks based on auto ANN, specifically pipeline ANN, which can deal with real-time schemes and resource-constrained systems.

3. MATERIALS AND METHODS

The materials and methods required in the proposed method are represented as follows:

3.1. DISTRIBUTED DENIAL OF SERVICE (DDOS)

Various compromised devices are utilized in a DDoS attack to target and disrupt a service. DDoS attacks are carried out through botnets. In a DDoS attack, users usually send a server authentication request to establish a connection. The server responds with the outcome of the authentication process. Once the asking user grants this authorization, a connection is established, and access to the server is provided [23]. Also, the attacker floods the server with many authentication requests. Since the requests have fake return addresses, the server needs assistance in identifying a

user to provide authentication approval. The session ends automatically after a set duration during this authentication process. The server typically prolongs the session for over a minute before terminating it. The attacker's continuous requests overwhelm the server, resulting in numerous open connections and a denial of service [24][25].

It frequently occurs when many systems overwhelm a victim's bandwidth or capacity. Such an attack occurs because numerous hacked systems (for instance, a botnet) bombard the targeted system and produce a lot of network traffic, which is performing a task [26]. Hostile botnets infiltrate computers with malicious scripts and programs. Once the botnet gains control of the system, it alerts the master computer. An attacker can take control of the system and send commands to try a DoS attack using this master machine [27].

The proposed approach operates in a distributed environment by deploying a detection method across multiple network nodes. Each node independently analyzes incoming traffic and shares anomaly reports with a central decision system. This distributed processing improves scalability and ensures real-time DDoS detection by reducing the computational burden on a single detection point. Unlike centralized methods, this framework enhances resilience against targeted attacks on a single detection server. Fig. 1 illustrates the threat model of a DDoS attack.

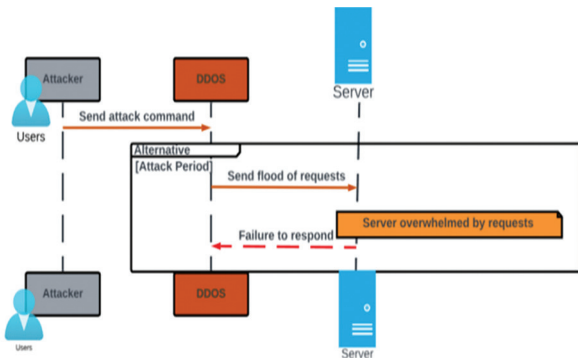


Fig. 1. Threat Model of DDoS Attack

3.2. ELEPHANT HERDING OPTIMIZATION (EHO)

The concept of the behavior of herding elephants can be summarized as follows:

The elephant population is partitioned into a predetermined number of clans, each customized for female elephants. Every elephant in a clan is under the leadership of the master female, known as the matriarch. This behavior aims to find the best solution within a smaller search space, known as a local search. Furthermore, male elephants depart from their clans after puberty and establish independent lives. This behavior is employed to guarantee a global search [28].

The solutions of every male elephant are regarded as wrong solutions. Conversely, all female elephants'

solutions are considered good, and the matriarch possesses the best solution within each clan. The EHO algorithm can be characterized based on elephant herding behavior [29].

The population of elephants is partitioned into j clans. The matriarch's influence determines the new place for each elephant in the ci . The j th elephant within the ci clan can be computed by Equation (1):

$$e_{new, ci, j} = e_{cij} + \alpha \times (e_{best, ci} - e_{ci, j}) \times r \quad (1)$$

The $e_{new, ci, j}$ represents the updated placement while e_{cij} Represents the previous placement for Elephant j in the clan ci . $e_{best, ci}$ refers to Matriarch CI , who is considered the best elephant. The scaling factor " α " belongs to the interval $[0,1]$, while the variable " r " belongs to the interval $[0,1]$, which is generated for each individual in each iteration [30]. The best elephant for each clan is computed by Equation (2):

$$e_{new, ci, j} = \beta \times e_{center, ci} \quad (2)$$

The factor β , between 0 and 1, defines the impact $e_{center, ci}$. In the new individual $e_{new, ci, j}$. The $e_{center, ci}$ indicates the ci clan's central individual (matriarch). It can be computed using Equation (3) for the d th dimension.

$$e_{center, ci, d} = \frac{1}{n_{ci}} \times \sum_{j=1}^{n_{ci}} e_{ci, j, d} \quad (3)$$

Where $1 \leq d \leq D$, and n_{ci} indicates the number of elephants in clan ci . $e_{ci, j, d}$ it is the d th dimension of the individual. $e_{ci, j, d}$ while the center (matriarch) of clan ci ($e_{center, ci, d}$) can be modified using Equation (3). When addressing optimization problems, male elephants leaving their families can be represented as separating operators. The individual with the lowest fitness in every iteration executes the separation operator, as demonstrated in Equation (4).

$$e_{worst, d} = e_{min} + (e_{max} - e_{min} + 1) \times rand \quad (4)$$

In the search space, the lower and upper limits are denoted by e_{min} and e_{max} , respectively. The variable " $rand$ " represents a randomly generated number from 0 to 1 [31].

EHO differs from other optimization algorithms because it does not utilize the prior individuals in the subsequent updating phase. EHO is an algorithm inspired by a swarm that handles global optimization tasks involving clan updates and searching operations. EHO does not focus on relaxation techniques because it is more noise resistant. EHO works very well in constrained and optimized environments. The key characteristics of EHO include a rapid convergence rate, the lowest mistakes in determining localizations, and efficient execution time. The algorithm can address ML problems that are not convex directly [32].

4. PROPOSED MODEL

This paper aims to develop an auto ANN system to identify DDoS attacks. The proposed system consists of four stages: loading the dataset, preprocessing, and

feature selection based on the EHO method. Then, a comprehensive pipeline process is established by building and evaluating an ANN model until the best model is identified and exported. Fig. 2 illustrates the generic framework of the detection model.

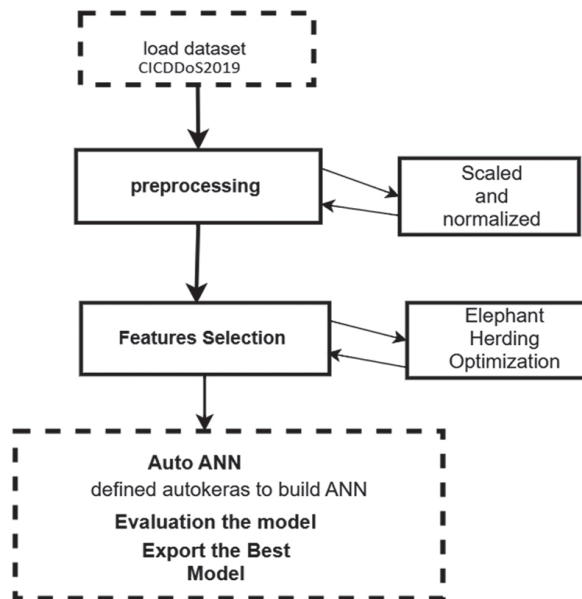


Fig. 2. Proposed Framework to detect DDOS attack

4.1. DATASET ASSEMBLING

The Canadian Institute for Cybersecurity generated the CICDDoS2019 dataset in an authentic network setting, including current genuine data. The collection includes a range of modern DDoS assaults targeting SYN, LDAP, Port Map, UDP, NetBIOS, UDP-Lag, SNMP, MSSQL, DNS, and NTP [33]. Analysis of the CICDDoS2019 dataset reveals that out of 1,048,575 network flow records, more than 58% were classified as attacks and around 42% as legitimate network traffic flows [34]. This study gathers and compiles DDOS attack data, a subset of the CICDDoS2018 dataset, as the proposed model focuses on DDOS attacks.

4.2 DATA PREPROCESSING

The dataset fed to a proposed model was created using preprocessing techniques with batch data preprocessing restriction. It was cleaned by removing null values, and normalization procedures were applied to scale and balance it. The essential characteristics relevant to the DDOS attack flows were collected from the datasets to enable the effective and efficient application of the proposed models. The focus is on differentiating between assaults and standard traffic patterns, rather than individual packets. The preprocessing phase procedure is explained in Figure 3. To address class imbalance, Synthetic Minority Over-sampling Technique (SMOTE) is applied to generate additional attack samples, ensuring a balanced dataset. This helps in improving the classifier's performance by preventing bias towards the majority class.

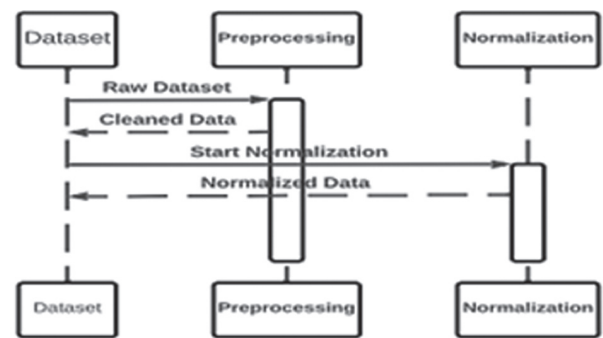


Fig. 3. Dataset pipeline preprocessing

4.3. FEATURE SELECTION

First, the features used in DDOS attacks are normalized, putting all the feature values on the same scale, thus helping the model learn more efficiently and train faster. Each column (feature) is normalized except the last column by converting each value into a float number; this can be done by subtracting the mean from each element in the column and then dividing the result by the column standard deviation. EHO is chosen for feature selection due to its ability to reduce dimensionality while maintaining classification performance. The algorithm iteratively selects the most relevant features based on their contribution to detection accuracy. After optimization, the retained features included packet size, flow duration, source port, destination port, and protocol type, which were identified as key indicators of DDOS attack patterns.

4.3.1. Population Initialized

A population is produced randomly within a search space. This population comprises multiple elephants (solutions), both females and males. Each solution has multiple features, expressed as floating-point values assigned during the normalization process.

4.3.2. Selecting Features Based on Threshold

Based on the predetermined threshold, the float numbers for features in each solution are converted into binary numbers. "Zero" indicates that this feature was not selected. "One" means choosing the corresponding column from the original dataset.

4.3.3. Find Accuracy

Initially, the newly converted dataset is split into training and test sets. At each iteration, the KNN model is trained on the training set (comprising 80% of the total dataset size), and its performance is evaluated based on the test set (comprising 20% of the total dataset size). Then, the model's accuracy is measured by comparing the model predictions to the actual class values, and the accuracy for each solution represents the fitness value for that solution. Table 1 displays the fitness value for each solution in a single iteration.

Table 1. The fitness value (accuracy using KNN) is in one iteration

#	No. of features	Fitness Value (accuracy)
1	43	76.381%
2	40	77.891%
3	36	77.548%
4	43	79.234%
5	37	78.543%
6	43	84.089%
7	43	83.749%
8	41	77.770%
9	41	76.476%
10	40	80.122%

4.3.4. Sorting the Solutions

After computing the fitness value (accuracy) for each solution, these solutions and their corresponding positions are sorted in descending order (from highest to lowest) based on their fitness values to define all the solutions, from the best-performing solution to the worst-performing solution, and their respective positions. Table 2 presents the fitness values in descending order.

Table 2. Order of the fitness function

New index	Previous Index	No. of features	Fitness
1	6	43	84.089%
2	7	43	83.749%
3	10	40	80.122%
4	4	43	79.234%
5	5	37	78.543%
6	2	40	77.891%
7	8	41	77.770%
8	3	36	77.548%
9	9	41	76.476%
10	1	43	76.381%

4.3.5. Clans Creation

Female elephants and their calves live in a group of many clans; each has one Matraich, also called the clan leader, who represents the best solution in that clan. The number of female solutions per clan is determined during each iteration by dividing the total population (ten solutions) by the predefined three clans. The remaining solutions are categorized as male solutions, representing male elephants that separate from their family upon reaching maturity and live independently. The female solutions indicate the randomly selected reasonable solutions, while the male ones indicate the worst ones.

4.3.6. Update Clan Operator

First, the centroid (best solution) of each clan is computed by taking the mean of all solutions within the clan, which is calculated separately for each feature. Second, each clan's best solution position is determined based on the highest fitness value. Third, for each solution in each clan, if the current solution doesn't equal the best, the position of the current so-

lution is updated using Equation (1). More specifically, Equations (2) and (3) are used to update the position of the best solution that is currently accessible. The clan solutions aim to update the current solution randomly and optimize their positions sequentially in each iteration, thereby increasing the probability of reaching the optimal solution.

4.3.7. Update Separate Operator

The position of the worst solution is optimized by adding to the male solutions, allowing them to explore new regions in the search space, which helps them avoid local solutions. The Updating aims to move the male solutions towards a better position in the search space.

4.3.8. Best Features

When the stopping condition is met, the best solution with the highest fitness value is selected in each iteration. Then, these best solutions from all iterations are ranked in descending order to choose the optimal solution. This solution comprises many features, denoted as 0s and 1s, and only features with a value of 1 are considered optimal. Table 3 shows the selected features.

Table 3. Selected features using EHO

fet1	fet2	fet3	feat4	fet5	fet6
'Flow Duration'	'Total Fwd Packets'	'Fwd Packet Length Mean'	'Fwd Packet Length Std'	'Bwd Packet Length Max'	'Bwd Packet Length Min'
fet7	fet8	fet9	fet10	fet11	fet12
'Bwd Packet Length Mean'	'Flow IATStd'	'Flow IAT Max'	'Flow IAT Min'	'Fwd IAT Total'	'Fwd IAT Mean'
fet13	fet14	fet15	fet16	fet17	fet18
'Fwd IAT Std'	'Fwd IAT Max'	'Fwd IAT Min'	'Bwd IAT Total'	'Bwd IAT Mean'	'Bwd IAT Std'

5. SPLIT DATASET AND CROSS VALIDATION

The data splitting process, represented by confirming examples of 1500 regular and anomalous DDoS attacks, is categorized into significant subgroups. The initial dataset is the training subset, comprising data selected randomly from the training dataset. Our solution utilizes the K-fold methodology [35]. To analyze the influence of various k values on the model performance estimation and compare it to the optimal test scenario. This can help determine the appropriate value of K. To determine the algorithm that is substantially associated, compare the distribution of the data with that of the optimal test scenario.

With the result distribution from an assessment of the same algorithms under ideal test conditions. The chosen configuration is a reliable approximation for the ideal test scenario, provided the outcomes are correlated at 30% for testing and 70% for training. To evaluate model performance, this work employed 10-fold cross-validation. Figure 4 demonstrates k-fold accuracy.

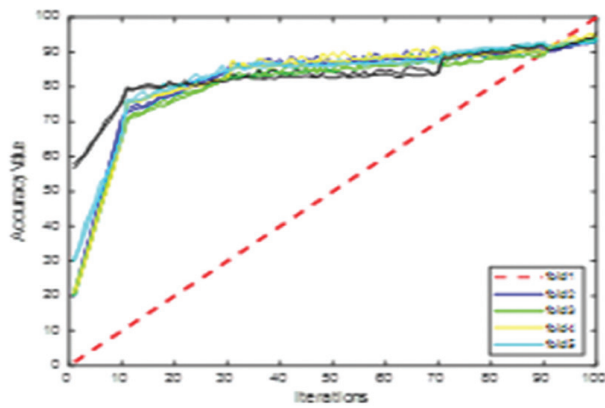


Fig. 4. K-fold Accuracy result

6. PIPELINE ARTIFICIAL NEURAL NETWORK

A machine learning pipeline is an extensive process that encompasses data collection, preprocessing, feature engineering, model training, hyperparameter op-

timization, evaluation, and deployment. Every phase is essential for constructing a proficient and effective machine learning model, and automation technologies such as Auto ANN help optimize these procedures. Fig. 5 illustrates the construction of a pipeline ANN.

Building an auto ANN, an automated design search is activated using the AutoKeras library to identify the optimal neural network design efficiently.

Hyperparameter optimization: It additionally automates the refinement of hyperparameters. The ANN is optimized using grid search tuning to select the best hyperparameters. The final model has a learning rate of 0.001, comprising 3 hidden layers with 128, 64, and 32 neurons, respectively. The ReLU activation function is used, and the Adam optimizer is employed with a batch size of 32 and 100 epochs. This configuration is chosen based on experimental results, which maximize the F1-score and minimize false positives. Table 4 shows the ANN design structure.

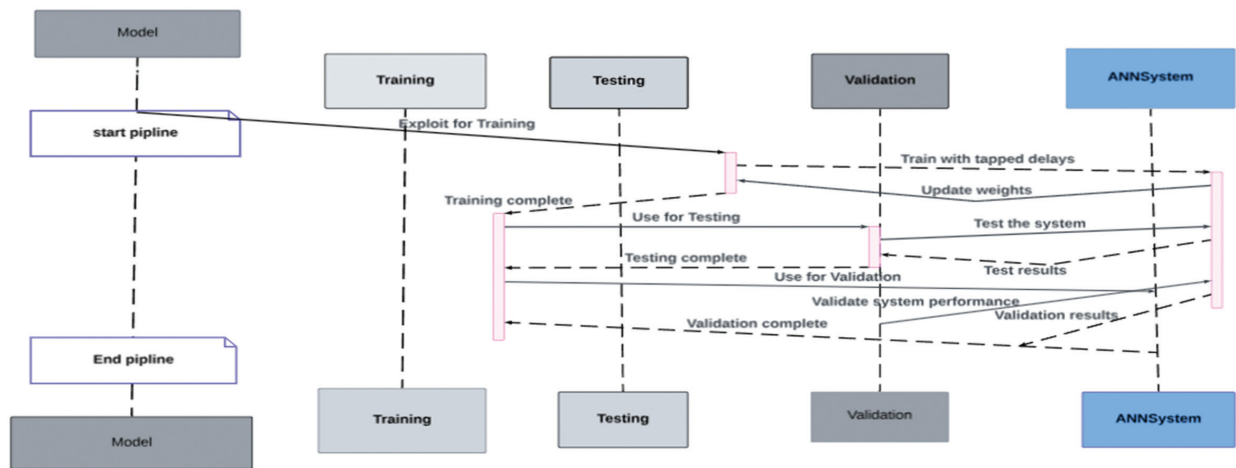


Fig. 5. Pipeline ANN

Table 4. ANN parameters

ANN Parameters	Value
Neurons	(10,100)
Activation function	"ReLU"
optimizer	"ADAM"
Learning rate	0.01
Batch size	200
Epochs	100
Dropout rate	0.2
Normalization	(0,1)

7. EXPERIMENTS AND RESULTS

This section focuses on evaluating the performance of the proposed model. The experiments were conducted on a system equipped with an Intel i7-12700K processor, 32 GB of RAM, and an NVIDIA RTX 3080 GPU. The software environment used is Python 3.9, which utilizes TensorFlow 2.10 and Scikit-learn 1.1 modules. With the ray[tune] python library, a distributed hyperparameter across multiple nodes.

The training time for 100 epochs is 45 minutes, and the average inference time is 3.2 milliseconds per sample, making it suitable for real-time deployment. The outputs are evaluated by comparing the confusion matrix and prediction time to analyze the performance differences between the classifiers.

7.1. EXPERIMENT MODEL

The testing bed is in a distributed structure for DDoS attack detection, as shown in Figure 6. To evaluate the system's performance, this test deploys a primary virtual machine (VM) that simulates legitimate traffic, while an attacker VM generates malicious traffic. A Server Cluster VMA is a multi-node virtual machine that simulates a server cluster. A DDoS Detection System VM runs a Python script to identify DDoS attacks. Client virtual machine: Use the Python scripting to create HTTP requests for the Server Cluster virtual machine. Several instances share traffic with the Server Cluster virtual machine. Use the suggested approach to find questionable traffic patterns and forward them to the virtual machine to detect DDoS attacks.

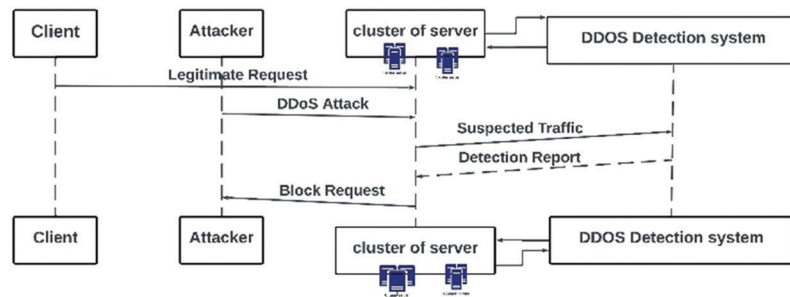


Fig. 6. The experiment testbed

The methodology has been tested in a live environment using QuasarRAT, JRat, and Black Shades, some of the most prevalent remote administration tools (RATs) readily available from both public and academic paper repositories. The results showed that none of these programs managed to evade our models and tools, as their operations consumed resources different from those of the user tasks. Detecting a DDoS attack is a binary classification with labels for benign and DDoS attacks. In this work, benign is seen as a standard class. An attack is considered a positive class because the interest is in finding out an assault, while an innocuous event is considered a negative class.

Analysis of the CICDDoS2019 dataset reveals that out of 1,048,575 network flow records, over 58% were classified as attacks, and approximately 42% were identified as legitimate network traffic flows. Fig. 7 illustrates how network traffic is visualized in the CICDDoS2019 dataset, as well as the unbalanced dataset.

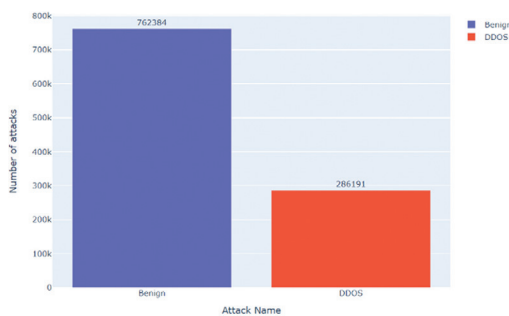


Fig. 7. Unbalanced CICDDoS2019 dataset

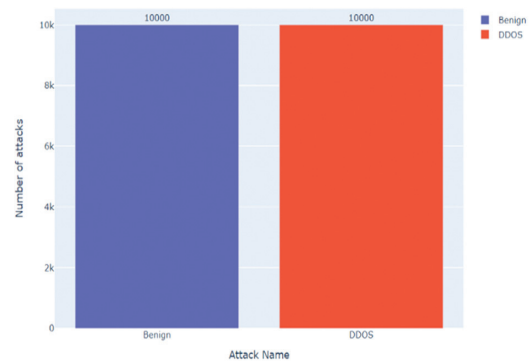


Fig. 8. Balanced dataset

Fig. 9 displays a sample of features that will be input into our model. These characteristics represent the outcomes of the preprocessing and feature selection stages. To provide a TCP connection activity, including the flags used during the handshake, the traffic direction, and the connection's current state.

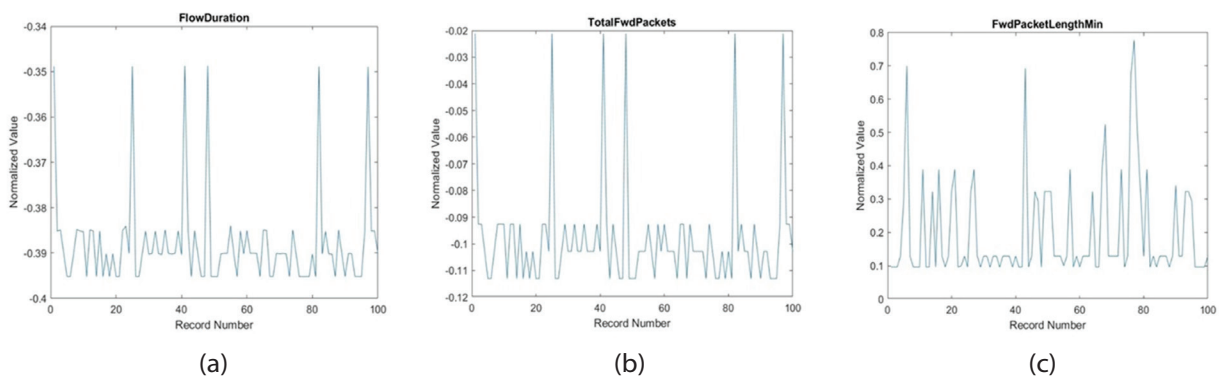


Fig. 9. Some samples of the normalization final features: (a) Flow duration Feature (b) forward packet total (c) Forward packet length min feature

The proposed approach begins with 100 and increases to 500 in 100-step increments, with other default settings. To evaluate the effectiveness of the splitting criterion, the dataset was partitioned into 70% for training and 30% for testing. Enumeration of hyperparameters is used in our proposed approach. Fig. 10 demonstrates a successful training process where the model's error (measured by MSE) significantly reduces and then stabilizes, indicating that the model has learned effectively and reached a point of convergence.

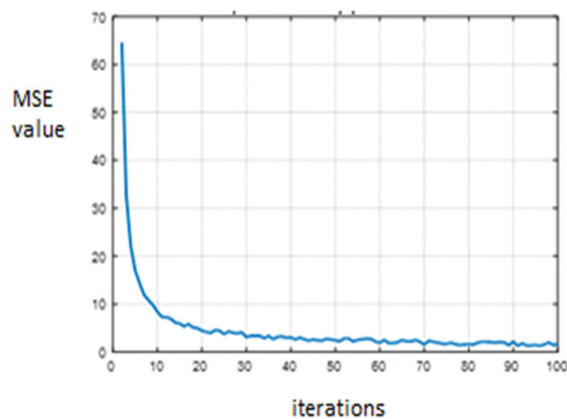


Fig.10. The MSE of ANN (training iterations)

Since our solution is based on pipeline ANN, Table 5 explains the hyperparameter tuning process. This process utilizes resources using the FIFO scheduling algorithm, emphasizing low complexity.

Table 5. Distributed hyperparameter tuning

Resource request	Memory Usage	Number of Trials	Learning Rates
Job requested 3 out of 5 available CPUs	3.5 GB of memory is currently used out of 29.00GB available on the node.	Total of 4 trials: 3 are running, and 1 has an error.	0.01
Ratio or a percentage (50%).	Ratio: $3.4 / 28.8 \approx 0.118$ (approximately 11.8%)	Ratio of Running Trials to Error Trials: $3 / 1 = 3$	Ratio:10
Macro average	0.50	0.50	0.50
Weight average	1.00	1.00	1.00

After numerous experiments in the testing phase, the ANN classifier was applied to detect DDoS attacks, and our model achieved a 99% accuracy (MSE of approximately 0.01). The classifiers exhibit nearly identical accuracy, with only a negligible discrepancy. To conclude, the topic at hand is resource consumption: Time Consumed: 51.5697112083435 seconds, Memory Consumed: 1.8 GB, and CPU Usage: 15.7%. Based on this comprehensive throughput, Pipelines can improve the efficiency of the development process and minimize redundant activities, enabling data scientists to spend more time on advanced tasks such as model selection and optimization. This can expedite overall advancement.

8. DISCUSSION

The outcomes achieved using pipeline machine-learning techniques are comparable to those reported by other researchers. Both models exhibit an accuracy of over 0.98, with a minimal false positive rate. Furthermore, our results demonstrate a very competitive false positive rate compared to models yielding the best outcomes.

Several of the models being compared lack the use of cross validation and, in certain instances, the confusion matrix, which is a fundamental approach to evaluation. Moreover, this model demonstrates substantial resource utilization and yields low-complexity outcomes by implementing a pipeline of machine learning approaches. This comparison demonstrates that the identical data set determines the classification accuracy. Table 6 presents the disparities between classification systems based on several parameters.

Table 6. compares recent DDOS detection models based on the CES-CICIDS2019 dataset

RF	Year	Methods	ML	Accuracy
[16]	2024	DDOS attack detection based on ML	Random Forest, SVM, Naive Bayes, KNN, XGBoost, and AdaBoos	Most accurate RF=99.9, AdaBoost and XGBoost =100
[36]	2024	Detection approach based on several MLalgorithms	Logistic Regression SVM, DT, RF, ANN, KNN	RF most accurate %99
Proposed model	2025	DDOS detection EHO features selection, and pipeline ANN	Pipeline ANN	%99.995 with a low complexity

While the proposed model's accuracy (99.99%) is comparable to existing approaches, its significant advantage lies in its lower false positive rate (FPR) (0.002%) and higher Precision (99.80%). As shown in Table 7, a lower FPR ensures fewer false alarms, making the system more reliable in real-world scenarios where excessive false positives can lead to service disruptions.

Table 7. Performance Comparison Table

Model	Accuracy	Precision	Recall	F1-score	False positive rate (FPR)
Proposed system	99.99%	99.80%	99.70%	99.75%	0.002%
Light Weiwei ML model [37]	98.72%	98.40%	97.80%	98.10%	0.005%
Deep learning [38]	98.55%	98.00%	97.50%	97.75%	0.006%
Transformer [39]	98.40%	97.80%	97.30%	97.55%	0.008%
CNN-LSTM [40]	98.30%	97.50%	97.00%	97.25%	0.009%

The proposed pipeline ANN + EHO model achieves the highest accuracy and the lowest false positive rate (FPR). Table 7 presents a comparative analysis of the proposed model using different performance measures with recent DDoS detection models. The results show that the proposed approach achieves an accuracy of 99.99% and a false positive rate of 0.002%. In contrast, the existing deep learning-based methods, such as CNN-LSTM [14] and transformer-based classifiers [39], demonstrate slightly lower precision and recall, resulting in a marginally higher false positive rate (FPR). This highlights the effectiveness of the proposed EHO-ANN model in distinguishing malicious traffic patterns while minimizing false alarms.

9. CONCLUSION

The DDOS attack is a risk that must be detected and prevented from harming the network system. This paper develops a pipeline ANN approach to detect a DDOS attack in a distributed manner.

Feature selection is used to reduce the dimension of the DDOS features and retain only relevant features that affect the attack's detection. A machine learning classifier, such as K-Nearest Neighbors (KNN), is used to evaluate the selected features. The final features are used for final training, while the pipelines are utilized to minimize the resources required for training techniques. The pipeline machine learning model has been shown to significantly impact both binary classifications. Moreover, the proposed solution demonstrates the effect of resource utilization in both the testing and training stages. The ANN pipeline model suggested in this paper can detect DDoS attacks that share comparable attributes. The model was evaluated using the CICDDoS2019 dataset. Though the proposed approach gives high detection accuracy, it also has certain limitations. First, the model has been primarily tested on the benchmark dataset CICDDoS2019, which may not fully represent real-world attack scenarios. Next, computational efficiency remains an issue, particularly in large-scale network deployments. Hence, future research should focus on optimizing model performance for real-time applications. This work will integrate adaptive learning techniques to enhance the detection of emerging attack patterns and validate the approach in real-world environments.

10. REFERENCE:

- [1] L. Eliyan, R. Di Pietro, "DoS and DDoS attacks in Software Defined Networks: A survey of existing solutions and research challenges", *Future Generation Computer Systems*, Vol. 122, 2021, pp. 149-171.
- [2] K. Adedeji, A. Abu-Mahfouz, A. Kurien, "DDoS Attack and Detection Methods in Internet-Enabled Networks: Concept, Research Perspectives, and Challenges", *Journal of Sensor and Actuator Networks*, Vol. 12, No. 4, 2023.
- [3] T. Ali, Y. Chong, S. Manickam, "Machine Learning Techniques to Detect a DDoS Attack in SDN: A Systematic Review", *Applied Sciences*, Vol. 13, No. 5, 2023, p. 3183.
- [4] K. Aggarwal, M. Mijwil, S. Sonia, A. Al-Mistarehi, S. Alomari, M. Gök, A. Alaabdin, S. Abdulrhman, "Has the Future Started? The Current Growth of Artificial Intelligence, Machine Learning, and Deep Learning", *Iraqi Journal for Computer Science and Mathematics*, Vol. 3, No. 1, 2022, pp. 115-123.
- [5] S. Muhamed, "Detection and Prevention WEB-Service for Fraudulent E-Transaction using APRIORI and SVM", *Al-Mustansiriyah Journal of Science*, Vol. 33, No. 4, 2022, pp. 72-79.
- [6] M. Mijwil, I. Salem, M. Ismaeel, "The Significance of Machine Learning and Deep Learning Techniques in Cybersecurity: A Comprehensive Review", *Iraqi Journal for Computer Science and Mathematics*, Vol. 4, No. 1, 2023, pp. 87-101.
- [7] X. He, K. Zhao, X. Chu, "AutoML: A survey of the state-of-the-art", *Knowledge-Based Systems*, Vol. 212, 2021.
- [8] I. Salem, M. Mijwil, A. Abdulqader, M. Ismaeel, A. Alkhazraji, A. Alaabdin, "Introduction to The Data Mining Techniques in Cybersecurity", *Mesopotamian Journal of CyberSecurity*, Vol. 2022, 2022, pp. 28-37.
- [9] P. Saini, S. Behal, S. Bhatia, "Detection of DDoS attacks using machine learning algorithms", *Proceedings of the 7th International Conference on Computing for Sustainable Global Development*, New Delhi, India, 12-14 March 2020, pp. 16-21.
- [10] R. Doriguzzi-corin, S. Millar, S. Scott-hayward, J. Mart, "LUCID: A Practical, Lightweight Deep Learning Solution for DDoS Attack Detection", *IEEE Transactions on Network and Service Management*, Vol. 17, No. 2, 2020, pp. 876-889.
- [11] V. Hnamte, A. Najjar, H. Nguyen, J. Hussain, M. Sugali, "DDoS attack detection and mitigation using deep neural network in SDN environment", *Computers & Security*, Vol. 138, 2024.
- [12] A. Mustapha. R. Khatoun, S. Zeadally, F. Chbib, "Detecting DDoS attacks using adversarial neural network", *Computers & Security*, Vol. 127, 2023.

- [13] M. Anley, A. Genovese, D. Agostinello, V. Piuri, "Robust DDoS attack detection with adaptive transfer learning", *Computers & Security*, Vol. 144, 2024, p. 103962.
- [14] M. Ouhssini, K. Afdel, E. Agherrabi, M. Akouhar, A. Abarda, "DeepDefend: A comprehensive framework for DDoS attack detection and prevention in cloud computing", *Journal of King Saud University - Computer and Information Sciences*, Vol. 36, No. 2, 2024, p. 101938.
- [15] Y. Beshah, S. L. Abebe, H. Melaku, "Drift Adaptive Online DDoS Attack Detection Framework for IoT System", *Electronics*, Vol. 13, No. 6, 2024, p. 1004.
- [16] U. Ashraf, H. Sharif, S. Usman, M. Hasnain, "A Machine Learning Based Approach for the Detection of DDoS Attacks on Internet of Things Using CICDDoS2019 Dataset - PortMapLahore Garrison University Research Journal of Computer Science and Information Technology", Vol. 8, No. 2, 2024.
- [17] F. Becerra-Suarez, I. Fernández-Roman, M. Forero, "Improvement of Distributed Denial of Service Attack Detection through Machine Learning and Data Processing", *Mathematics*, Vol. 12, No. 9, 2024, p. 1294.
- [18] M. Elsadig, "Detection of Denial-of-Service Attack in Wireless Sensor Networks: A Lightweight Machine Learning Approach", *IEEE Access*, Vol. 11, No. 2, 2023, pp. 83537-83552.
- [19] A. Silivery, K. Rao, L. Kumar, "An effective deep learning based multi-class classification of DoS and DDoS attack detection", *International Journal of Electrical and Computer Engineering Systems*, Vol. 14, No. 4, 2023, pp. 421-431.
- [20] S. Naiem, A. Kheder, A. Idrees, M. Marie, "Iterative feature selection-based DDoS attack prevention approach in cloud", *International Journal of Electrical and Computer Engineering Systems*, Vol. 14, No. 2, 2023, pp. 197-205.
- [21] A. Akinwale, E. Olajubu, A. Aderonmu, "A Regeneration Model for Mitigation Against Attacks on HTTP Servers for Mobile Wireless Networks", *International Journal of Electrical and Computer Engineering Systems*, Vol. 15, No. 5, 2024, pp. 395-406.
- [22] T. Hussein, "Deep Learning-based DDoS Detection in Network Traffic Data", *International Journal of Electrical and Computer Engineering Systems*, Vol. 15, No. 5, 2024, pp. 407-414.
- [23] P. Kumari, A. Jain, "A comprehensive study of DDoS attacks over IoT network and their counter-measures", *Computers & Security*, Vol. 127, 2023.
- [24] M. Mittal, K. Kumar, S. Behal, "Deep learning approaches for detecting DDoS attacks: a systematic review", *Soft Computing*, Vol. 27, No. 18, 2023, pp. 13039-13075.
- [25] S. Lee, Y. L. Shiue, C. Cheng, Y. Li, Y. Huang, "Detection and Prevention of DDoS Attacks on the IoT", *Applied Sciences*, Vol. 12, No. 23, 2022.
- [26] M. Gelgi, Y. Guan, S. Arunachala, M. Samba, N. Dragoni, "Systematic Literature Review of IoT Botnet DDOS Attacks and Evaluation of Detection Techniques", *Sensors*, Vol. 24, No. 11, 2024, p. 3571.
- [27] Z. Shah, I. Ullah, H. Li, A. Levula, K. Khurshid, "Blockchain Based Solutions to Mitigate Distributed Denial of Service (DDoS) Attacks in the Internet of Things (IoT): A Survey", *Sensors*, Vol. 22, No. 3, 2022, p. 1094.
- [28] N. Ahmed, Y. Mohialden, D. Abdulrazzaq, "A new method for self-adaptation of genetic algorithms operators", *International Journal of Civil Engineering and Technology*, Vol. 9, No. 11, 2018, pp. 1279-1285.
- [29] M. Ali, K. Balasubramanian, G. Krishnamoorthy, S. Muthusamy, S. Pandiyan, H. Panchal, S. Mann, S. Thangaraj, N. El-Attar, L. Abualigah, D. Abd Elminaam, "Classification of Glaucoma Based on Elephant-Herding Optimization Algorithm and Deep Belief Network", *Electronics*, Vol. 11, No. 11, 2022, p. 1763.
- [30] A. Malki, A. Mohamed, Y. Rashwan, R. El-Sehiemy, M. Elhosseini, "Parameter identification of photovoltaic cell model using modified elephant herding optimization-based algorithms", *Applied Sciences*, Vol. 11, No. 24, 2021.
- [31] A. Ismaeel, I. Elshaarawy, E. Houssein, F. Ismail, A. Hassanien, "Enhanced Elephant Herding Optimization for Global Optimization", *IEEE Access*, Vol. 7, 2019, pp. 34738-34752.
- [32] Y. Duan, C. Liu, S. Li, X. Guo, C. Yang, "Gradient-based elephant herding optimization for cluster

- analysis", *Applied Intelligence*, Vol. 52, No. 10, pp. 11606-11637, 2022.
- [33] A. Hagar, B. Gawali, "Deep Learning for Improving Attack Detection System Using CSE-CICIDS-2018", *NeuroQuantology*, 2022.
- [34] J. Leevy, T. Khoshgoftaar, "A survey and analysis of intrusion detection models based on CSE-CICIDS2018 Big Data", *Journal of Big Data*, Vol. 7, No. 1, 2020.
- [35] L. Yates, Z. Aandahl, S. Richards, B. Brook, "Cross validation for model selection: A review with examples from ecology", *Ecological Monographs*, Vol. 93, No. 1, 2023, pp. 1-24.
- [36] R. Gautam, R. Padmavathy, "Distributed denial of service attack detection using machine learning classifiers", *International Journal of Ad Hoc and Ubiquitous Computing*, Vol. 46, No. 3, 2024, pp. 123-149.
- [37] S. Sadhwani, B. Manibalan, R. Muthalagu, P. Pawar, "A lightweight model for DDoS attack detection using machine learning techniques", *Applied Sciences*, Vol. 13, No. 17, 2023, p. 9937.
- [38] A. Alahmadi, M. Aljabri, F. AL Haidari, D. Alharthi, "DDoS attack detection in IoT-based networks using machine learning models: A survey and research directions", *Electronics*, Vol. 12, No. 14, 2023, p. 3103.
- [39] G. Kirubavathi, I. Sumathi, J. Mahalakshmi, "Detection and mitigation of TCP-based DDoS attacks in cloud environments using a self-attention and intersample attention transformer model", *The Journal of Supercomputing*, Vol. 81, 2025, p. 474.
- [40] M. L. Viñuela, J.-A. R. Gallego, "Systematic Literature Review of Machine Learning Models for Detecting DDoS Attacks in IoT Networks", *Advances in Distributed Computing and Artificial Intelligence Journal*, Vol. 13, 2024.

Lettuce Yield Prediction: ElasticNet Regression Model (EINetRM) for Indoor Aeroponic Vertical Farming System

Original Scientific Paper

Gowtham Rajendiran*

Department of Computing Technologies, School of Computing,
College of Engineering and Technology, SRM Institute of Science and Technology,
Kattankulathur Campus-603203, Chengalpattu, Tamil Nadu, India
gr6047@srmist.edu.in

Jebakumar Rethnaraj

Department of Computing Technologies, School of Computing,
College of Engineering and Technology, SRM Institute of Science and Technology,
Kattankulathur Campus-603203, Chengalpattu, Tamil Nadu, India
jebakumr@srmist.edu.in

*Corresponding author

Abstract – Indoor aeroponic vertical farming systems have revolutionized agriculture by allowing efficient use of space and resources, eliminating the need for soil. These systems improve crop productivity and growth rates. However, accurately predicting lettuce yield in aeroponic environments remains a complex task due to the intricate interactions between environmental, nutrient, and growth parameters. This work aims to address these issues by integrating advanced sensor technologies with ElasticNet Regression Model (EINetRM) for its hybrid L1 and L2 regularization capabilities, handling multicollinearity and feature selection problems effectively in order to develop a reliable yield prediction framework. The predictive results showcases that the EINetRM model forecasts lettuce yield with high accuracy of 92% and less error score (RMSE) of 2.28 using a comprehensive dataset from a sensor-equipped indoor aeroponic system. Also, the results demonstrate the superior predictive power of EINetRM in capturing complex variable relationships, enhancing yield prediction reliability.

Keywords: indoor aeroponic vertical farming, elasticnet regression, machine learning, yield prediction

Received: January 29, 2025; Received in revised form: July 4, 2025; Accepted: July 7, 2025

1. INTRODUCTION

Machine learning applications in lettuce production is revolutionizing the farming, improving efficiency, adaptability, and sustainability is being examined in this research study. This study highlights the integration of convolutional neural networks (CNNs) and YOLO-based models in lettuce crop cultivation which involves pest and disease diagnosis, precision spraying, pesticide residue detection, crop condition monitoring, growth stage classification, yield prediction, weed management, and irrigation and fertilization management [1]. Another research highlights that the advanced Machine learning (ML) techniques are crucial for food security and hydroponic systems, but inconsistent predictions due to diverse features and datasets require further research. Integrating advanced ML

techniques with hydroponic systems holds promise for accurate yield forecasts and sustainability [2]. Artificial intelligence and IoT in aeroponics enable accurate regulation of fertilizer concentrations, misting cycles, temperature, and humidity. The integration of plasma-activated water and plasma-activated mist improves resource efficiency and plant health. Machine learning applications in lettuce production improve efficiency, adaptability, and sustainability [3].

In order to satisfy the increasing demands of the rising population, global food production must quadruple by 2050 [4, 5]. Nevertheless, the present rates of grain growth are insufficient to attain this objective [6]. Climate change impacts agriculture, potentially reducing crop production and increasing food scarcity. With a projected 9 billion global population by 2050, governments

must manage sudden crop availability disruptions [7, 8]. The Food and Agriculture Organization (FAO) reports a significant rise in grain demand and consumption in emerging countries like India between 1964 and 2030, with cereal imports increasing from 39 million tons in 1970 to 130 million tons between 1997 and 1999. This trend is expected to persist and potentially accelerate [9]. Importing conventional crops can lead to food security issues. Precision agriculture requires accurate crop forecasting, but weather conditions influence production. Models for accurate forecasting are needed for informed planning [10, 11]. Forecasting in applied sciences is crucial for precision farming techniques like aeroponic indoor farming. Machine learning techniques are being integrated into marketing software, equipment maintenance, health-monitoring systems, agricultural yield prediction, and soil analysis to improve crop growth and productivity [12].

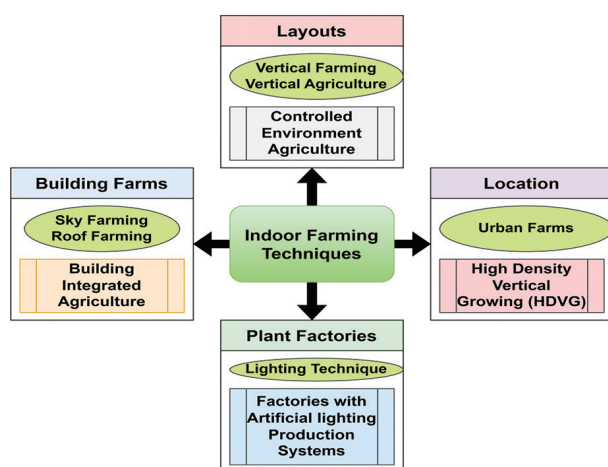


Fig. 1. Various forms of Vertical Farming

This research article provides a detailed description on the lettuce yield prediction with the efficacy of the ElasticNet machine learning regression model. This manuscript is organized in such a way, section-2 deals with the detailed literature survey, section-3 with the methodology involved, section-4 with the implementation results and discussions and finally, section-5 and section-6 with the conclusion and future scope of the research that leads to the further enhancement of the proposed work.

2. LITERATURE SURVEY

Using precision farming techniques integrated with the applications of machine learning algorithms, the researchers have analyzed, implemented and developed the crop prediction models in an effective manner.

Historically, crop-cutting experiments were used to quantify agricultural yield, but it's time-consuming and requires human labor. Currently, artificial neural networks, LASSO-Least Absolute Shrinkage and Selection Operator, and ELNET are used for predicting yields [13-17]. Sridhara *et al.* [18] used the LASSO, ENET, PCA, ANN, and SMLR

methodologies for predicting the Sorghum harvest. The researchers found that the developed artificial neural network (ANN) model outperformed the ENET regression model in estimating the yield of wheat crop.

Raja and Shukla [19, 20] used an Extreme Learning Machine (ELM) and hybrid grey-wolf-optimization Artificial Neural Network (ANN) models to get a more accurate forecast of the final bearing capacity and settlement of a geosynthetic-reinforced sandy soil.

Vertical farming, an innovative agricultural method, is attracting interest due to its capacity to enhance crop productivity per unit of land area [21, 22]. Nevertheless, the increase in expenses may have an adverse effect on profitability. This novel agricultural system tackles issues such as population expansion, limited cultivable area, and environmental limitations. Vertical farming efficiently utilizes space, minimizing the need for land and development in urban areas, while also catering to the increasing need for organic food [23]. Nevertheless, the presence of obstacles in sensor technologies, inventive cultivation approaches, energy optimization, and automation is anticipated to propel progress towards more effective production systems.

Modern technologies and sensors are used in vertical farming to keep an eye on the growing environments and make sure they are perfect for food growth, health, and development [24, 25]. This makes it easier to control energy and use resources more efficiently. Vertical farming is better for the earth than traditional farming because it increases food output and reduces trash [26]. However, it can be hard to combine sensors, control systems, and machine learning methods because they need complex robotics and data management systems.

Vertical farming technology is changing quickly, with a focus on gathering and analyzing data to get the best crop response [27]. This trend is good for the environment, society, and the economy [28, 29] and it looks like it will help keep food fresh in cities. Vertical farming has shown promise for growing a number of different crops, but more study is needed to make it more efficient and cost-effective. It has a lot of promise, but more study is needed to successfully apply it.

A lot of research has been done on different aspects of vertical farming, such as its types [21], how it works [28], how to control the environment and make the best use of resources [25, 30], how to build a smart indoor farm, sensing technologies, trends, and engineering challenges [31].

AI is a powerful computer program that lets computers learn from their mistakes, adapt to new information, and do jobs like people [32]. Vertical farming is a great example of how technology has changed the way food is grown [33]. AI, which is driven by machine-learning algorithms, looks at data and makes choices. It tracks plant growth, improves weather conditions, and makes the best use of resources [32]. AI finds trends and pre-

dicts plant health by looking at data from physical and image sensors [34]. This helps farmers make smart choices and get the most crops.

Recent technological advances are being used in vertical farming to lower costs and protect the environment [35]. AI is very important for keeping an eye on food growth and making output better. Color pictures are used for plant phenotyping under artificial lighting, which lets growers keep an eye on and improve crop growth all the time [36]. Hwang *et al.* [34] created an image-based system to track the growth of crops, and Vorapatratorn *et al.* [37] created an AI-powered system for plant farms to automatically run their operations. Crop-growth records from multiple planting rounds are saved and used to train machine-learning models for automatic plant growth [31]. Rizkiana *et al.* [38] used resilient backpropagation ANNs to guess how tall plants would get, taking into account external factors and the heights of the plants at the start. They did this by growing cabbage in a plant workshop.

A machine-vision method was used by Story *et al.* [39] to find cabbage grown in gardens suffers from deficiency of the essential nutrient called calcium. To showcase the difference between healthy and nutrient-deficient plants, they used a gray-level co-occurrence matrix and dual segmentation regression analysis. A group of researchers led by Hao [40] created a multi-scale hierarchical convolutional neural network (MFC-CNN) design to measure the amount of stress in leaves. Sun *et al.* [41] used a CNN design to collect features from RGB pictures and sensor data to figure out how much water plants were losing. A study by Gozzovelli *et al.* [42] used WGAN and a deep CNN architecture to identify the lettuce plants that were stressed by tip-burn.

AI is revolutionizing vertical farming by improving crop productivity, resource allocation, and automation, but small and medium-sized farms face challenges like high costs associated with collecting and analyzing extensive data from sensors and cameras [32, 43]. The absence of a common software platform may impede the incorporation of AI algorithms [33, 36]. In addition, the use of AI gives rise to issues about privacy and security, particularly in relation to the handling of sensitive data pertaining to crops, farmers, and customers particularly lettuce crops [42, 44].

Data quality challenges such as sensor noise, environmental unpredictability, and human error might impede the accuracy and effectiveness of AI in vertical farming [45, 46]. The process of training and implementing AI models might require a significant amount of resources, particularly for farms that are small or medium-sized. Gaining insight into the decision-making processes of AI models is essential for ensuring transparency and maintaining food safety [47, 45]. Ensuring fairness and accuracy in AI models necessitates the mitigation of bias [40, 44]. Notwithstanding these difficulties, AI has the capacity to transform vertical farming, enhancing its efficiency, productivity, and sustainabil-

ity. Vertical farming may enhance global food security, resource conservation, and environmental stewardship by effectively tackling these challenges and improving AI approaches [31].

3. METHODOLOGY

3.1. ELASTIC-NET REGRESSION FOR LETTUCE CROP YIELD PREDICTION

Lettuce, a popular leafy green herb, is a popular choice due to its health benefits. Aeroponic lettuce can be grown in a controlled environment with nutrient solution sprays. However, yield prediction of lettuce crop in conventional and precision farming is crucial. Machine learning has emerged as a solution for predicting lettuce crop yield effectively. The Elastic-Net (ENet) regression algorithm is chosen for this work, which is a balanced fitting of the dataset, combining LASSO and RIDGE regularization algorithms. This algorithm is chosen for its ability to handle high-dimensional data, handle noise and outliers, optimize hyper-parameters, adapt to new data, make decisions, and provide statistical insights on prediction results. High variance algorithms like decision trees and KNN are examples of high bias.

3.2. MATHEMATICS BEHIND ELASTICNET REGRESSION

Multiple Linear Regression involves more than one independent variable and one dependent variable. It is mathematically represented as,

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n \quad (1)$$

where, Y is the dependent variable, $X_1, X_2, \dots, X_n$ are the independent variables, β_0 is the intercept and $\beta_1, \beta_2, \dots, \beta_n$ are the slopes.

The main goal of the algorithm is to find the best fit line equation that can predict the values based on the independent variables. In regression, set of records (dataset) are present with X and Y values and these values are used to learn a function so that to predict Y from an unknown X this learned function can be used. So here, to find the value of Y , a function is required that predicts continuous Y in the case given X as independent features.

In order to best fit line in linear regression, its not easy to get it easily in real life cases so we need to calculate errors that affects it. These errors to be calculated to mitigate them. Thus, the equation for calculating the error function or cost function is represented as,

$$J = \sum_{i=1}^n (\hat{y}_i - y_i)^2 \quad (2)$$

where, J is the error function, $\hat{y}_i$ is the predicted values and y_i is the actual values.

The purpose is to determine the optimal values for the intercept θ_0 and the coefficient of the input feature θ_2 providing the best fit line for representing this relationship,

$$\hat{y}_i = \theta_0 + \theta_2 x_i \quad (3)$$

In order to reduce the error function, the parameter values need to be updated. The technique behind this is to start θ_1 and θ_2 with random values and iteratively update until reaching the minimum error or cost.

So the cost function with respect to θ_1 ,

$$J_{\theta_1} = \frac{\partial(\theta_1, \theta_2)}{\partial \theta_1} \quad (4)$$

$$= \frac{\partial}{\partial \theta_1} \left[\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2 \right] \quad (5)$$

$$= \frac{1}{n} \left[\sum_{i=1}^n 2(\hat{y}_i - y_i) \left(\frac{\partial}{\partial \theta_1} (\hat{y}_i - y_i) \right) \right] \quad (6)$$

$$= \frac{1}{n} \left[\sum_{i=1}^n 2(\hat{y}_i - y_i) \left(\frac{\partial}{\partial \theta_1} (\theta_1 + \theta_2 x_i - y_i) \right) \right] \quad (7)$$

$$= \frac{1}{n} \left[\sum_{i=1}^n 2(\hat{y}_i - y_i) (1 + 0 - 0) \right] \quad (8)$$

$$= \frac{1}{n} \left[\sum_{i=1}^n 2(\hat{y}_i - y_i) \right] \quad (9)$$

$$J_{\theta_1} = \frac{2}{n} \left[\sum_{i=1}^n (\hat{y}_i - y_i) \right] \quad (10)$$

Similarly, for finding J_{θ_2} , the equation is represented below.

$$J_{\theta_2} = \frac{2}{n} \left[\sum_{i=1}^n (\hat{y}_i - y_i) x_i \right] \quad (11)$$

Hence, finding the coefficients of a linear equation that best fits the training data is the objective of the linear regression. The respective intercept and coefficient X will be if α is the learning rate.

$$\theta_1 = \theta_1 - \alpha(J_{\theta_1}) \quad (12)$$

$$= \theta_1 - \alpha \left[\frac{2}{n} \left[\sum_{i=1}^n (\hat{y}_i - y_i) \right] \right] \quad (13)$$

Similarly,

$$\theta_2 = \theta_2 - \alpha(J_{\theta_2}) \quad (14)$$

$$= \theta_2 - \alpha \left[\frac{2}{n} \left[\sum_{i=1}^n (\hat{y}_i - y_i) x_i \right] \right] \quad (15)$$

Since, the Elastic Net model is also a linear regression model that incorporates all the functionalities of

multiple linear regression model. In addition, it also embeds a composite penalty term including both L_1 (Lasso) and L_2 (Ridge) regularization techniques. The objective function of Elastic Net is the sum of the L_1 and L_2 penalty terms, which are added to the ordinary least squares (OLS) objective function. The Elastic Net objective function is expressed in a generic form as follows:

$$Objective = \sum_{i=1}^N (y_i - \hat{y}_i)^2 + \alpha (\lambda_1 \sum_{j=1}^p |\beta_j| + \lambda_2 \sum_{j=1}^p \beta_j^2) \quad (16)$$

where, N is the number of observations; y_i is the actual output for observation i ; $\hat{y}_i$ is the predicted output for observation i ; p is the number of features (predictors); β_j is the coefficient of feature j ; α is the elastic net mixing parameter, controlling the balance between L_1 and L_2 regularization; λ_1 is the L_1 regularization strength; λ_2 is the L_2 regularization strength.

The Elastic Net algorithm aims to find the values of β_j that minimize this objective function. The regularization terms $(\lambda_1 \sum_{j=1}^p |\beta_j| + \lambda_2 \sum_{j=1}^p \beta_j^2)$ help prevent overfitting by penalizing large coefficient values. The mixing parameter α allows you to control the trade-off between L_1 and L_2 regularization.

When $\alpha=0$, the Elastic Net reduces to Ridge regression, and when $\alpha=1$, it reduces to Lasso regression.

The Elastic Net approach is particularly useful when dealing with datasets where many features are correlated, as it can select groups of correlated features together (similar to Lasso) while still providing some of the shrinkage properties of Ridge. This makes Elastic Net a versatile choice for feature selection and regularization in linear regression models.

4. FLOW DIAGRAM, IMPLEMENTATION RESULTS AND DISCUSSIONS

The results and discussions section deals with the detailed notes on various phases from dataset collection to the comprehensive evaluation of the experimental results produced during the implementation of improved ElasticNet regression model on the aeroponic lettuce crop yield estimation as shown in Fig. 2.

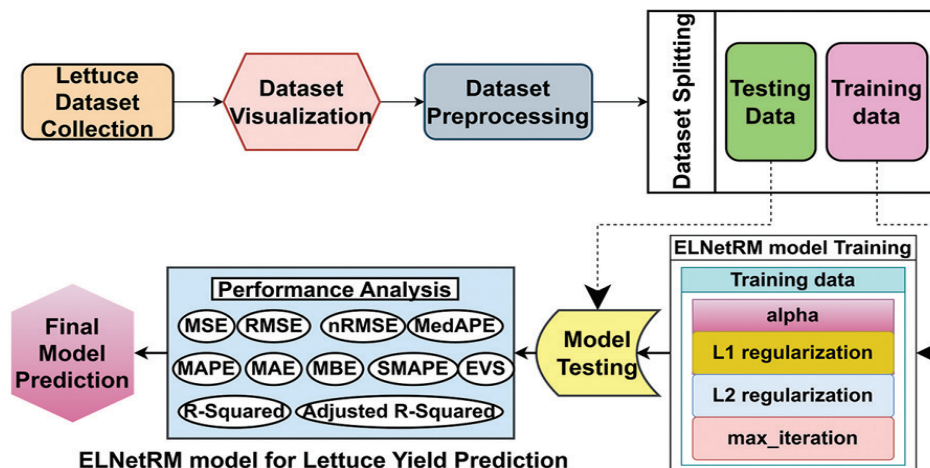


Fig. 2. Architecture diagram of Lettuce Yield prediction using ElnetRM model

4.1. DATASET COLLECTION AND VISUALIZATION

Lettuce growth dataset have been collected from the aeroponic farming Centre for a cycle of 45-60 days. The sample dataset is shown in Fig. 3.

```
dataset.head()
```

	pH	TDS	Temperature	EC	Turbidity	Humidity	Light	Growth	Yield
0	6	150.0	28.0	0.29	197.0	55.023	9.7	1.0	20.453
1	6	953.0	27.0	1.72	196.0	85.023	9.7	2.0	21.876
2	6	898.0	27.0	0.28	195.0	115.023	9.7	3.0	23.000
3	6	892.0	27.0	1.34	194.0	145.023	9.7	4.0	27.521
4	6	819.0	27.0	1.84	193.0	175.023	9.7	5.0	30.000

Fig. 3. Sample lettuce growth dataset

There are nine different lettuce features were utilized for training the EInetRM model. All the parameters listed here are related to the indoor or controlled environment agriculture; they are nutrient solution characteristics such as power of hydrogen (pH), total dissolved salts (TDS), electrical conductivity (EC), turbidity and related to the crop growth ambience namely, temperature, humidity, light, growth and yield.

In order to better understand the lettuce datasets utilized for the implementation process, the datasets

were visually represented in the form of pictures. Fig. 3 showcases the distribution plot of the parameter pH with respect to the other input variables.

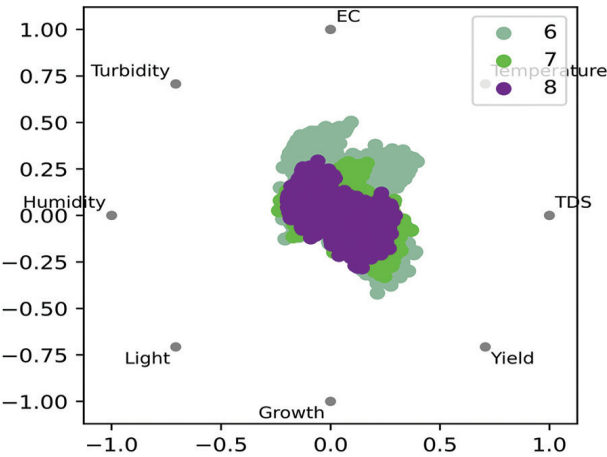


Fig. 4. Distribution Plot of pH parameter

Though there are multiple number of visualization plots are available, histogram technique (bootstrap distribution) was highly utilized to represent the features in an individual manner or as a whole. Fig. 5 showcases the distribution of lettuce growth parameters as separate plots in such a manner that one feature do not collide with another.

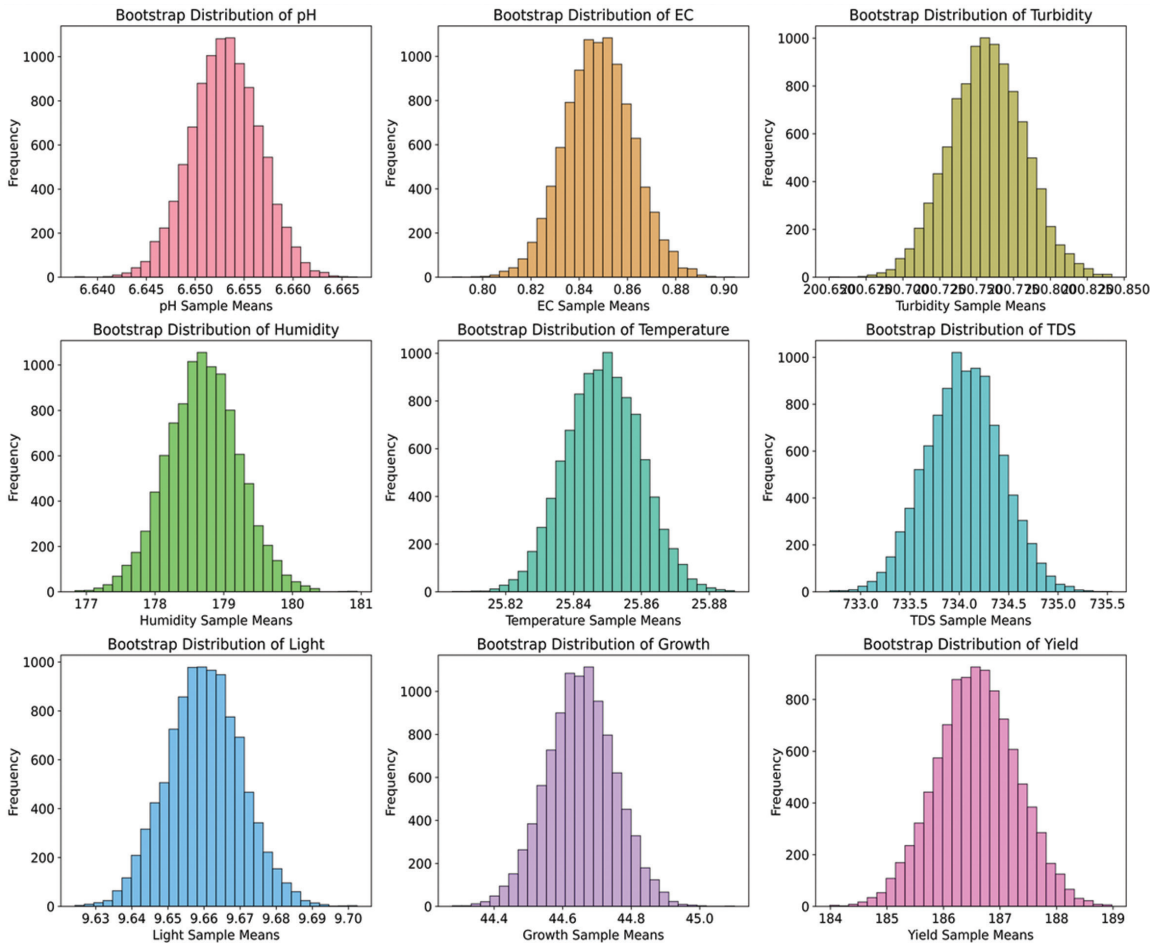


Fig. 5. Bootstrap Distribution Plot of lettuce growth parameters

4.2. PREPROCESSING TOWARDS MODEL TESTING

One of the most important steps before training any machine learning models is preprocessing. Here, the authors have utilized the data cleaning technique called removal of outliers in order to achieve high yield prediction accuracy.

Fig. 6. Represents the boxplot visualization of the original dataset size comprising of 225792 features [25088

rows multiplied with 9 columns]. One of the foremost used preprocessing technique is the removal of outliers, which are being removed from the original dataset with the help of Inter Quartile Range (IQR) method. The features which does not fall within the quartiles i. e within the specified inliers between 25th and 75th quartiles were neglected during the model training.

Once the outliers are identified and removed (cleaned), once again the preprocessed dataset were visualized in Fig. 7.

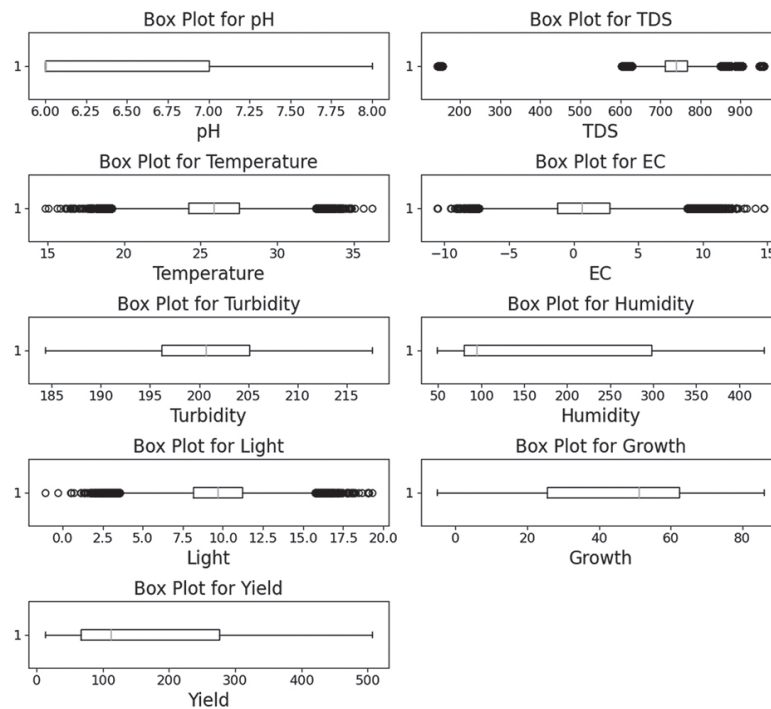


Fig. 6. Boxplot representing dataset with outliers

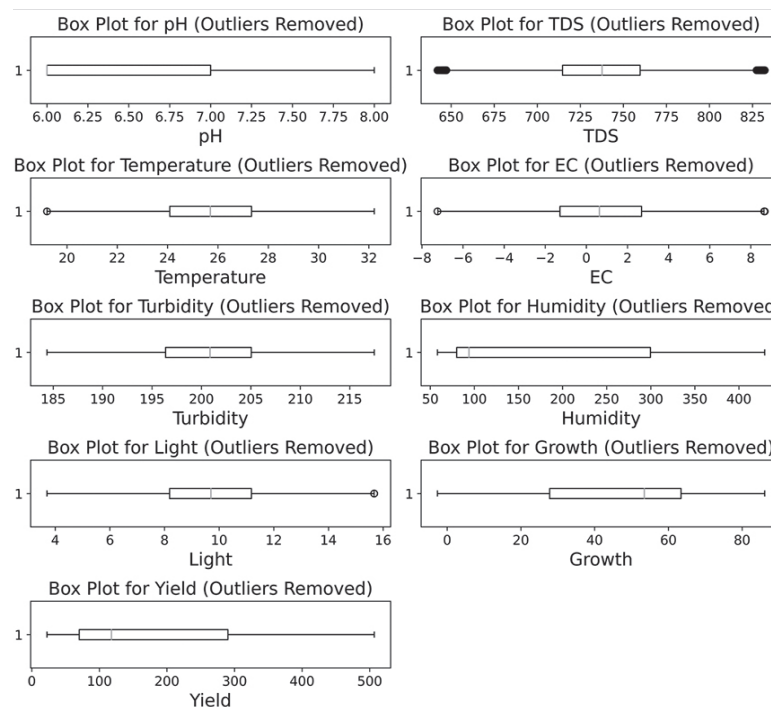


Fig. 7. Boxplot representing dataset without outliers

After preprocessing, the size of the dataset size is reduced to 205251 features i.e [225792-20541 outlier features=205251 features].

4.3. FEATURE SELECTION, DATASET SPLITTING, HYPERPARAMETER TUNING, MODEL TRAINING AND TESTING

Once the dataset is being preprocessed, the core part of the machine learning implementation begins to pro-

cess. The initial step is to select the certain set of lettuce growing features such as pH, EC, light, turbidity, temperature, TDS, turbidity, humidity and growth.

All these growing factors were considered as the independent variables. When the values of these individual crop growth variables changes, it has the significant impact on the lettuce cultivation. The strength and nature of their interrelationships with each other and the output variable can be graphically represented through the correlogram image as represented in Fig. 8.



Fig. 8. Correlation diagram of input variables

The next step is the dataset splitting where the pre-processed dataset was split into training and testing datasets in the ratio of 80:20 (where 80% of data utilized for training and 20% of data utilized for testing) for efficient model building.

Once the splitting process is done, the training phase starts. The EInetRM model was trained on the lettuce dataset for learning the underlying patterns. The performance is being measured and kept aside. Obviously, for the first time, the performance would be poor which has to fine-tuned for the further improvements in the yield prediction.

So, the hyperparameter tuning process gets initiated. As the ElasticNet regression is the combination of Lasso and Ridge regression, after performing hyperparameter tuning, the values of the regularization parameters such as Alpha and L_{1ratio} are fixed to 0.5 i.e moderate regularization for alpha and equal contribution of $[L_1$ and $L_2]$ penalties.

After the model gets trained with the training dataset, the model is exposed to the testing dataset where the real research work was concentrated to prove the efficiency

of the improved model. Here, hyper-parameter tuning via cross-validation was not utilized as the fixed threshold "0.5" provides the least error metric and higher accuracy.

4.4. PERFORMANCE METRICS USED FOR MODEL EVALUATION

This is the final phase of the regression model where various performance metrics were highly utilized for evaluating the EInetRM model's performance. They are listed below in an elaborated manner.

Mean Squared Error (MSE)

It is the average of the squared differences between the predicted ($\hat{y}_i$) and the actual values (y_i).

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (17)$$

Root Mean Squared Error (RMSE)

It is the square root of the average of the squared differences between the predicted and the actual values.

$$RMSE = \sqrt{MSE} \quad (18)$$

Normalized Root Mean Squared Error (nRMSE)

It differs from the RMSE metric that normalizes the RMSE by dividing it by the range of the target variable. This normalization allows for comparing the performance of the models on the datasets with different scales.

$$nRMSE = \frac{RMSE}{\max(y) - \min(x)} \quad (19)$$

Mean Absolute Error (MAE)

It measures the average absolute differences between the predicted and the actual values. It provides the more interpretable measures of the average magnitude of errors.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (20)$$

Mean Bias Error (MBE)

It is used to evaluate the bias or systematic error in a regression model. It measures the average difference between the predicted and the actual values. This metric does not provide the information about the spread or variability of errors, so it is often used in conjunction with other metrics like MAE or RMSE.

$$MBE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i) \quad (21)$$

Median Absolute Percentage Error (MedAPE)

The metric measures the median of absolute percentage differences between predicted and actual values, providing a reliable measure of prediction accuracy, especially in the presence of outliers. It is scale-independent and suitable for comparing models across different datasets, allowing understanding of typical error magnitudes in terms of percentages of actual values.

$$MedAPE = \text{median} \left(\frac{|y_i - \hat{y}_i|}{|y_i|} \times 100 \right) \quad (22)$$

Absolute Percentage Error (MAPE)

It measures the average percentage error between the predicted and the actual values. A lower MAPE indicates better model performance.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left(\frac{|y_i - \hat{y}_i|}{|y_i|} \times 100 \right) \quad (23)$$

Mean Percentage Error (MPE)

This metric is used to evaluate the accuracy of predictions in a regression model. It measures the average percentage difference between the predicted and the actual values. It is similar to MBE which provides the information on whether the model tends to overestimate or underestimate the actual values.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left(\frac{y_i - \hat{y}_i}{y_i} \times 100 \right) \quad (24)$$

Symmetric Mean Absolute Percentage Error (SMAPE)

It addresses some of the limitations of other percentage-based metrics by providing a symmetric view of

the percentage errors that is not affected by the scale of the data. This metric is expressed as a percentage and it ranges from 0 to 200%. A lower the SMAPE score, better the performance of the model.

$$SMAPE = \frac{1}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{\left(\frac{|y_i| + |\hat{y}_i|}{2} \right)} \times 100\% \quad (25)$$

R-Squared metrics

R^2 metrics or the coefficient of determination used to evaluate the goodness of fit of a regression model. It is the widely used metric for prediction and regression models, but it has limitations such as its sensitivity to the number of predictors and inability to distinguish between the good and bad predictions in certain rare cases.

It represents the proportion of the variance in the dependent variable that is explained by the independent variables.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (26)$$

where, y_i is the actual values, $\hat{y}_i$ is the predicted values and $\bar{y}$ is the mean of actual values.

Adjusted R-Squared metrics

Adjusted R^2 is a metric that addresses the limitations of R^2 metrics by accounting for the number of predictors in a regression model. It penalizes the inclusion of unnecessary predictors that do not significantly improve the model, providing a more realistic measure of a model's explanatory power. This metric is particularly useful in preventing inflation of R^2 due to the inclusion of irrelevant predictors, making it a more accurate measure of a model's explanatory power.

$$\text{Adjusted } R^2 = 1 - \frac{(1 - R^2)(n-1)}{(n-p-1)} \quad (27)$$

where, R^2 is the regular R -squared, n is the number of observations and p is the number of predictors.

Explained Variance Score (EVS)

EVS is a metric that assesses the variance in dependent variables explained by a regression model, similar to R -squared metrics. It prioritizes the model's ability to capture target variable variability, considering both bias and variance of predictions. Its scale differs from R -squared metrics, ensuring a more accurate understanding of the model's performance.

$$EVS = 1 - \left[\frac{\text{var}(y - \hat{y})}{\text{var}(y)} \right] \quad (28)$$

where, y is the actual values, $\hat{y}$ is the predicted values and var is the variance of actual values.

4.5. PERFORMANCE ANALYSIS

Table 1 Represents the lettuce yield prediction performance analysis using the improved EINetRM model with the help of number of distinct performance metrics.

Table 1. Performance Analysis of ElasticNet regression with other Regression methodologies

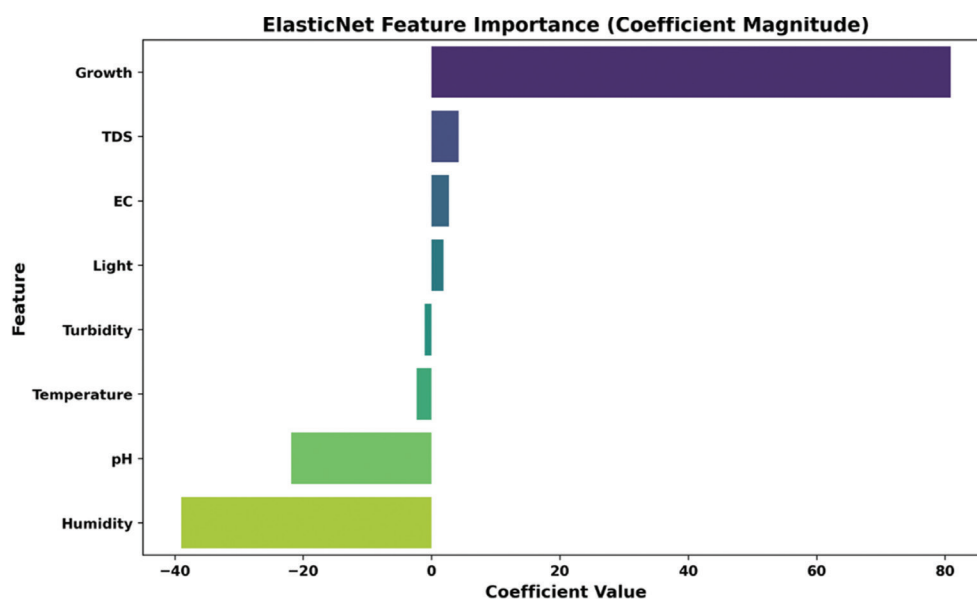
Regression Type	Performance Measurements											
	MSE	RMSE	nRMSE	MAE	MBE	MedAPE	MAPE	MPE	SMAPE	R-squared	Adjusted R-squared	EVS
Linear Regression	8.591	2.931	0.3094	7.667	0.58	4.85	4.20	1.41	46.332	0.89	0.90	0.721
Lasso Regression	8.640	2.94	0.3012	7.64	0.47	4.15	4.07	1.27	43.076	0.84	0.85	0.703
Ridge Regression	8.591	2.931	0.2012	6.73	0.40	4.03	4.07	1.63	41.375	0.87	0.88	0.83
Support Vector	7.246	2.691	0.1922	6.67	0.406	4.026	4.06	1.59	41.026	0.88	0.89	0.8256
Random Forest	6.875	2.622	0.1568	6.49	0.391	4.006	3.99	1.48	40.368	0.90	0.906	0.854
ElasticNet regression	5.239	2.288	0.1015	6.24	0.38	3.92	3.74	1.04	39.51	0.91	0.92	0.88

The use of the EInetRM in predicting lettuce production in controlled indoor aeroponic environments produced valuable findings in this carried out research. The Mean Squared Error (MSE) of 5.239 is the average of the squared differences between the predicted and actual yields. The Root Mean Squared Error (RSME) of 2.288, calculated as the square root of the Mean Squared Error (MSE), represents the standard deviation of the residuals and indicates the accuracy of the model. The NRMSE of 0.1015 indicated the model's high accuracy in predicting the yield range. The Mean Absolute Error (MAE) of 6.24 is the average absolute difference between the predicted and actual values, serving as a concise measure of the model's performance. The model's predictions showed a small Mean Bias Error (MBE) of 0.38, indicating a minor tendency to underestimate. The investigation also found a Median Absolute percentage Error (MedAPE) of 3.92, indicating that the minimum number of the predictions differed from the actual values by this proportion. The Mean Absolute Percentage Error (MAPE) of 3.74 indicates the average percentage difference, while the Mean Percentage Error (MPE) of 1.04 suggests a tiny value of underestimate. The Symmetric Mean Absolute Percentage Error (SMAPE) of 39.51 represented a measure of symmetric percentage difference where, this minimum symmetric

error showcases the better training and testing of the model on the data. The R-squared value of 0.91 demonstrates a strong correlation between the model and the real data, indicating its strength on lettuce yield prediction. The EInetRM demonstrated resilience, as shown by its Adjusted R-squared value of 0.92, which takes into account the number of predictors. In addition, the Explained variance Score (EVS) of 0.88 accurately quantified the amount of fluctuation in lettuce production that was successfully accounted for by the EInetRM. In conclusion, these results collectively suggest that the EInetRM stands out as a promising and accurate approach for predicting indoor aeroponic lettuce yield, substantiated by a comprehensive evaluation of diverse metrics.

4.6. FEATURE IMPORTANCE VISUALIZATION

The visualization of feature importance for the proposed research work has been represented using the Coefficient bar plot which is useful when many number of input features were being utilized for predictions. Here, from the plot we can observe and interpret the most influential variables that affects the growth of lettuce crop based on the positive (bars >0) and the negative impact (bars <0) as represented in Fig. 9.

**Fig. 9.** Coefficient magnitude diagram of input variables

4.7. PREDICTION GRAPH

In the context of aeroponic lettuce yield prediction by the improved EInetRM model, the prediction graph as represented in Fig. 10. highlights the efficiency of utilizing the regression model using the scatter plot.

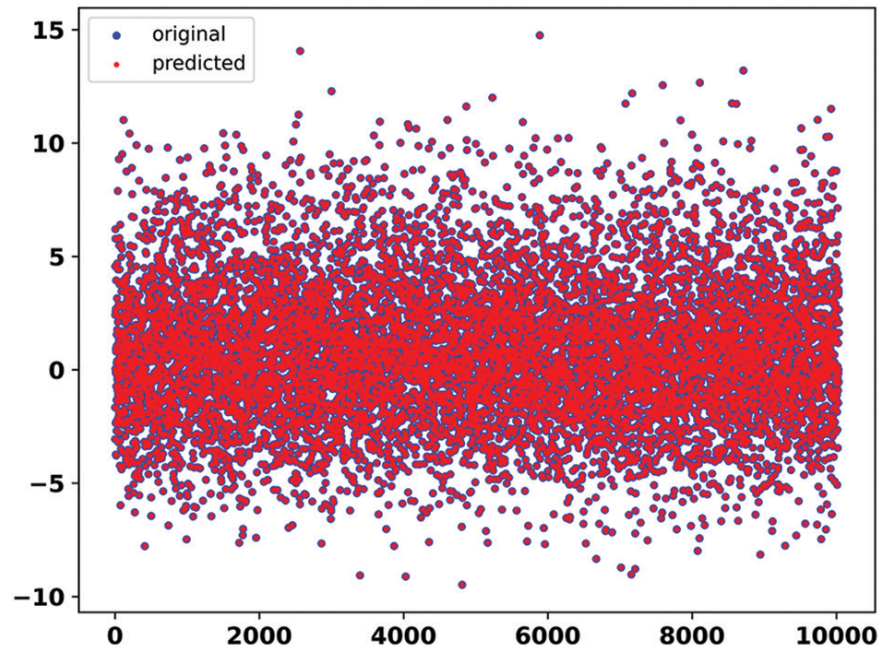


Fig. 10. Lettuce Yield Prediction Graph (Scatter Plot)

The blue line in the prediction graph represents the original values or the actual values (input parameters) of the lettuce dataset and the other color represents the predicted values (lettuce yield in weights) by the EInetRM model. As the predicted values overlaps with the actual values, the successful prediction by the model is highly depicted while the small discrepancies highlight the areas of further improvement in the prediction process.

5. CONCLUSION AND FUTURE SCOPE

Ultimately, this research work evaluates the Elastic Net machine learning regression model's effectiveness in accurately forecasting lettuce production in indoor aeroponic systems. The model effectively handles the complex dynamics of indoor aeroponic settings, addressing multicollinearity and balancing sparsity and variable relevance. The results show superior performance compared to conventional models like linear regression and Ridge regression. The model's interpretability enhances its usefulness by providing valuable insights for decision-making in indoor aeroponic lettuce growth. The study also contributes to precision agriculture knowledge by highlighting the special benefits of Elastic Net in indoor aeroponic systems. The findings have practical applications for farmers, agronomists, and researchers involved in enhancing agricultural yield in controlled settings.

The Future research should focus on fine-tuning parameters, integrating additional variables, dynamic model adaptation, ensemble approaches, practical im-

plementation, and extension to other crops. The model's performance can be improved by incorporating environmental and nutrient variables and advanced sensor data. On-farm trials and validations should assess the model's feasibility in real-world indoor aeroponic farming scenarios. The ongoing exploration and refinement of machine learning models will contribute to precision farming, resource utilization, and sustainable food production.

6. REFERENCES:

- [1] S. Sharmin, M. T. Hossan, M. S. Uddin, "A review of machine learning approaches for predicting lettuce yield in hydroponic systems", *Smart Agricultural Technology*, Vol. 11, 2025, p. 100925.
- [2] Y.-M. Qin, Y.-H. Tu, T. Li, Y. Ni, R.-F. Wang, H. Wang, "Deep learning for sustainable agriculture: A systematic review on applications in lettuce cultivation", *Sustainability*, Vol. 17, No. 7, 2025, p. 3190.
- [3] W. A. Qureshi, J. Gao, O. Elsherbiny, A. H. Mosha, M. H. Tunio, J. A. Qureshi, "Boosting aeroponic system development with plasma and high-efficiency tools: AI and IoT—A review", *Agronomy*, Vol. 15, No. 3, 2025, p. 546.
- [4] The resource outlook to 2050: By how much do land, water and crop yields need to increase by

2050?, Food and Agriculture Organization (FAO), Rome, Italy, 2009.

- [5] D. Tilman, C. Balzer, J. Hill, B. L. Befort, "Global food demand and the sustainable intensification of agriculture", *Proceedings of the National Academy of Sciences, USA*, Vol. 108, 2011, pp. 20260-20264.
- [6] D. K. Ray, N. D. Mueller, P. C. West, J. A. Foley, "Yield trends are insufficient to double global crop production by 2050", *PLoS One*, Vol. 8, 2013, p. e66428.
- [7] R. Bhadouria et al. "Agriculture in the era of climate change: Consequences and effects", *Climate Change and Agricultural Ecosystems*, Elsevier, 2019, pp. 1-23.
- [8] D. Deryng, D. Conway, N. Ramankutty, J. Price, R. Warren, "Global crop yield response to extreme heat stress under multiple climate change futures", *Environmental Research Letters*, Vol. 9, 2014, p. 034011.
- [9] "World Agriculture: Towards 2015/2030", <https://www.fao.org/3/y3557e/y3557e08.htm> (accessed: 2023)
- [10] A. K. Singh et al. "Development of multi stage district level wheat yield forecast models", *Journal of Agricultural Physics*, Vol. 14, 2014, pp. 189-193.
- [11] S. D. Attri, L. S. Rathore, "Pre-harvest estimation of wheat yield for NW India using climate and weather forecast", *Mausam*, Vol. 54, 2003, pp. 729-738.
- [12] M. Casdagli, S. Eubank, J. D. Farmer, J. Gibson, "State space reconstruction in the presence of noise", *Physica D: Nonlinear Phenomena*, Vol. 51, 1991, pp. 52-98.
- [13] B. Das et al. "Comparative evaluation of linear and nonlinear weather-based models for coconut yield prediction in the west coast of India", *International Journal of Biometeorology*, Vol. 64, 2020, pp. 1111-1123.
- [14] B. Das, B. Nair, V. K. Reddy, P. Venkatesh, "Evaluation of multiple linear, neural network and penalised regression models for prediction of rice yield based on weather parameters for west coast of India", *International Journal of Biometeorology*, Vol. 62, 2018, pp. 1809-1822.
- [15] S. Sridhara et al. "Weather-based neural network, stepwise linear and sparse regression approach for rabi sorghum yield forecasting of Karnataka, India", *Agronomy*, Vol. 10, 2020, p. 1645.
- [16] A. K. Singh, A. Vashisth, P. Krishnan, B. Das, "Wheat yield prediction based on weather parameters using multiple linear, neural network and penalised regression models", *Journal of Agrometeorology*, Vol. 24, 2022, pp. 18-25.
- [17] B. Das, D. Murgaonkar, S. Navyashree, P. Kumar, "Novel combination artificial neural network models could not outperform individual models for weather-based cashew yield prediction", *International Journal of Biometeorology*, Vol. 66, 2022, pp. 1627-1638.
- [18] S. Khaki, L. Wang, "Crop yield prediction using deep neural networks", *Frontiers in Plant Science*, Vol. 10, 2019, p. 621.
- [19] M. N. A. Raja, S. K. Shukla, "An extreme learning machine model for geosynthetic-reinforced sandy soil foundations", *Proceedings of the Institution of Civil Engineers - Geotechnical Engineering*, Vol. 175, 2022, pp. 383-403.
- [20] M. N. A. Raja, S. K. Shukla, "Predicting the settlement of geosynthetic-reinforced soil foundations using evolutionary artificial intelligence technique", *Geotextiles and Geomembranes*, Vol. 49, 2021, pp. 1280-1293.
- [21] A. M. Beacham, L. H. Vickers, J. M. Monaghan, "Vertical farming: A summary of approaches to growing skywards", *Journal of Horticultural Science and Biotechnology*, Vol. 94, 2019, pp. 277-283.
- [22] C. Nájera, V. M. Gallegos-Cedillo, M. Ros, J. A. Pascual, "Role of spectrum-light on productivity, and plant quality over vertical farming systems: Bibliometric analysis", *Horticulturae*, Vol. 9, 2023, p. 63.
- [23] Statista, "Global vertical farming market projection 2022-2032", <https://www.statista.com/statistics/487666/projection-vertical-farming-market-worldwide/> (accessed: 2023)
- [24] H. Li, Y. Guo, H. Zhao, Y. Wang, D. Chow, "Towards automated greenhouse: A state of the art review on greenhouse monitoring methods and technologies based on internet of things", *Computers*

- and Electronics in Agriculture, Vol. 191, 2021, p. 106558.
- [25] C. Vatisstas, D. D. Avgoustaki, T. Bartzanas, "A systematic literature review on controlled-environment agriculture: How vertical farms and greenhouses can influence the sustainability and footprint of urban microclimate with local food production", *Atmosphere*, Vol. 13, 2022, p. 1258.
- [26] S. Oh, C. Lu, "Vertical farming—Smart urban agriculture for enhancing resilience and sustainability in food security", *Journal of Horticultural Science and Biotechnology*, Vol. 98, 2023, pp. 133-140.
- [27] F. Kalantari, O. Mohd Tahir, A. M. Lahijani, S. Kalantari, "A review of vertical farming technology: A guide for implementation of building integrated agriculture in cities", *Advanced Engineering Forum*, Vol. 24, 2017, pp. 76-91.
- [28] K. Al-Kodmany, "The vertical farm: A review of developments and implications for the vertical city", *Buildings*, Vol. 8, 2018, p. 24.
- [29] F. Sandison, J. Yeluripati, D. Stewart, "Does green vertical farming offer a sustainable alternative to conventional methods of production?: A case study from Scotland", *Food and Energy Security*, Vol. 12, 2023, p. e438.
- [30] T. Kozai, "Towards sustainable plant factories with artificial lighting (PFALs) for achieving SDGs", *International Journal of Agricultural and Biological Engineering*, Vol. 12, 2019, pp. 28-37.
- [31] G. Rajendiran, J. Rethnaraj, "Smart aeroponic farming system: Using IoT with LCGM-Boost regression model for monitoring and predicting lettuce crop yield", *International Journal of Intelligent Engineering & Systems*, Vol. 16, No. 5, 2023, pp. 251-262.
- [32] M. Siropayan, O. Celikel, O. Pinarer, "Artificial intelligence driven vertical farming management system", *Proceedings of the World Congress on Engineering 2022*, London, UK, 6-8 July 2022, pp. 108-113.
- [33] T. Talaviya, D. Shah, N. Patel, H. Yagnik, M. Shah, "Implementation of artificial intelligence in agriculture for optimisation of irrigation and application of pesticides and herbicides", *Artificial Intelligence in Agriculture*, Vol. 4, 2020, pp. 58-73.
- [34] Y. Hwang, S. Lee, T. Kim, K. Baik, Y. Choi, "Crop growth monitoring system in vertical farms based on region-of-interest prediction", *Agriculture*, Vol. 12, 2022, p. 656.
- [35] P. Morella, M. P. Lambán, J. Royo, J. C. Sánchez, "Vertical farming monitoring: How does it work and how much does it cost?", *Sensors*, Vol. 23, 2023, p. 3502.
- [36] B. Franchetti, V. Ntouskos, P. Giuliani, T. Herman, L. Barnes, F. Pirri, "Vision based modeling of plants phenotyping in vertical farming under artificial lighting", *Sensors*, Vol. 19, 2019, p. 4378.
- [37] S. Vorapatratorn, "Development of automatic plant factory control systems with AI-based artificial lighting", *Proceedings of the 13th International Conference on Information Technology and Electrical Engineering*, Chiang Mai, Thailand, 14-15 October 2021, pp. 69-73.
- [38] A. Rizkiana et al. "Plant growth prediction model for lettuce (*Lactuca sativa*) in plant factories using artificial neural network", *IOP Conference Series: Earth and Environmental Science*, Vol. 733, 2021, p. 012027.
- [39] D. Story, M. Kacira, C. Kubota, A. Akoglu, L. An, "Lettuce calcium deficiency detection with machine vision computed plant features in controlled environments", *Computers and Electronics in Agriculture*, Vol. 74, 2010, pp. 238-243.
- [40] X. Hao et al. "MFC-CNN: An automatic grading scheme for light stress levels of lettuce (*Lactuca sativa* L.) leaves", *Computers and Electronics in Agriculture*, Vol. 179, 2020, p. 105847.
- [41] Y. Sun, Y. Liu, G. Wang, H. Zhang, "Deep learning for plant identification in natural environment", *Computational intelligence and neuroscience*, Vol. 2017, 2017, p. 7361042.
- [42] R. Gozzovelli, B. Franchetti, M. Bekmurat, F. Pirri, "Tip-burn stress detection of lettuce canopy grown in plant factories", *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, Montreal, BC, Canada, 2021, pp. 1259-1268.
- [43] G. Kaur, P. Upadhyaya, P. Chawla, "Comparative analysis of IoT-based controlled environment and uncontrolled environment plant growth monitor-

ing system for hydroponic indoor vertical farm", Environmental Research, Vol. 222, 2023, p. 115313.

- [44] V. Galaz et al. "Artificial intelligence, systemic risks, and sustainability", Technology in Society, Vol. 67, 2021, p. 101741.
- [45] S. Nagano et al. "Leaf-movement-based growth prediction model using optical flow analysis and machine learning in plant factory", Frontiers in Plant Science, Vol. 10, 2019, p. 227.
- [46] D. Vela et al. "Temporal quality degradation in AI models", Scientific Reports, Vol. 12, 2022, p. 11654.
- [47] H. Wang et al. "Leaf area index estimation for a greenhouse transpiration model using external climate conditions based on genetic algorithms, back-propagation neural networks and non-linear autoregressive exogenous models", Agricultural Water Management, Vol. 183, 2017, pp. 107-115.

Ensemble Deep Learning Approach For Multi-Class Skin Cancer Classification

Original Scientific Paper

Ali Abdulameer*

Electrical Engineering Technical College, Middle Technical University
Baghdad, Iraq
bdc0063@mtu.edu.iq

Raaed Hassan

Electrical Engineering Technical College, Middle Technical University
Baghdad, Iraq
drxaaed_alanbaki@mtu.edu.iq

Abbas Humadi

Electrical Engineering Technical College, Middle Technical University
Baghdad, Iraq
drabbas@mtu.edu.iq

*Corresponding author

Abstract – Skin cancer is one of the most prevalent types of cancer, often caused by prolonged exposure to ultraviolet (UV) radiation, such as sunlight. This cancer is mainly categorized into benign and malignant lesions, where the latter could cause severe complications and even death. Traditional diagnostic methods, such as visual inspection and dermoscopy, often lack accuracy, while biopsy, though highly accurate, is invasive, time-consuming, and costly. This study aims to develop an automated deep learning model that leverages an ensemble of “Convolutional Neural Networks” (CNNs) to perform four-class classification of common skin lesions: Basal Cell Carcinoma (BCC), Benign Keratosis Lesion (BKL), Melanocytic Nevus (NV), and Melanoma (MEL). Seven widely used CNNs in medical imaging, GoogLeNet, InceptionV3, Xception, ResNet18, ResNet50, ResNet101, and DenseNet201, were evaluated for their performance in this classification task. The ISIC2018 and ISIC2019 datasets were employed, and data augmentation techniques were applied to address dataset imbalances. The analysis identified InceptionV3, Xception, and DenseNet201 as the top-performing networks. Therefore, they are utilized for the ensemble model. These networks were used as feature extractors, and their output features were combined and classified using a “Support Vector Machine” (SVM) algorithm. This approach demonstrates the potential of combining CNNs and SVM in an ensemble framework to enhance the accuracy and reliability of automated skin cancer classification. The proposed model achieved an accuracy of 94.46%, outperforming individual CNNs (93.27%) and existing ensemble methods such as Max Voting (94.12%) and hybrid models like DenseNet201 with Random Forest (91.28%).

Keywords: Skin Cancer, Deep Learning, CNN, Machine Learning, Artificial Intelligence

Received: March 6, 2025; Received in revised form: June 12, 2025; Accepted: June 12, 2025

1. INTRODUCTION

Skin cancer is one of the most common cancers and was the fourth most diagnosed cancer in 2020 [1]. In the United States, 9,500 persons are diagnosed with skin cancer every day [2]. It is mainly caused by ultraviolet (UV) radiation from sources such as sunlight or tanning devices, leading to DNA mutations and consequently to abnormal growth of skin cells [3]. This type of carcinoma commonly occurs in fair-skinned individuals due to their lack of melanin pigmentation, which protects the skin from UV

light [3]. There are many types of this cancer according to malignancy and origin. Melanoma is considered the most dangerous lesion, and early detection of this type is critical because it is highly malignant and can cause severe complications or even death. Statistics show that one person dies from every 7.8 melanoma cases [4]. Diagnosis typically begins with visual inspection of the lesion, but its accuracy is below 60%, even among experienced dermatologists [5]. Dermoscopy, a non-invasive procedure using a light source and magnifying lenses, was introduced to improve diagnostic accuracy by enhancing visualization

of tumor features achieving diagnostic accuracy between 75% and 84% [6]. Accurate diagnosis often requires a biopsy, an invasive procedure in which a skin sample is extracted and examined microscopically [7]. A biopsy is an expensive and time-consuming procedure, making early diagnosis of this serious disease difficult and sometimes inaccessible in rural or underserved areas [7].

Computer-Aided Diagnosis (CAD) systems were developed to assist dermatologists in accurately diagnosing skin lesions [8]. Early systems employed machine learning (ML) algorithms such as logistic regression, KNN, decision trees, random forest, SVM, and ANN, relying on manually inserted ABCDE features. This manual feature extraction was tedious and limited, especially since certain melanoma types, such as nodular melanoma, do not conform to the ABCDE criteria [9]. Later, convolutional neural networks (CNNs) were introduced as automated feature extractors, using convolutional filters trained via backpropagation to identify patterns [10]. The increased availability of GPUs in personal computers enabled researchers to efficiently train CNNs on dermoscopic images [11].

Bazgir *et al.* [12] conducted a binary classification to differentiate between benign and malignant skin lesions using a modified version of the InceptionV3 network. The modification involved adding an extra dense layers at the end of the original architecture. They achieved a maximum classification accuracy of 85.94%, which is relatively low for a binary classification task. We believe this is because they trained their modified network from scratch, without utilizing the pre-trained weights of InceptionV3.

Dahdouh *et al.* [13] conducted a seven-class skin cancer classification using the HAM10000 dataset. They integrated a convolutional neural network (CNN) with reinforcement learning (RL), where a Q-network replaced the CNN's dense layer. Preprocessing and segmentation steps were also applied. The model achieved a classification accuracy of approximately 80%. However, the proposed CNN architecture was not reported, and the specific contribution of the Q-network remains unclear, as the results section did not present the CNN's performance without RL integration.

Dogan and Ozdemir [14] developed a hybrid model to distinguish benign lesions from melanoma by evaluating multiple pre-trained CNNs, ResNet152V2, VGG16, Xception, InceptionV3, MobileNetV2, DenseNet201, InceptionResNetV2, and EfficientNetB2, in combination with machine learning algorithms such as K-Nearest Neighbors (KNN) and Random Forest (RF). The best performance was achieved using DenseNet201 as a feature extractor combined with Random Forest as the classifier, yielding an accuracy of approximately 91.28%. This result highlights the effectiveness of using a hybrid approach that integrates CNN-based feature extraction with traditional ML classifiers.

Natha *et al.* [15] conducted a seven-class skin cancer classification using three machine learning algorithms: Random Forest (RF), Multi-layer Perceptron Neural

Network (MLPN), and Support Vector Machine (SVM), which were used as classifiers and combined using the Max Voting method. Color and texture features were extracted using basic image processing techniques, and a genetic algorithm was employed to optimize the feature vector by selecting the most relevant features extracted from the ISIC2018 dataset. The study achieved a classification accuracy of approximately 94.70%, demonstrating the effectiveness of an ensemble approach that combines multiple algorithms.

Researchers achieved relatively high diagnostic accuracy compared to traditional methods, but they remain far from achieving 100% accuracy. This challenge arises due to the high visual similarity between different skin lesions. For instance, melanocytic nevi look similar to melanoma, but the two lesion types differ in their malignancy. Another limitation is the lack of sufficient and balanced dermoscopic image datasets. This research aims to address these limitations by introducing the following contributions:

- Eliminating rare skin lesion types from the dataset to increase the diagnostic performance on the common types, BCC, BKL, MEL, and NV.
- Introducing a new data balancing approach that combines external dermoscopic images from multiple datasets with targeted augmentation techniques to address the class imbalance problem inherent in dermatological datasets.
- Utilizing a distinctive feature-level fusion method that directly concatenates high-dimensional deep features extracted from three diverse CNN architectures without dimensionality reduction, preserving the complete feature information for classification.
- Utilization of a computationally efficient SVM classifier that leverages the concatenated feature vector rather than traditional voting-based ensemble methods or prediction score fusion approaches.

The rest of the paper is organized as follows: Section 2 describes the methodology, including the dataset, preprocessing techniques, CNN model, support vector machine model, and ensemble model. Section 3 presents the results, and Section 4 provides the conclusion, discussion, and directions for future work.

2. METHODOLOGY

2.1. DATASET

The ISIC2018 dataset [16] was used to train the CNN models. It contains seven lesion classes: vascular lesion (VASC), dermatofibroma (DF), "benign keratosis lesion" (BKL), "actinic Keratoses and intraepithelial carcinoma" (AKIEK), "Melanocytic Nevus" (NV), melanoma (MEL), and "basal cell carcinoma" (BCC). The ISIC2018 dataset is divided into training, validation, and test sets containing 10,015; 193; and 1,512 dermoscopic images, respectively. This dataset suffers from class imbalance,

with the NV class comprising 67% of the total images, while the DF class accounts for only 1%.

This imbalance is addressed in this work through two main contributions. The first approach involves two steps: first, excluding rare lesion types from the classification task due to their uncommon occurrence. This results in focusing on four common classes: BCC, BKL, NV, and MEL.

Secondly, additional images are imported from the ISIC2019 dataset. Since the ISIC2019 dataset includes all images from ISIC2018, the import process ensures that no duplicate images are included in the combined dataset. The distribution of the resulting dataset, referred to as "ISIC2018+" for simplicity, is illustrated in Fig. 1.

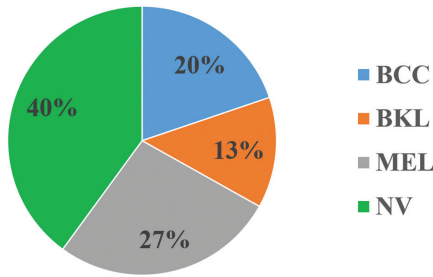


Fig. 1. The percentage of classes in the ISIC2018+ dataset

The second contribution involves data augmentation to increase the number of images for underrepresented classes. The augmentations include image scaling in the range of 1 to 1.2, image flipping along the x-axis and y-axis, and image rotation from -90° to 90°. All these augmentations are applied with random parameters to balance the dataset equally across the four classes. As a result, the augmented dataset achieves an equal distribution, with each class representing 25% of the data.

As a result, the ISIC2018+ dataset contains 16,787 images, while the ISIC2018+(augmented) comprises 26,819 images. Fig. 2 shows an example of augmented images generated by the augmentation process. The validation and test sets remained unchanged after excluding the underrepresented categories, AKIEC, DF, and VASC, from the original dataset. This study adheres to using the official test set provided by the ISIC archive to ensure the validity and comparability of evaluation results. Randomly selecting a test subset from the resulting dataset is discouraged, as it does not meet strict evaluation standards.

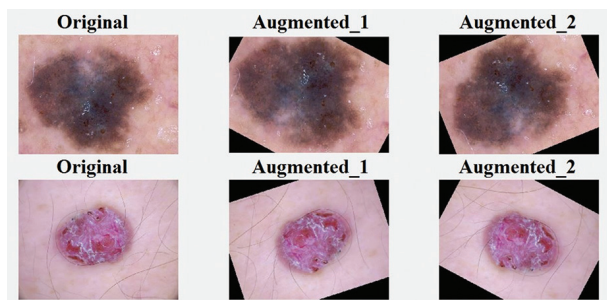


Fig. 2. Example of image augmentation process

2.2. PEPROCESSING TECHNIQUES

The preprocessing procedure in this work includes image resizing and normalization. Image resizing is essential to adapt dermoscopic images to the required input dimensions of the CNN model. For example, GoogLeNet requires input images of size 224x224 pixels. Normalization converts pixel intensity values from the [0,255] range to the [0,1] range. This process helps prevent issues such as exploding gradients and ensures that all features contribute equally to gradient updates. By scaling pixel intensities, brighter pixels do not overpower dimmer ones, which stabilizes and accelerates the training process. Normalization is performed using Eq. (1) [10].

$$x_{normalized} = \frac{x}{255} \quad (1)$$

Here, x represents the pixel intensity of the input image.

2.3. CNN MODEL

This research utilizes CNN as an automated feature extractor. CNN captures image patterns at various layers, extracting features ranging from low-level details to high-level abstractions. Filters (kernels) with initially random weights are updated during training to minimize the error between true labels and predicted probabilities [10]. For classification tasks, the "Cross-Entropy Loss Function" is preferred over "Mean Squared Error" (MSE) because it penalizes confident incorrect predictions more heavily, enabling faster convergence during training. "Cross-Entropy Loss Function" is computed in Eq. (2) [10].

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_{i,c} (\log \hat{y}_{i,c}) \quad (2)$$

Here, N is the total number of samples. C is the number of classes. $y_{i,c}$ is the true label for the i^{th} sample and c^{th} class. $\hat{y}_{i,c}$ is the predicted probability for (i, c) indices.

The convolution process is mathematically expressed in Eq. (3) [10].

$$y_{[i,j,c]} = \sum_{m=1}^K \sum_{n=1}^K \sum_{d=1}^D x_{[i+m-1,j+n-1,d]} \cdot w_{[m,n,d,c]} + b_{[c]} \quad (3)$$

Here, $x_{[i,j,d]}$ represents the pixel value at indices (i, j) in the d^{th} channel of the input feature map. Similarly, $y_{[i,j,c]}$ represents the pixel value at indices (i, j) in the c^{th} channel of the output feature map.

The weight tensor, $w_{[m,n,d,c]}$, corresponds to the filter indexed by (m, n) , which defines the kernel dimensions, and (d, c) , which specifies the input and output channels of the feature maps. K is the kernel size, D is number of input channels, $b_{[c]}$ is the bias term which is added to each output channel.

The spatial size of the output feature map of every convolution process is governed by Eq. (4) [10].

$$Z = \frac{I-K+2P}{s} + 1 \quad (4)$$

Here, Z is the spatial size of the output feature map. I is the spatial size of the input matrix. K is the kernel

size. P is the padding applied around the input matrix. S is the stride value.

In this research, seven commonly used CNNs are employed for transfer learning to perform skin cancer classification: GoogLeNet, InceptionV3, Xception, ResNet18, ResNet50, ResNet101, and DenseNet201. Their selection is justified by their architectural diversity, varying depths, and proven success in medical imaging applications [17, 18]. These models represent distinct design philosophies in deep learning, each offering unique strengths. The CNN models are customized to perform a four-class classification task. This customization involves replacing the dense layer with one having four outputs, and the classification layer with one that outputs four class probabilities. The properties of those seven models are detailed in Table 1.

Table 1. Properties of CNN models

Network name	Depth	Input size	Parameter memory	No. of parameters (millions)
GoogLeNet	22	224×224	27 MB	7.0
InceptionV3	48	299×299	91 MB	23.9
Xception	71	299×299	88 MB	22.9
DenseNet201	201	224×224	77 MB	20.0
ResNet18	18	224×224	45 MB	11.7
ResNet50	50	224×224	98 MB	25.6
ResNet101	101	224×224	171 MB	44.6

2.4. SUPPORT VECTOR MACHINE MODEL

Support Vector Machine (SVM) is a widely used machine learning algorithm primarily designed for binary classification tasks. It can also be extended to multi-class classification by employing a one-vs-all approach. The core concept involves using a hyperplane to separate data points into two classes within a high-dimensional space. The hyperplane is represented by Eq. (5) [19].

$$W^T \cdot X_i + b = 0 \quad (5)$$

Here, W is the weight vector normal to the hyperplane. X_i is the feature vector for the i^{th} point. b is the bias term.

The weight vector is optimized to maximize the margin, defined as the perpendicular distance between the hyperplane and the nearest data points. This margin is symmetric on both sides of the hyperplane and is given by $(2/\|W\|)$, as illustrated in a 2-dimensional perspective in Fig. 3.

To maximize the margin, the denominator term $(\|W\|)$ of the margin should be minimized. For mathematical convenience, the term $((1/2)\|W\|^2)$ is used instead for two key reasons. First, the squared term simplifies optimization by enabling the use of “Convex Optimization Techniques,” Second, the constant factor $(1/2)$ makes the calculation of derivatives with respect to (W) more efficient. To train the model, each data point should be

considered with its true label (y_i) , where $y_i \in \{1, -1\}$ represents the class labels for binary classification. For a correctly classified point, the following constraint must be satisfied as expressed in Eq. (6) [19].

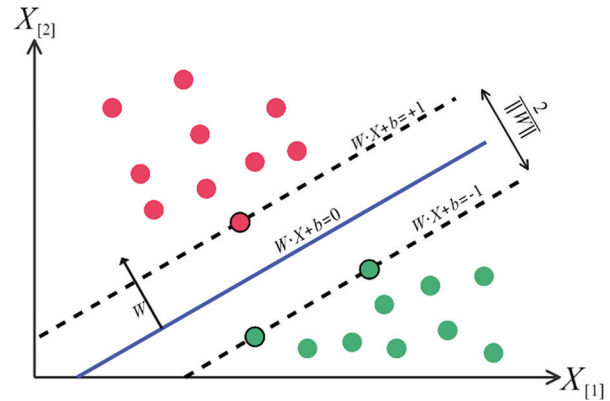


Fig. 3. Two-dimensional perspective of the hyperplane of SVM

$$y_i(W^T \cdot X_i + b) \geq 1 \quad (6)$$

The “Lagrange Multipliers” method is utilized to optimize the model and update weights, where the loss function L is introduced by Eq. (7) [19].

$$\mathcal{L}_{(W,b,\alpha)} = \frac{1}{2} \|W\|^2 - \sum_{i=1}^n \alpha_i (y_i(W^T \cdot X_i + b) - 1) \quad (7)$$

Where the first term $((1/2)\|W\|^2)$ measures the value responsible for maximizing the margin, while the second term adds a penalty for violating the constraint in Eq. (6). This penalty is scaled by the Lagrange multipliers (α_i) . This method focuses on points near the hyperplane, known as support vectors while excluding other points. This property makes it computationally efficient during the optimization process. As a result, a new data point (X) is classified based on the sign of the hyperplane equation as expressed in Eq. (8) [19].

$$y(X) = \text{sign}(W^T \cdot X + b) \quad (8)$$

2.5. ENSEMBLE MODEL

In this work, an ensemble model has been proposed which includes three main steps. First, seven CNN architectures are trained on the ISIC2018+ dataset to identify the top-performing models. Second, the best-performing models from the previous step are trained on the ISIC2018+(augmented) dataset individually. Finally, The output features from the top-performing models in the first two steps are concatenated into a single matrix with full dimensionality, which serves as the input to an SVM model for generating the final predictions. In essence, the proposed ensemble model leverages CNNs as feature extractors and utilizes an SVM for classification. The CNN features are extracted from the inputs to the dense layers of each network. The complete methodology for this work is illustrated in Fig. 4, where the chosen models are justified in the results section.

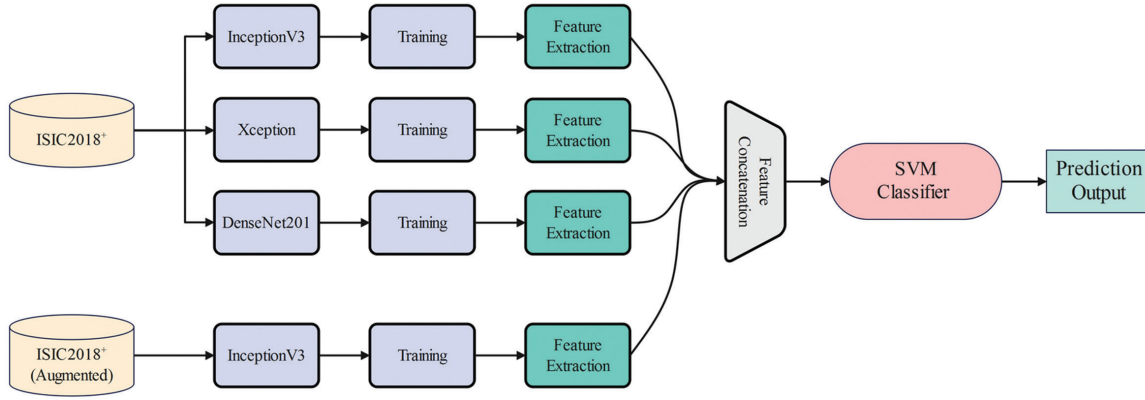


Fig. 4. The proposed ensemble model architecture

2.6. TRAINING PROCESS

The training is conducted in a MATLAB environment using the Deep Learning Toolbox (version 14.3) to customize and train the CNN architectures. Additionally, the Statistics and Machine Learning Toolbox (version 12.2) is employed to implement the SVM classifier. All experiments are performed on a "Dell Precision 7740" laptop equipped with an "NVIDIA Quadro RTX 5000" featuring 16 GB of GDDR6 VRAM. The CNN training process utilizes a scheduled learning rate, which starts at 0.01 and decays by a factor of 1/10 every 10 epochs, over a total of 30 epochs. Training is carried out using the "Stochastic Gradient Descent with momentum" (SGDM) optimizer, configured with a momentum value of 0.9. All hyperparameters are detailed in Table 2.

Table 2. Training hyperparameters

Hyperparameter	Value
Learning rate	(0.01), (0.001), (0.0001)
epochs	30
Optimizer	SGDM
Batch size	64
Momentum	0.9
L2 Regularization	0.0001

The same augmentation techniques used in the ISIC2018*(augmented) dataset are employed during the training of the CNN models to mitigate overfitting and enhance the model's generalization. However, the augmentation process is explained in details in Section 2.1.

2.7. EVALUATION METRICS

In this task, evaluation metrics are essential to rate the performance of the deep learning model. The test set is imported from the ISIC archive, which has 1512 dermoscopic images for seven classes. After the elimination of the minor classes, a test set is achieved with four classes, BCC, NV, BKL, and MEL, forming 1390 images in total. In this section, accuracy, recall, precision, F1-score, ROC ("Receiver Operating Characteristic") curve, AUC ("Area Under Curve"), and the confusion matrix are explained.

Accuracy is the main evaluation metric used in a deep learning context. It is simply the ratio of the cor-

rect predictions to the total number of predictions. It's expressed further in Eq. (9) as multi-class accuracy.

$$\text{Multi Class Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \quad (9)$$

Here, TP is true positive predictions. TN is true negative predictions. FN is false negative predictions. FP is false positive predictions.

Since multi-class accuracy could be misleading for imbalanced datasets, another accuracy metric is considered in this research, the mean accuracy, which is expressed in Eq. (10).

$$\text{Accuracy}_{class(k)} = \frac{TP_{class(k)}}{\text{Total number of prediction per class}(k)} \quad (10)$$

Precision is another metric that expresses the ratio of the positive predictions over all positive predictions. as expressed in Eq. (11).

$$\text{Precision} = \frac{TP}{TP+FP} \quad (11)$$

Recall gives the indication of how the actual positive predictions are correctly identified as expressed in Eq. (12).

$$\text{Recall} = \frac{TP}{TP+FN} \quad (12)$$

F1-Score balances precision and recall by taking the harmonic mean as expressed in Eq. (13).

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (13)$$

A confusion matrix summarizes true positives, true negatives, false positives, and false negatives to evaluate classification performance. The ROC curve shows the trade-off between the true positive and false negative rates across thresholds, while the AUC represents the area under the ROC curve.

3. RESULTS

3.1. PERFORMANCE EVALUATION

This section outlines the evaluation process of this work, which is divided into three steps. The first step

evaluates the performance of seven CNN models with softmax classifier trained on the ISIC2018+ to identify the best-performing networks. In the second step, only the top-performing models from the previous step are trained individually on the ISIC2018+(augmented) dataset. This approach avoids the need to train all seven models on the larger dataset, saving time and effort. The third step involves selecting the best models from the previous steps to be used as feature extractors for

the ensemble model. Table 3 summarizes the evaluation results. InceptionV3, Xception, and DenseNet201 achieved the highest accuracy.

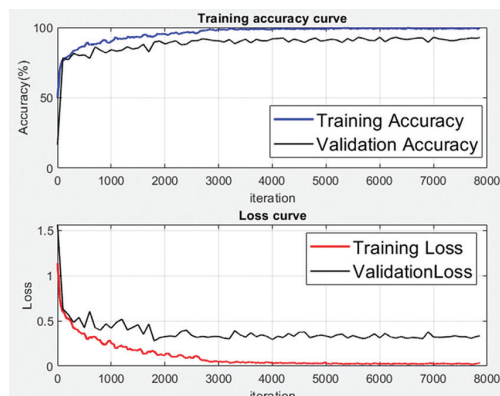
Their superior performance is attributed to architectural strengths: InceptionV3 and Xception combine depth and width to capture multi-scale lesion features, while DenseNet201's dense connections promote feature reuse and reduce redundancy.

Table 3. The evaluation metrics for the experienced CNN models.

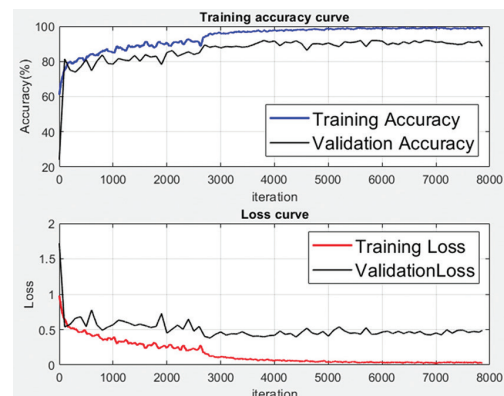
Dataset	CNN model	Precision	Recall	F1-Score	AUC	Mean Accuracy	Multi-Class Accuracy
ISIC2018+	ResNet18	77.47%	75.71%	76.50%	95.17%	92.09%	84.17%
	ResNet50	75.25%	74.42%	74.73%	93.94%	91.55%	83.10%
	ResNet101	75.54%	75.47%	75.33%	94.48%	91.40%	82.80%
	GoogLeNet	77.42%	74.52%	75.79%	94.48%	92.13%	84.25%
	InceptionV3	77.97%	81.01%	79.20%	95.65%	92.56%	85.10%
	Xception	80.00%	83.31%	80.95%	96.27%	92.82%	85.54%
	DenseNet201	80.45%	78.75%	79.19%	95.59%	92.95%	85.90%
ISIC2018+ (augmented)	InceptionV3	80.78%	80.80%	80.50%	95.75%	93.27%	86.55%
	Xception	77.94%	80.95%	78.98%	96.11%	92.30%	84.60%
	DenseNet201	78.74%	77.94%	78.16%	95.75%	92.77%	85.54%

The ISIC2018+(augmented) dataset proved to be beneficial in improving the performance of InceptionV3 but did not yield similar enhancements for the other two networks. Subsequently, the three best-performing models from the first dataset, along with the top-per-

forming model from the second dataset, were selected as feature extractors for the ensemble model. Due to space constraints, results charts are provided only for the two best-performing models from the first dataset. Training accuracy and loss curves are shown in Fig. 5.



(a)



(b)

Fig. 5. The accuracy and loss curves for (a) Xception. (b) DenseNet201.

Confusion Matrix				
True Class	BCC	BKL	MEL	NV
	78	4	7	4
	5	173	25	14
	1	5	140	25
NV	8	21	82	798
	BCC	BKL	MEL	NV
Predicted Class				

Fig. 6. The confusion matrix for Xception

Confusion Matrix					
True Class	BCC	BKL	MEL	NV	
	81	2	6	4	
	8	149	26	34	
	1	8	112	50	
NV	11	7	39	852	
		BCC	BKL	MEL	NV
		Predicted Class			

Fig. 7. The confusion matrices for DenseNet201

Fig. 8 shows the ROC curves for the designated models.

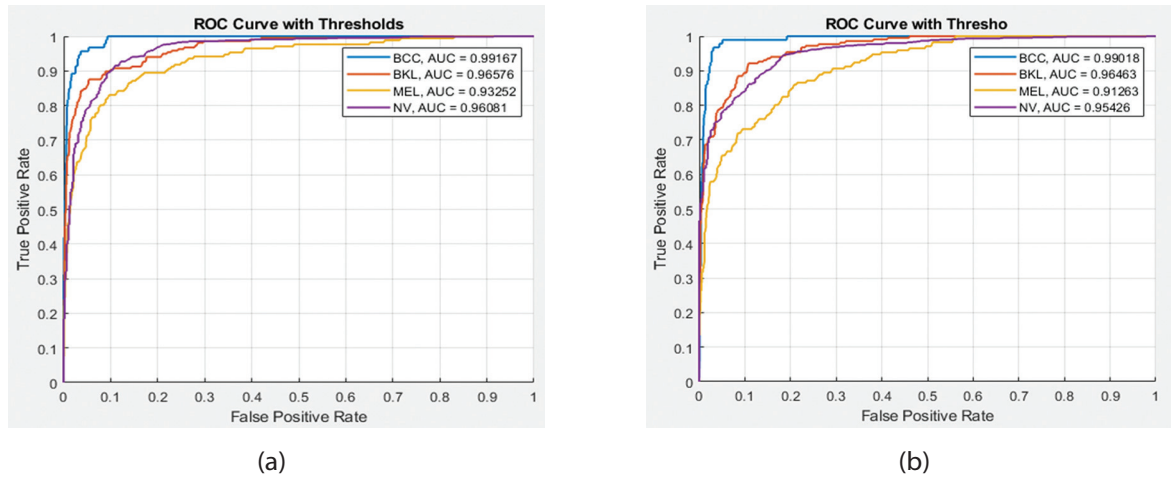


Fig. 8. The ROC curves for (a) Xception. (b) DenseNet201

SVM classifier is applied to the best three CNNs individually to study its impact to boost the Performance. And lastly, the ensemble model is restructured from the best models as previously illustrated in Fig. 4 to enhance the model further. Table 4 shows the evaluation metrics for the best-performing CNNs combined with SVM individually, along with the ensemble model that resulted from the combination of the four selected models. The ensemble model demonstrated improved performance by combining the strengths of the selected individual models into a single framework. Fig. 9 illustrates the confusion matrix of the ensemble model.

Table 4. The evaluation metrics for the best-performing CNNs combined with SVM classifier in comparison with the ensemble model

Model / [Dataset]	Precision	Recall	F1-Score	Mean Accuracy	Muti-Class Accuracy
InceptionV3 / [ISIC2018+ (augmented)]	81.72%	79.56%	80.33%	91.45%	86.50%
Xception / [ISIC2018+]	82.77%	82.12%	82.31%	93.89%	87.77%
DenseNet201 / [ISIC2018+]	79.79%	78.72%	78.91%	88.38%	85.76%
Ensemble	84.25%	83.01%	83.45%	94.46%	88.90%

Confusion Matrix				
True Class	BCC	BKL	MEL	NV
BCC	82	3	5	3
BKL	7	161	19	30
MEL	2	6	128	35
NV	8	9	27	865
Predicted Class				

Fig. 9. The confusion matrix of the ensemble model

3.2. COMPARISON WITH OTHER WORKS

The methodology employed in this research contributed to improving the classification of common skin cancer types compared to other researchers' models. Table 5 presents a comparison of the results with previous works. However, this comparison is not strictly one-to-one, as some of the previous studies did not utilize the test set provided by the ISIC archive. Additionally, while some researchers performed multi-class classification on two, seven or eight classes, while this research focuses on only four classes.

Table 5. Results comparison with other researchers

Reference	Dataset	Model	Accuracy
Alwakid <i>et al.</i> [20]	HAM10000	ResNet50	86%
Jain <i>et al.</i> [21]	HAM10000	Multiple CNNs, with Xception as the top-performing model	90.48%
Alam <i>et al.</i> [22]	HAM10000	S2C-DeLeNet	91.03%
Dogan and Ozdemir [14]	ISIC archive	Hybrid model of DenseNet201 with Random Forest.	91.28%
Natha <i>et al.</i> [15]	ISIC2018	Max Voting ensemble method includes Random Forest, (MLPN), and SVM	94.12%
Proposed method	ISIC2018 and ISIC2019	Proposed ensemble model	94.46%

4. CONCLUSION

The ensemble model demonstrated performance enhancements over the best individual CNNs by leveraging the strengths of multiple architectures. Applying image augmentation techniques to balance the dataset proved beneficial for the InceptionV3 model, but no significant improvement was observed in the other two CNNs, Xception and DenseNet201.

Despite the high performance of this framework in the skin cancer classification task, its computational complexity is a notable limitation. The proposed model was trained exclusively on dermoscopic images captured under controlled conditions, with high clarity and specific lighting provided by dermoscopic equipment. This limitation may affect its performance when applied to real-world data captured under varying conditions. Furthermore, the dataset lacks diversity in terms of skin tones and age groups, potentially introducing biases in predictions and reducing the generalizability of the model. Additionally, the exclusion of the AKIEC, DF, and VASC classes further reduces the model's applicability across the full spectrum of skin lesion types. Nevertheless, the proposed ensemble model holds promise for integration into clinical decision support systems. It can be deployed in modest computing environments within dermatology clinics to assist practitioners or be incorporated into teledermatology platforms, extending diagnostic support to patients in remote or underserved areas. In the future, expanding the dataset with a greater number of image samples categorized by ethnicity and skin tone could enable the development of a two-level model. The first level would classify images by ethnicity, and the second would perform skin lesion classification within each group. This approach has the potential to mitigate skin color biases and improve classification accuracy.

5. REFERENCES:

- [1] H. Sung et al. "Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries", *CA: A Cancer Journal for Clinicians*, Vol. 71, No. 3, 2021, pp. 209-249.
- [2] R. L. Siegel, K. D. Miller, A. Jemal, "Cancer statistics, 2019", *CA: A Cancer Journal for Clinicians*, Vol. 69, No. 1, 2019, pp. 7-34.
- [3] J. L. Bolognia, J. L. Jorizzo, J. V. Schaffer, "Dermatology", Elsevier Health Sciences, 2012.
- [4] A. C. Geller et al. "Melanoma epidemic: an analysis of six decades of data from the Connecticut Tumor Registry", *Journal of Clinical Oncology*, Vol. 31, No. 33, 2013, pp. 4172-4178.
- [5] H. Haenssle et al. "Reader study level-I and level-II Groups. Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists", *Ann Oncol*, Vol. 29, No. 8, 2018, pp. 1836-1842.
- [6] M. Vestergaard, P. Macaskill, P. Holt, S. Menzies, "Dermoscopy compared with naked eye examination for the diagnosis of primary melanoma: a meta-analysis of studies performed in a clinical setting", *British Journal of Dermatology*, Vol. 159, No. 3, 2008, pp. 669-676.
- [7] T. J. Hieken, R. Hernández-Irizarry, J. M. Boll, J. E. J. Coleman, "Accuracy of diagnostic biopsy for cutaneous melanoma: implications for surgical oncologists", *International Journal of Surgical Oncology*, Vol. 2013, No. 1, 2013, p. 196493.
- [8] L. F. di Ruffano et al. "Computer-assisted diagnosis techniques (dermoscopy and spectroscopy-based) for diagnosing skin cancer in adults", *Cochrane Database of Systematic Reviews*, Vol. 2018, No. 12, 1996.
- [9] Y. LeCun, Y. Bengio, G. Hinton, "Deep learning", *Nature*, Vol. 521, No. 7553, 2015, pp. 436-444.
- [10] I. Goodfellow, "Deep learning", MIT Press, 2016.
- [11] R. Brito et al. "GPU-enabled back-propagation artificial neural network for digit recognition in parallel", *The Journal of Supercomputing*, Vol. 72, 2016, pp. 3868-3886.
- [12] E. Bazgir, E. Haque, M. Maniruzzaman, R. Hoque, "Skin cancer classification using Inception Network", *World Journal of Advanced Research and Reviews*, Vol. 21, No. 02, 2024, pp. 839-849.
- [13] Y. Dahdouh, A. A. Boudhir, M. B. Ahmed, "A new approach using deep learning and reinforcement learning in healthcare: skin cancer classification", *International Journal of Electrical and Computer Engineering Systems*, Vol. 14, No. 5, 2023, pp. 557-564.
- [14] Y. Doğan, C. Özdemir, "Enhancing Skin Cancer Diagnosis through the Integration of Deep Learning and Machine Learning Approaches", *Bilişim Teknolojileri Dergisi*, Vol. 17, No. 4, 2024, pp. 339-347.
- [15] P. Natha, S. P. Tera, R. Chinthaginjala, S. O. Rab, C. V. Narasimhulu, T. H. Kim, "Boosting skin cancer diagnosis accuracy with ensemble approach", *Scientific Reports*, Vol. 15, No. 1, 2025, p. 1290.
- [16] International Skin Imaging Collaboration (ISIC), <https://www.isic-archive.com> (accessed: 2024)
- [17] M. Naqvi, S. Q. Gilani, T. Syed, O. Marques, H.-C. Kim, "Skin cancer detection using deep learning—a review", *Diagnostics*, Vol. 13, No. 11, 2023, p. 1911.

- [18] Z. G. Hadi, A. R. Ajel, A. Q. Al-Dujaili, "Comparison Between Convolutional Neural Network CNN and SVM in Skin Cancer Images Recognition", *Journal of Techniques*, Vol. 3, No. 4, 2021, pp. 15-22.
- [19] C. M. Bishop, N. M. Nasrabadi, "Pattern Recognition and Machine Learning", Springer, 2006.
- [20] G. Alwakid, W. Gouda, M. Humayun, N. U. Sama, "Melanoma detection using deep learning-based classifications", *Healthcare*, Vol. 10, No. 12, 2022, p. 2481.
- [21] S. Jain, U. Singhania, B. Tripathy, E. A. Nasr, M. K. Aboudaif, A. K. Kamrani, "Deep learning-based transfer learning for classification of skin cancer", *Sensors*, Vol. 21, No. 23, 2021, p. 8142.
- [22] M. J. Alam et al. "S2C-DeLeNet: A parameter transfer based segmentation-classification integration for detecting skin cancer lesions from dermoscopic images", *Computers in Biology and Medicine*, Vol. 150, 2022, p. 106148.

INTERNATIONAL JOURNAL OF ELECTRICAL AND COMPUTER ENGINEERING SYSTEMS

Published by Faculty of Electrical Engineering, Computer Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia.

About this Journal

The International Journal of Electrical and Computer Engineering Systems publishes original research in the form of full papers, case studies, reviews and surveys. It covers theory and application of electrical and computer engineering, synergy of computer systems and computational methods with electrical and electronic systems, as well as interdisciplinary research.

Topics of interest include, but are not limited to:

- Power systems
- Renewable electricity production
- Power electronics
- Electrical drives
- Industrial electronics
- Communication systems
- Advanced modulation techniques
- RFID devices and systems
- Signal and data processing
- Image processing
- Multimedia systems
- Microelectronics
- Instrumentation and measurement
- Control systems
- Robotics
- Modeling and simulation
- Modern computer architectures
- Computer networks
- Embedded systems
- High-performance computing
- Parallel and distributed computer systems
- Human-computer systems
- Intelligent systems
- Multi-agent and holonic systems
- Real-time systems
- Software engineering
- Internet and web applications and systems
- Applications of computer systems in engineering and related disciplines
- Mathematical models of engineering systems
- Engineering management
- Engineering education

Paper Submission

Authors are invited to submit original, unpublished research papers that are not being considered by another journal or any other publisher. Manuscripts must be submitted in doc, docx, rtf or pdf format, and limited to 30 one-column double-spaced pages. All figures and tables must be cited and placed in the body of the paper. Provide contact information of all authors and designate the corresponding author who should submit the manuscript to <https://ijeces.ferit.hr>. The corresponding author is responsible for ensuring that the article's publication has been approved by all coauthors and by the institutions of the authors if required. All enquiries concerning the publication of accepted papers should be sent to ijeces@ferit.hr.

The following information should be included in the submission:

- paper title;
- full name of each author;
- full institutional mailing addresses;
- e-mail addresses of each author;
- abstract (should be self-contained and not exceed 150 words). Introduction should have no subheadings;
- manuscript should contain one to five alphabetically ordered keywords;
- all abbreviations used in the manuscript should be explained by first appearance;
- all acknowledgments should be included at the end of the paper;
- authors are responsible for ensuring that the information in each reference is complete and accurate. All references must be numbered consecutively and citations of references in text should be identified using numbers in square brackets. All references should be cited within the text;
- each figure should be integrated in the text and cited in a consecutive order. Upon acceptance of the paper, each figure should be of high quality in one of the following formats: EPS, WMF, BMP and TIFF;
- corrected proofs must be returned to the publisher within 7 days of receipt.

Peer Review

All manuscripts are subject to peer review and must meet academic standards. Submissions will be first considered by an editor-

in-chief and if not rejected right away, then they will be reviewed by anonymous reviewers. The submitting author will be asked to provide the names of 5 proposed reviewers including their e-mail addresses. The proposed reviewers should be in the research field of the manuscript. They should not be affiliated to the same institution of the manuscript author(s) and should not have had any collaboration with any of the authors during the last 3 years.

Author Benefits

The corresponding author will be provided with a .pdf file of the article or alternatively one hardcopy of the journal free of charge.

Units of Measurement

Units of measurement should be presented simply and concisely using System International (SI) units.

Bibliographic Information

Commenced in 2010.

ISSN: 1847-6996

e-ISSN: 1847-7003

Published: semiannually

Copyright

Authors of the International Journal of Electrical and Computer Engineering Systems must transfer copyright to the publisher in written form.

Subscription Information

The annual subscription rate is 50€ for individuals, 25€ for students and 150€ for libraries.

Postal Address

Faculty of Electrical Engineering,
Computer Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia
Kneza Trpimira 2b
31000 Osijek, Croatia

IJECES Copyright Transfer Form

(Please, read this carefully)

This form is intended for all accepted material submitted to the IJECES journal and must accompany any such material before publication.

TITLE OF ARTICLE (hereinafter referred to as "the Work"):

COMPLETE LIST OF AUTHORS:

The undersigned hereby assigns to the IJECES all rights under copyright that may exist in and to the above Work, and any revised or expanded works submitted to the IJECES by the undersigned based on the Work. The undersigned hereby warrants that the Work is original and that he/she is the author of the complete Work and all incorporated parts of the Work. Otherwise he/she warrants that necessary permissions have been obtained for those parts of works originating from other authors or publishers.

Authors retain all proprietary rights in any process or procedure described in the Work. Authors may reproduce or authorize others to reproduce the Work or derivative works for the author's personal use or for company use, provided that the source and the IJECES copyright notice are indicated, the copies are not used in any way that implies IJECES endorsement of a product or service of any author, and the copies themselves are not offered for sale. In the case of a Work performed under a special government contract or grant, the IJECES recognizes that the government has royalty-free permission to reproduce all or portions of the Work, and to authorize others to do so, for official government purposes only, if the contract/grant so requires. For all uses not covered previously, authors must ask for permission from the IJECES to reproduce or authorize the reproduction of the Work or material extracted from the Work. Although authors are permitted to re-use all or portions of the Work in other works, this excludes granting third-party requests for reprinting, republishing, or other types of re-use. The IJECES must handle all such third-party requests. The IJECES distributes its publication by various means and media. It also abstracts and may translate its publications, and articles contained therein, for inclusion in various collections, databases and other publications. The IJECES publisher requires that the consent of the first-named author be sought as a condition to granting reprint or republication rights to others or for permitting use of a Work for promotion or marketing purposes. If you are employed and prepared the Work on a subject within the scope of your employment, the copyright in the Work belongs to your employer as a work-for-hire. In that case, the IJECES publisher assumes that when you sign this Form, you are authorized to do so by your employer and that your employer has consented to the transfer of copyright, to the representation and warranty of publication rights, and to all other terms and conditions of this Form. If such authorization and consent has not been given to you, an authorized representative of your employer should sign this Form as the Author.

Authors of IJECES journal articles and other material must ensure that their Work meets originality, authorship, author responsibilities and author misconduct requirements. It is the responsibility of the authors, not the IJECES publisher, to determine whether disclosure of their material requires the prior consent of other parties and, if so, to obtain it.

- The undersigned represents that he/she has the authority to make and execute this assignment.
- For jointly authored Works, all joint authors should sign, or one of the authors should sign as authorized agent for the others.
- The undersigned agrees to indemnify and hold harmless the IJECES publisher from any damage or expense that may arise in the event of a breach of any of the warranties set forth above.

Author/Authorized Agent

Date

CONTACT

International Journal of Electrical and Computer Engineering Systems (IJECES)
Faculty of Electrical Engineering, Computer Science and Information Technology Osijek
Josip Juraj Strossmayer University of Osijek
Kneza Trpimira 2b
31000 Osijek, Croatia
Phone: +38531224600,
Fax: +38531224605,
e-mail: ijeces@ferit.hr