

ISJET

**INTERNATIONAL SCIENTIFIC
JOURNAL OF ENGINEERING AND TECHNOLOGY**

Volume 10 No. 1 January-June 2026



ISSN 2586-8527 (Online)

Panyapiwat Institute of Management

Indexed in the Thai-Journal Citation Index (TCI 1)

**INTERNATIONAL SCIENTIFIC
JOURNAL OF ENGINEERING AND TECHNOLOGY
(ISJET)**

Volume 10 No. 1 January-June 2026

**ISSN 2586-8527 (Online)
PANYAPIWAT INSTITUTE OF MANAGEMENT**

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET)

Volume 10 No. 1 January-June 2026

ISSN 2586-8527 (Online)

Copyright

Panyapiwat Institute of Management

85/1 Moo 2, Chaengwattana Rd.,

Bang Talat, Pakkred,

Nonthaburi, 11120, Thailand

Tel. +66 2855 1560

E-mail: isjet@pim.ac.th

Website: <https://ph02.tci-thaijo.org/index.php/isjet/index>

Copyright©2017, Panyapiwat Institute of Management

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET)

Volume 10 No. 1 January-June 2026 ISSN 2586-8527 (Online)

THE INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET), with online ISSN 2586-8527, has been published since 2017. It is currently a journal accredited by the Thai-Journal Citation Index Centre (TCI), classified in Tier 1 for the field of Science and Technology.

Aim and Scope:

The purpose of publishing and disseminating academic articles in the fields of Engineering, Logistics, Agricultural Science, Food Science, and Other Areas of Science and Technology. Fields for academics, researchers, instructors, students, and the general.

The ISJET submissions are welcome from both the Panyapiwat Institute of Management (PIM) and other institutions in Thailand and abroad.

Peer Review Process

Reviewing Policy:

1. Any manuscript to be accepted for publication must have been reviewed and approved by at least two peer reviewers in that particular field or related fields. The Journal has a double-blind peer review policy, which means that neither the peer reviewer nor the author knows the identity of the other.
2. The submitted manuscript must have never been published in any other periodical, and must not be in the approval process for publication by any other periodical. Also, the author must not plagiarize others' work.
3. The article, expression, illustrations, and tables that are published in the Journal are the sole responsibility of the author and not that of the Panyapiwat Institute of Management.
4. The Editorial Board of the ISJET reserves the right to change or revise the name(s) and unit(s) of the author(s) in all cases after the issuance of the letter.
5. The Editorial Board of the ISJET reserves the right to cancel the publication that has been issued a certification of publication in the Panyapiwat Journal.
6. The Editorial Board of the ISJET reserves the right to decide on publishing an article in the Journal.

Type of Articles:

Research article: Original full-length research papers that have not been published previously. These are articles where authors report on studies they conducted. The length should exceed 15 pages, excluding references.

Academic articles: These are critical evaluations of studies in particular areas that have already been published, or theoretical articles in which the authors aim to formulate new theories on existing research. The length should exceed 15 pages, excluding references.

Review articles: Will be accepted in areas of topical interest and will normally focus on literature published over the previous five years. A review article typically includes: Abstract, Introduction, Main Body, Discussion, Conclusion, and References. The length should exceed 15 pages, excluding references.

Language:

The ISJET accepts manuscripts in English. All submissions must be well-written and adhere to correct, standard, and academically appropriate language. For English manuscripts, proofreading by a native speaker is required to ensure clarity and quality.

To maintain the Journal's high standards and publication ethics, the editorial board has implemented the Turnitin Program for all submitted manuscripts to address plagiarism issues.

Frequency of Publication: 2 Issues/ Year

The first issue: January - June

The second issue: July – December

The number of article publishes is 10 per issue.

Publication and Access Charges:

Authors are required to pay a publication fee of 4,000 Baht per article via direct transfer to the Panyapiwat Institute of Management's account. Payment must be made upon notification of the editorial board's successful preliminary review.

Workflow of Journal:

https://drive.google.com/file/d/1bUeIYuL7nOsAfviUB0h3AtEcNHKtuQRg/view?usp=drive_link

Open Access Policy:

All articles published in ISJET are open access and freely available for reading, downloading, and sharing. However, proper citations of these articles are required for their reproduction and public distribution by any medium.

Ownership and Management:

The ISJET is published by the Research and Development Department at Panyapiwat Institute of Management, Thailand, with financial support provided by the Institute. The journal's operation is led by the Editor-in-Chief and the Associate Editor-in-Chief, who work closely with the editorial team and staff. This funding ensures that the journal maintains high standards of publishing and a thorough peer-review process, and remains freely accessible to researchers globally. The INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET) is dedicated to fostering innovation and scientific advancement, and its support of this journal helps promote the exchange of knowledge and encourages collaborations both within Thailand and internationally.

Publisher:

The Office of Research and Development, Panaypiwat Institute of Management (PIM), is responsible for preparing and publishing the INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET).

The Address of the Institute is:

85/1 Moo 2, Chaengwattana RD., Bang Talat, Pakkred, Nonthaburi 11120, Thailand
Email: isjet@pim.ac.th Tel. +66 855 1560

Copyright and Licensing:

Prospective authors should be aware that all submissions must be original and be accompanied by the consent form listing all named authors. Additionally, authors must obtain official permission to include texts, figures, data, or any other information attributable to others.

Upon submission, the authors grant ISJET the right to publish the article without restrictions. However, the authors retain the copyright to their work and the right to reproduce, distribute, and display it. Authors may use the articles in their academic work or share them with their students, while the ISJET Journal controls how they are disseminated.

Revenue Sources:

The entire publication process is fee-free. The ISJET relies solely on the support of Panyapiwat Institute of Management (PIM) to generate revenue and encourage scholars and researchers to publish their valuable work.

Advertising:

In alignment with its policy, the journal does not accept any form of advertisement funding. Relevant information about academic conferences, journal updates, or research projects may be shared on the journal's official website when deemed appropriate. The inclusion of such content does not compromise the journal's ethical principles.

Direct Marketing:

The journal does not engage in direct marketing practices. Announcements, such as calls for papers or journal promotions, are made available through official channels, including the journal's website, social media, or affiliated organizations. These communications adhere to ethical guidelines and respect individual privacy.

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET)

Volume 10 No. 1 January-June 2026

ISSN 2586-8527 (Online)

Advisors Board

Assoc. Prof. Dr. Somrote Komolavanij
Assoc. Prof. Dr. Voravit Siripholvat
Prof. Dr. Jakrapong Kaewkhao
Assoc. Prof. Dr. Paritud Bhandhubanyong
Assoc. Prof. Dr. Pisit Charnkeitkong

Panyapiwat Institute of Management, Thailand
Panyapiwat Institute of Management, Thailand
Nakhon Pathom Rajabhat University, Thailand
Independent Scholar, Thailand
Independent Scholar, Thailand

Editor-in-chief

Assoc. Prof. Dr. Parinya Sanguansat

Panyapiwat Institute of Management, Thailand

Associate Editors

Asst. Prof. Dr. Anupong Thuengnaitham
Asst. Prof. Dr. Nivet Chirawichitchai
Asst. Prof. Dr. Phannachet Na Lamphun
Dr. Wirin Sonsrettee

Panyapiwat Institute of Management, Thailand
Panyapiwat Institute of Management, Thailand
Panyapiwat Institute of Management, Thailand
Panyapiwat Institute of Management, Thailand

Editorial Board

Prof. Dr. Anand Marya
Prof. Dr. Arnulfo P. Azcarraga
Prof. Dr. Bob Rastall
Prof. Dr. Chidchanok Lursinsap
Prof. Dr. Kyoko Kato
Prof. Dr. Feng Wang
Prof. Dr. Methi Wecharatana
Prof. Dr. Nigel Scollan
Prof. Dr. Ninshu Ma
Prof. Dr. Panich Intra
Prof. Dr. Parames Chutima
Prof. Dr. Phadungsak Rattanadecho
Prof. Dr. Prabhas Chongstitvatana
Prof. Dr. Prasanta Kumar Dey
Prof. Dr. Rattikorn Hewett
Prof. Dr. Rosemary R. Seva
Prof. Dr. Sandhya Babel

Prof. Dr. Saraju Mohanty
Prof. Dr. Stephen Beck
Prof. Dr. Takashi Yukawa
Prof. Dr. Thanaruk Theeramunkong

Prof. Dr. Venkata Seshachala Sarma Yadavalli
Prof. Dr. Vira Chankong
Prof. Dr. Yap Bee Wah
Prof. Chen Wei Ning, William
Prof. Duane P. Bartholomew
Prof. Jun Li
Prof. Yizhen Yang
Assoc. Prof. Bo Jing
Assoc. Prof. Dr. Chawalit Jeenanunta

Assoc. Prof. Dr. Ivan Henderson V. Gue
Assoc. Prof. Dr. Nattapon Chantarapanich
Assoc. Prof. Dr. Wilaiporn Lee
Asst. Prof. Dr. Adisak Joomwong
Asst. Prof. Dr. Rangsimma Chanphana
Dr. Jochen Hermann Josef Amrehn

University of Puthisastra, Cambodia
De La Salle University, Philippines
University of Reading, UK
Chulalongkorn University, Thailand
Aoyama Gakuin University, Japan
University of California, Berkeley, USA
New Jersey Institute of Technology, USA
Queen's University Belfast, UK
University of Osaka, Japan
Rajamangala University of Technology Lanna, Thailand
Chulalongkorn University, Thailand
Thammasat University, Thailand
Chulalongkorn University, Thailand
Aston Business School, Aston University, UK
Texas Tech University, USA
De La Salle University, Philippines
Sirindhorn International Institute of Technology,
Thammasat University, Thailand
University of North Texas, USA
University of Sheffield, UK
Nagaoka University of Technology, Japan
Sirindhorn International Institute of Technology,
Thammasat University, Thailand
University of Pretoria, South Africa
Case Western Reserve University, USA
Institute of Big Data Analytics and Artificial Intelligence, Malaysia
Nanyang Technological University, Singapore
University of Hawaii at Manoa, Honolulu, USA
Curtin University, Australia
Nanjing Tech University Pujiang Institute, China
Nanjing Audit University, China
Sirindhorn International Institute of Technology,
Thammasat University, Thailand
De La Salle University, Philippines
Kasetsart University, Sriracha Campus, Thailand
King Mongkut's University of Technology North Bangkok, Thailand
Maejo University, Thailand
Chulalongkorn University, Thailand
Nanjing Tech University Pujiang Institute, China

Dr. Muhammad Akhsin Muflikhun
Dr. Risdy Risdiyono
Dr. Viet Hoang

Gadjah Mada University, Indonesia
Islamic University of Indonesia, Indonesia
Van Lang University, Vietnam

Journal Secretary
Ms. Suchinda Chaluai

Panyapiwat Institute of Management, Thailand

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET)

Volume 10 No. 1 January-June 2026

ISSN 2586-8527 (Online)

Peer Reviewers

Prof. Dr. Parames Chutima
Assoc. Prof. Dr. Chana Raksiri
Assoc. Prof. Dr. Chanboon Sathitwiriawong

Assoc. Prof. Dr. Nattapon Chantarapanich
Assoc. Prof. Dr. Nivit Charoenchai
Assoc. Prof. Dr. Panomkhawn Riyamongkol
Assoc. Prof. Dr. Papot Jaroenapibal
Assoc. Prof. Dr. Siriporn Supratid
Assoc. Prof. Dr. Sunan Pansakorn

Assoc. Prof. Dr. Werasak Kurutach
Assoc. Prof. Dr. Wilaiporn Lee

Assoc. Prof. Chaiyan Junsiri
Asst. Prof. Dr. Adisak Joomwong
Asst. Prof. Dr. Chutima Beokhaimook
Asst. Prof. Dr. Korawit Chaisu
Asst. Prof. Dr. Narongdech Keeratipranon
Asst. Prof. Dr. Satien Janpla
Asst. Prof. Dr. Sukumal Kitisin
Asst. Prof. Dr. Sumeth Yuenyong
Asst. Prof. Dr. Wongkot Sriurai
Dr. Sanparith Marukatat

Chulalongkorn University, Thailand
Kasetsart University, Thailand
King Mongkut's Institute of Technology Ladkrabang, Thailand
Kasetsart University Sriracha Campus, Thailand
Chiang Mai University, Thailand
Naresuan University, Thailand
Khon Kaen University, Thailand
Rangsit University, Thailand
Rajamangala University of Technology Thanyaburi, Thailand
Mahanakorn University of Technology, Thailand
King Mongkut's University of Technology North Bangkok, Thailand
Khon Khen University, Thailand
Maejo University, Thailand
Rangsit University, Thailand
Chiang Mai University, Thailand
Chulalongkorn University, Thailand
Suan Sunandha Rajabhat University, Thailand
Kasetsart University, Thailand
Mahidol University, Thailand
Ubon Ratchathani University, Thailand
National Electronics and Computer Technology center, Thailand

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET)

Volume 10 No. 1 January-June 2026

ISSN 2586-8527 (Online)

Dear Colleagues,

Welcome to the tenth volume of the International Scientific Journal of Engineering and Technology (ISJET). This milestone issue arrives at a dynamic period where physical environments, digital infrastructure, and human psychology increasingly intersect. The ten exceptional papers featured herein reflect our ongoing commitment to harnessing advanced computational methods to solve complex, real-world problems.

Modern logistics and data management demand robust optimization. To this end, our authors present a hybrid optimization-simulation approach for supply chain network design under uncertainty and in highly perishable environments. Complementing this operational focus, another study addresses cost minimization in 3D bin packing under orientation and operational constraints, providing a timely case study in e-commerce fulfillment. Pushing the boundaries of edge computing, researchers also explore an Artificial Bee Colony Algorithm using cosine similarity on a Raspberry Pi cluster to enhance medical data classification.

In the realm of autonomous mobility and computer vision, a critical study evaluates the robustness of road-tracking systems for autonomous driving to environmental disruptions. Similarly, a comprehensive review of the Donkey Car autonomous driving platform, which uses a neural network, provides vital insights for accessible, small-scale deployments. Bridging computer vision and generative AI, a detailed paper on face photo-to-sketch synthesis using CycleGAN breaks down the crucial roles of residual blocks and spectral normalization.

Technological interventions in precision agriculture are highlighted by an automated, IoT-based mushroom greenhouse control system that enables real-time environmental management via the Blynk application. Further exploring agricultural efficiency, a specialized agronomic study details the effect of intermittent photoperiod on yield and light-use efficiency of the common ice plant (*Mesembryanthemum crystallinum* L.) under controlled-environment conditions.

Finally, we explore AI's computational and sociological dimensions. Addressing natural language processing, a novel hybrid of syntactic structure and contextual semantics is introduced to enhance product review sentiment analysis. Moving to the human-AI interface, a thought-provoking piece examines AI as a form of escapism, analyzing the psychological dimensions of modern user interactions.

We extend our deepest gratitude to our peer reviewers, the editorial board, and our authors. We hope this volume inspires continued cross-disciplinary collaboration within the global engineering and technology community.

With kind regards,

Assoc. Prof. Dr. Parinya Sanguansat
Editor-in-chief
isjet@pim.ac.th

CONTENTS

• Artificial Bee Colony Algorithm with Cosine Similarity on Raspberry Pi Cluster for Enhancing Medical Data Classification <i>Anan Banharnsakun</i>	1
• A Hybrid Optimization-Simulation Approach for Supply Chain Network Design under Uncertainty and Highly Perishable Environments <i>Jirawat Chatchaichalermporn and Navee Chiadamrong</i>	12
• A Study of Road Tracking Immunity for Autonomous Driving <i>Chuanxiang Bi and Jian Qu</i>	27
• A Review on DonkeyCar Autonomous Driving Platform Using Neural Network <i>Yangyang Li and Jian Qu</i>	35
• An Approach of AI as a Practice of Escapism <i>Waralak Vongdoiwang Siricharoen</i>	42
• Automated IoT-Based Mushroom Greenhouse Control System for Real-Time Environmental Management via Blynk Application <i>Chainarong Janthoom, Suphalak Batpho, Sittisak Thongsuk, and Maytagorn Thongkaobua</i>	50
• Cost Minimization in 3D Bin Packing under Orientation and Operational Constraints: A Case Study in E-Commerce Fulfillment <i>Anupong Thuengnaitham and Salilathip Thippayakraisorn</i>	59
• Effect of Intermittent Photoperiod on Yield and Light-Use Efficiency of Common Ice Plant (<i>Mesembryanthemum crystallinum</i> L.) under Controlled-Environment Conditions <i>Kornlawat Tantivit, Voravit Siripholvat, and Nopparat Tatmala</i>	73
• Face Photo and Sketch Synthesis Using CycleGAN: The Role of Residual Blocks and Spectral Normalization <i>Nuntipat Phisutthangkoon, Patikorn Anchuen, Thiansiri Luangwilai, and Phummipat Daungklang</i>	84
• Hybrid of Syntactic Structure and Contextual Semantics for Enhanced Product Review Sentiment Analysis <i>Damrong Sattayawaksakul and Pimpa Cheewaparakobkit</i>	96

Artificial Bee Colony Algorithm with Cosine Similarity on Raspberry Pi Cluster for Enhancing Medical Data Classification

Anan Banharnsakun*

Computational Intelligence Research Laboratory, Computer Engineering Department,
Faculty of Engineering at Sriracha, Kasetsart University, Chonburi, Thailand
Email: anan@eng.src.ku.ac.th*

Received: June 9, 2025 / Revised: November 11, 2025 / Accepted: November 19, 2025

Abstract—Machine learning plays a very important role in our daily lives. Machine learning is used to solve difficult problems encountered in many fields of study. Especially in medical diagnosis, machine learning is used to support patient diagnosis. In this research, the artificial bee colony approach, a well-known bio-inspired algorithm in the field of machine learning, is proposed to enhance the classification of cancer diagnosis data by combining it with cosine similarity measurements and parallel computing on Raspberry Pi clusters. The performance of the proposed method is validated and compared with other well-known algorithms against the Breast Cancer Wisconsin dataset taken from the UCI machine learning repository. Experimental results indicate that the proposed method achieves superior classification accuracy compared to existing approaches, yielding improvements of up to 6.45%. Furthermore, it can scale effectively to support large-scale medical data for disease diagnosis.

IndexTerms—Artificial Bee Colony (ABC), Cosine Similarity, Classification, Parallel Computing, Raspberry Pi

I. INTRODUCTION

In the field of medical care, a large volume of data has been generated and collected, including patient information, symptoms, and treatment methods. These data need to be analyzed and managed effectively to support an accurate diagnosis [1]. However, data analysis is a complex and challenging task. Therefore, the classification of medical data, which is one of the important data analysis processes in healthcare, has attracted the attention of many researchers to present methods to help categorize data more efficiently [2].

Over the past decades, an increasing number of methods have been introduced to classify medical data. This has been done to support clinicians in

personalizing medical treatments and engaging in shared decision-making. Fuente-Tomas et al. [3] proposed using the *k*-means method for cluster-based classification to account for the varying degrees of severity of bipolar disorder. On the other hand, Nanglia et al. [4] presented a hybrid algorithm that combines Support Vector Machine (SVM) and neural networks to improve accuracy and reduce computational complexity in lung cancer classification. Hamed et al. [5] introduced a variant of the *k*-Nearest Neighbors (*k*-NN) algorithm for classifying COVID-19 data, aiming to improve patient survival rates through early detection. Alam et al. [6] proposed a feature-ranking-based methodology that leverages feature selection with the random forest algorithm to train and construct models for medical data classification.

While current algorithms for data classification are effective in improving accuracy and producing favorable outcomes, they are not without limitations [7]-[9]. These algorithms have various drawbacks, including sensitivity to initial values, incompatibility with high-dimensional datasets, susceptibility to overfitting, and extensive computational time. Despite these limitations, there remains a need for further advancements in developing an efficient and effective data classification method that can overcome these challenges.

Over the past two decades, the use of algorithms inspired by biology has demonstrated its effectiveness in addressing numerous challenges in science and engineering [10]-[13]. Various techniques utilize these biologically inspired algorithms to tackle problems in data classification [14]-[16]. However, as no single approach is sufficient to meet all the requirements of data classification, researchers are continually motivated to develop new algorithms that offer more efficient solutions and enhance classification precision.

The ABC algorithm, developed by Karaboga [17], is a widely used approach for finding optimal solutions to optimization problems [18]-[20]. This technique is based on the natural process of honeybees searching

for high-quality food. Prior research [21]-[23] has demonstrated the ABC method's efficacy and efficiency in resolving a diverse array of optimization problems, surpassing other methods. In this study, we employ the ABC approach incorporating cosine similarity (ABC-CS) to classify medical data. This study aims to demonstrate the utility and effectiveness of the ABC-CS method, a straightforward and efficient approach inspired by bee foraging, for classifying medical data. The significance of this work lies in its demonstration of the ABC-CS method's applicability to the medical domain.

However, the large volume of medical data necessitates parallel computing to efficiently process and analyze this wealth of information, enabling quicker and more accurate clinical decision-making and medical research [24]-[26]. The Raspberry Pi, originally designed as an educational tool, has evolved into a versatile platform for IoT applications due to its low cost, compact form factor, and energy efficiency [27]-[29]. In this work, we also present the Raspberry Pi combined with parallel computing techniques, specifically our proposed ABC-CS method, which enables processing and analysis of large medical datasets.

The paper is structured as follows: Section 2 provides an overview of the background and knowledge, including data classification with cosine similarity and the artificial bee colony algorithm. Section 3 proposes enhancing data classification using the ABC-CS method. Section 4 introduces the ABC-CS with parallel computing on Raspberry Pi clusters. Section 5 presents the experimental settings and results. Finally, Section 6 concludes the paper.

II. BACKGROUND AND KNOWLEDGE

A. Classification with Cosine Similarity

Data classification refers to the systematic arrangement of data into meaningful categories based on their characteristics. In this specific context, centroid-based data classification is used to categorize the data. This method involves calculating a centroid vector that represents the data of each class. To determine the class of new data, the cosine similarity function is used to compare it to each centroid vector. The new data is then assigned to the class with the most similar centroid vector, as measured by cosine similarity [30]. This approach is useful in identifying patterns and relationships within data [31], [32], making it easier to analyze and make informed decisions based on the data.

Let $x = (x_1, x_2, x_3, \dots, x_n)$ and $y = (y_1, y_2, y_3, \dots, y_n)$ be two vectors for comparison. Using the cosine measure as a similarity function, it can be defined as in (1):

$$\text{sim}(x, y) = \frac{x \cdot y}{\|x\| \|y\|} \quad (1)$$

where $\|x\|$ and $\|y\|$ are the Euclidean norm of vectors, defined as $\sqrt{x_1^2 + x_2^2 + x_3^2 + \dots + x_n^2}$ and $\sqrt{y_1^2 + y_2^2 + y_3^2 + \dots + y_n^2}$, respectively

The calculation of cosine similarity between two vectors, x and y , involves determining the cosine of the angle between them. A cosine similarity score of 0 indicates that the vectors are completely orthogonal, meaning they have no overlap or correlation. In contrast, a cosine similarity score of 1 means that the vectors are perfectly aligned, with no angle between them. The closer the cosine similarity score is to 1, the stronger the correlation and similarity between the vectors, as the angle between them is smaller.

In terms of using cosine similarity to classify the data, if x is the centroid vector of a class and y is a data vector, $\text{sim}(x, y)$ is used to determine whether the data vector y should belong in that class. When the data vector y is compared to the centroid vector of all classes, the data vector y will be assigned to the class that gives the most $\text{sim}(x, y)$ value.

B. ABC Method

In the last decade, the ABC approach has been the subject of a wide range of research and effective applications in the optimization domain. It is an optimization algorithm inspired by the intelligent foraging behavior of honey bee swarms. The bee agents in the ABC approach are categorized into three types. The first type is called employed bees. The next type is the onlooker bee, and the last type is the scout bee. The food source position represents a feasible solution in the optimization problem, and the food source quality measures solution quality, referred to as 'fitness'. The process of bees searching for a desirable food source through foraging is similar to identifying the optimal solution.

A randomly distributed initial food source position (solution) is generated in the first step of the ABC approach and assigned to each employed bee. Next, the employed bees search for new candidate food source positions around the old ones in their memory. When an employed bee discovers a new food source, it assesses the fitness of both the new and the previous food source positions. If the fitness of the new food source is greater than that of the old one, the bee will memorize the position of the new food source and forget the position of the old one. However, if the fitness of the old food source is still higher, the bee will retain the position of the old food source in its memory. The employed bees then share the information regarding their food sources with the onlooker bees. Each onlooker evaluates the fitness of the food source positions taken by all employed bees and then chooses a food source based on the fitness of each position. Similar to the employed bees, the onlooker bees then search for new candidate food

source positions around the old ones in their memory. An onlooker bee will memorize the position of a new food source if its fitness is higher than that of the previous one, and forget the old one. Any food source position discovered by employed or onlooker bees that does not show quality improvement will be discarded and replaced with a new position determined by a scout bee.

III. ENHANCEMENT OF DATA CLASSIFICATION BY USING ABC-CS

The proposed ABC-CS method aims to find optimal centroid values for data classification with improved efficiency. The centroid values, which serve as parameters for the data classes, are a crucial aspect of the classification process and need to be optimized. This is achieved using the ABC-CS method, which optimizes these values and improves the overall effectiveness of the data classification process.

The classification problem involves assigning N objects, represented as $O = \{o_1, o_2, o_3, \dots, o_N\}$, to K classes, represented as $C = \{c_1, c_2, c_3, \dots, c_K\}$. Objects and classes are characterized by D values that specify their positions within a multidimensional space. Each object, denoted as o_i , and each class centroid, denoted as c_j , is formally represented as a D -dimensional vector, where each component corresponds to a distinct dimension. The purpose of the classification process is to optimize the total

cosine similarity between each data point, also referred to as an “object,” and the central point, known as the centroid, of the class it is classified into. The aim is to maximize the sum of these cosine similarity scores to improve classification accuracy. This is achieved by adjusting the centroids’ positions to align as closely as possible with the objects they represent, thereby increasing the similarity between each object and its corresponding centroid. In this way, the classification process optimizes cosine similarity between objects and centroids, yielding more accurate and meaningful classifications. This is expressed as the objective function, which serves as the fitness function of ABC-CS and can be represented by (2). The objective is to allocate each object to one of the K classes in a way that maximizes the cosine similarity between the object and the centroid of its assigned class.

$$\text{Fitness} = \sum_i^N \sum_j^K w_{ij} \cdot \text{sim}(o_i, c_j) \quad (2)$$

where w_{ij} is the association weight between object o_i and class j . It is a value that indicates the degree of association between the object and the class. If i is allocated to class j , then w_{ij} equals 1. On the other hand, if o_i is not allocated to class j , then w_{ij} equals 0.

The ABC-CS algorithm for finding the optimal centroid values in the data classification process is described and shown in Fig. 1.

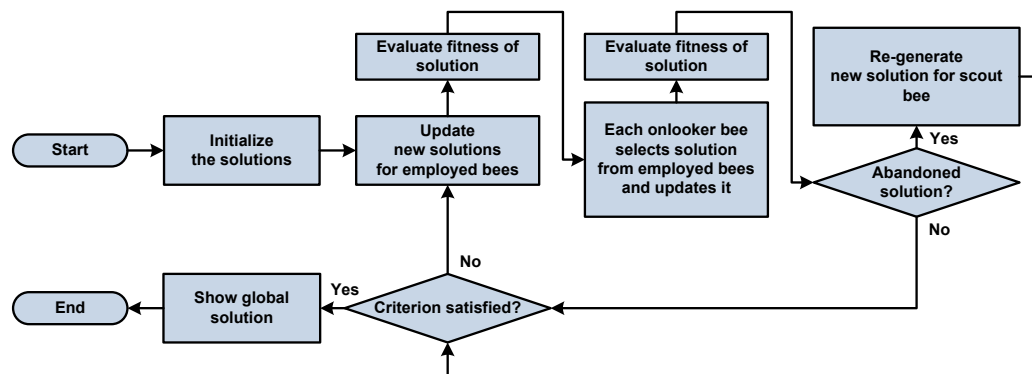


Fig. 1. Flowchart for finding optimal centroid values using the ABC-CS algorithm

As shown in Fig. 1, the artificial bee colony algorithm begins by randomly generating initial solutions (centroid values) that serve as the food source positions for the bee agents. These bee agents will then carry out three crucial functions. Firstly, the employed bees update the feasible food source positions by searching for new ones using (3) and comparing their own food source position with their neighbors’ previously selected food source positions. Secondly, the onlooker bees choose the feasible food source positions. Finally, the scout bees prevent further deterioration of the quality of the food sources.

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \quad (3)$$

where the solution v_{ij} is obtained by modifying its original value x_{ij} through a comparison with a randomly selected solution x_{kj} from its neighboring solutions. The alteration is performed using a random number ϕ_{ij} in the range $(-1, 1)$. This number is used to make a random adjustment to the previous solution, resulting in a new solution in the next iteration. The indices k and j are chosen randomly, with k , being any number from 1 to the number of food sources (SN), excluding i , and j being any number from 1 to D . The

difference between x_{ij} and x_{kj} represents a variation in the position of the solution in a specific dimension.

The employed bees in a hive are responsible for finding food sources and updating their memories with the positions of new candidate food sources. If the new food source has a better fitness value than the previous one, the memory of the employed bee will be updated accordingly. Once the employed bee has discovered a new food source, it returns to the hive and communicates its fitness value to the onlooker bees. In the second task, the onlooker bees use the information provided by the employed bees to select one of the food sources based on the fitness values. The selection process is determined by (4), which calculates the probability of a food source being chosen.

$$P_i = \frac{fit_i}{\sum_{i=1}^{SN} fit_i} \quad (4)$$

where fit_i is the fitness value of the food source i , which is calculated by using (2).

The likelihood that onlooker bees select a proposed food source is proportional to its fitness value, with higher fitness values leading to a higher probability of selection. Once the onlooker bees have selected a food source, they visit that location and determine whether a nearby site is a potential new food source. The calculation for this new candidate food source site can be done using (3).

The third task involves replacing any food source position that has not shown an improvement in its fitness value. This is done to prevent suboptimal solutions. The new position that replaces the abandoned one will be selected randomly by a scout bee, using (5). This random selection helps to ensure that the solution is not stuck at a suboptimal point and allows the algorithm to explore new potential solutions.

$$x_{ij} = x_j^{\min} + rand[0,1](x_j^{\max} - x_j^{\min}), \quad (5)$$

where $x_j^{\min}$ and $x_j^{\max}$ are the lower bound and the upper bound of the food source position in dimension j , respectively.

The algorithm will repeat the three major tasks described earlier until the number of iterations reaches a pre-determined value. The number of iterations is used as a termination criterion, meaning it determines when the algorithm will stop running. Once the number of iterations reaches the specified value, the algorithm will end, and the final solution will be presented.

IV. ABC-CS WITH PARALLEL COMPUTING ON RASPBERRY PI CLUSTERS

From the past to the present, Raspberry Pi has undergone continuous development, improving its performance and enabling it to handle a variety of processing tasks, much like a single computer. The advantage of Raspberry Pi is its low cost and low energy consumption. For this reason, using a Raspberry Pi to create clusters for parallel processing of large datasets is both interesting and challenging. In our research, we have implemented parallel processing on a cluster system built from Raspberry Pi, as shown in Fig. 2, and applied the ABC-CS algorithm we proposed to this cluster. This is to evaluate the proposed algorithm's performance in terms of parallel processing efficiency.

In our parallel approach, referred to as parallel ABC-CS, we employ the Manager-worker model [33] for overseeing process management. This model comprises a singular manager and multiple workers, each with distinct responsibilities. The manager primarily handles tasks such as initialization, load balancing, partitioning the bee colony, distributing bee subgroups to workers, and collecting and presenting global results.

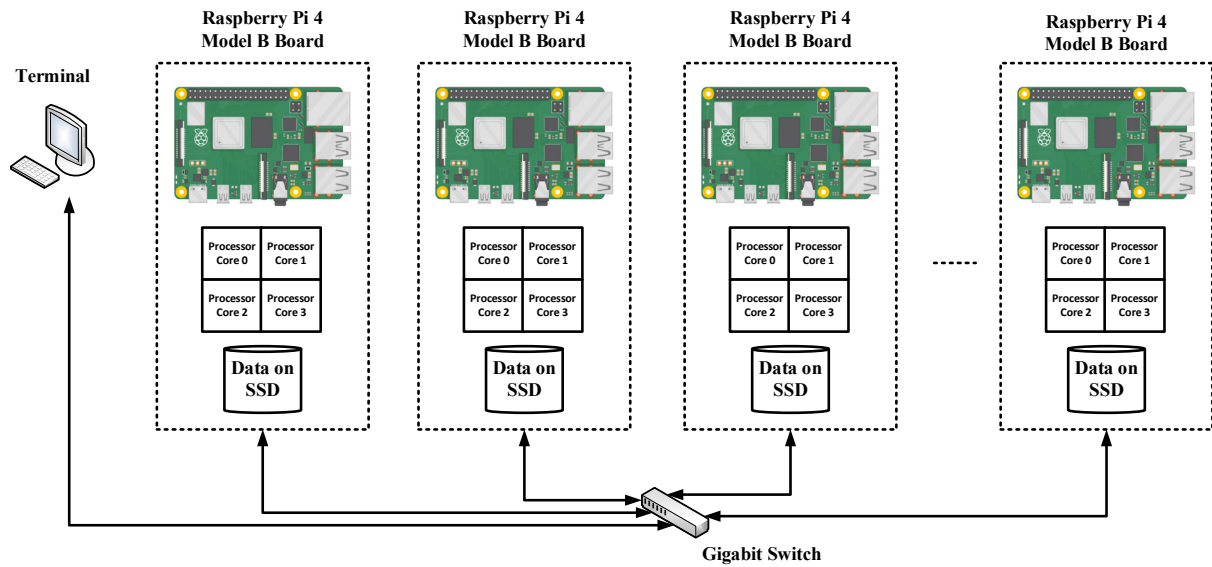


Fig. 2. Raspberry Pi 4 Cluster Topology in the proposed work

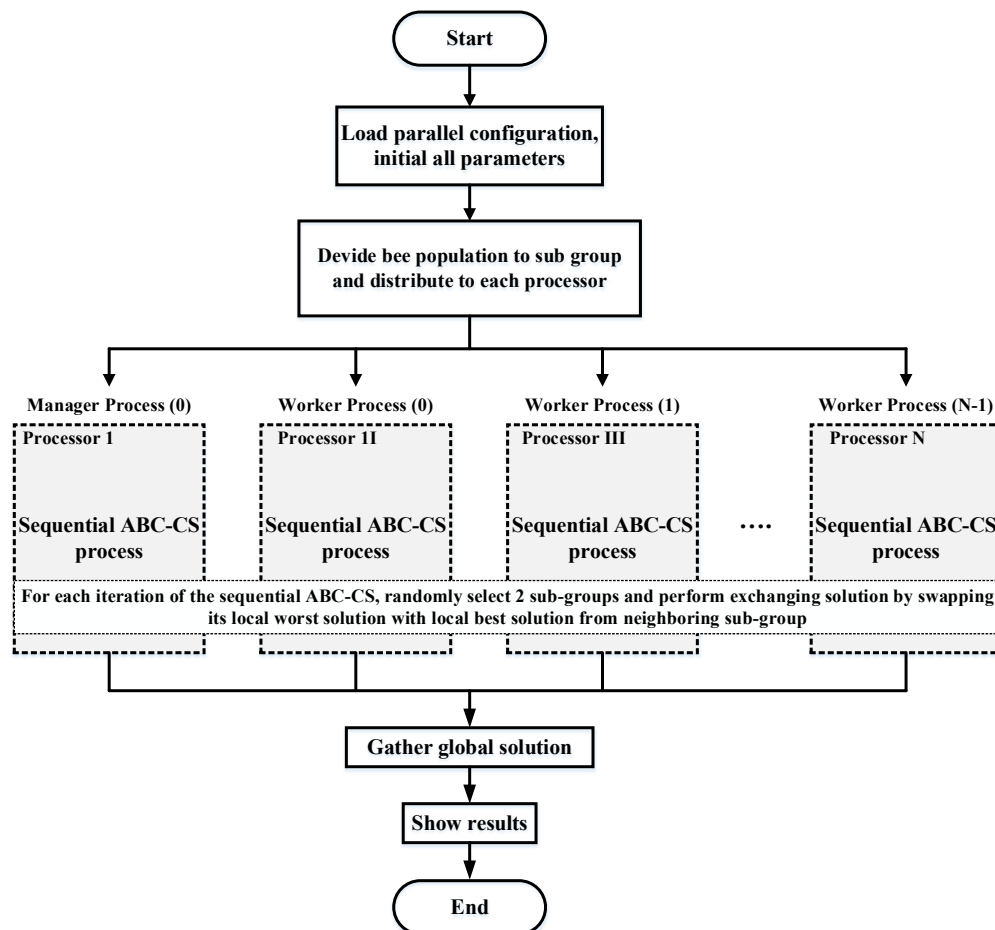


Fig. 3. The parallel ABC-CS flowchart

Meanwhile, the workers are responsible for executing the primary computations, including receiving bee subgroups from the manager, processing them, and returning the local best solution results at the final iteration.

The communication within our presented model occurs between the Manager and the Worker, as well as among the Workers themselves. This communication uses the Message-Passing Interface (MPI), a widely accepted standard communication framework. MPI provides a set of routines for message-passing operations, such as sending and receiving messages. The Manager and Workers employ these operations to exchange information and collaborate on their computations. Communicators, or process groups in MPI, define sets of processes that can communicate with one another. In the Worker-Manager Model, communicators help specify the group of processes involved in the computation, including both the Manager and Workers. Each process within the communicator is uniquely identified by an integer rank, ranging from 0 to the total number of processes minus 1. In the Worker-Manager Model, rank 0 is typically assigned to the Manager, and each Worker has a unique rank. MPI routines are integrated into the application code, facilitating seamless communication within the distributed system, including sending and receiving messages and other necessary message-passing operations.

In the presented work, each process loads a complete parallel ABC-CS code, as depicted in Fig. 3. The Manager's role involves dividing the entire bee swarm into subgroups and allocating these subgroups to individual Worker processes. Although coordinating computations is not the Manager's primary responsibility, the Manager actively participates in processing, similar to the Workers, in each round. Bees within each subgroup collaborate to improve their solutions. In the final step of each iteration, the Manager selects two random subgroups from the overall population. These two subgroups exchange their best solutions, replacing the worst solution in one group with the best solution from the other. This process emphasizes that data exchange occurs only at these synchronization points, thereby reducing communication time.

In the last round, the Manager consolidates the best solutions from all Worker processes at the local level and computes the globally optimal solution.

V. EXPERIMENTAL SETTINGS AND RESULTS

In this section, a comprehensive evaluation of the proposed method for improving the classification of medical data is presented. To assess the efficacy of the proposed method, several other widely-used approaches, such as the K-Nearest Neighbors (k-NN) [34], Support Vector Machine (SVM) [35], and the Artificial Bee Colony with K-Means (ABC-KMeans) [36], were utilized for comparison with the classification outcomes of the medical data. The experiment was performed using the Breast Cancer Wisconsin dataset obtained from the UCI Machine Learning Repository [37]. This dataset was selected as it provides a standardized benchmark for evaluating the performance of various classification algorithms. The results of the experiment aim to demonstrate the superiority of the proposed method in accuracy and efficiency compared with other well-known approaches. This dataset contains 699 instances, each with nine input attributes used to classify into two classes: Malignant or benign, as shown in Table I. The training set is randomly assigned from 75% of the data instances, and the testing set is the remaining 25%. Accuracy is used as the primary evaluation metric to compare the effectiveness of the proposed method with the previously mentioned methods.

All methods were implemented in C++ and tested on a Raspberry Pi 4 Model B with a Cortex-A72 (ARMv8) 64-bit core running at 1.8 GHz and 8 GB of memory. The number of iterations for each method was fixed at 200, and for the ABC methods, 20 bees were used as employed bees and onlooker bees. The radial basis function was used as the kernel in the SVM, with C and γ set to 1 and 0.1, respectively. In the k-NN method, K was set to 6. These parameter settings were determined to be optimal after a preliminary study was conducted as part of this work.

The Confusion matrix results for the proposed method, k-NN, SVM, and ABC-KMeans are presented in Table II, and clear visual representations of the comparison of classification accuracy, precision, recall, and F1 score for each class are provided in Fig. 4 through Fig. 7.

TABLE I
INFORMATION ON THE ATTRIBUTES OF THE BREAST CANCER WISCONSIN DATASET

No.	Attribute Description	Domain
1	Clump Thickness (CT): The clump thickness refers to the concentration or grouping of cancer cells that are stacked on top of each other. This measurement provides insight into tumor density and how cancer cells are growing and spreading.	1-10
2	Uniformity of Cell Size (UCS): The uniformity of cell size is used to assess the extent of metastasis, or the spread of cancer, to the lymph nodes. This metric can help determine whether cancer cells have spread to other parts of the body and are growing there.	1-10
3	Uniformity of Cell Shapes (UCSh): The uniformity of cell shapes is used to identify cancerous cells that vary in size. This metric helps to distinguish between normal cells and cancerous cells, which can have irregular shapes and sizes.	1-10
4	Marginal Adhesion (MA): Marginal adhesion refers to a loss of the ability of cancer cells to adhere to one another. The loss of this property is a hallmark of malignancy, as the cancer cells can become more invasive and spread to other parts of the body. Conversely, the retention of adhesion can indicate the presence of malignant cells.	1-10
5	Single Epithelial Cell Size (SECS): The size of single epithelial cells is another factor used to assess the presence of cancer. If the SECS become larger, it may indicate malignancy, as cancer cells can grow and divide faster than normal cells.	1-10
6	Bare Nuclei (BN): Bare nuclei refer to cells that lack a cytoplasmic coating. This is a common feature of benign tumors and helps to distinguish them from malignant tumors.	1-10
7	Bland Chromatin (BC): Bland chromatin is typically found in benign cells and is characterized by a uniform, light appearance. This contrasts with the denser, darker appearance of malignant cells.	1-10
8	Normal Nucleoli (NN): Normal nucleoli are usually very small in benign cells, whereas they can be larger and more prominent in cancer cells.	1-10
9	Mitoses: Mitoses refer to the process of cell division by which the nucleus of a cell splits into two new nuclei. This process is an important part of cell growth and development, and an increase in the number of mitoses can indicate a rapidly growing cancer.	1-10

The results in Table II indicate that the proposed method outperforms the other methods in terms of True Positive and True Negative values, and also has lower False Positive and False Negative values. Furthermore, Figs. 4 through 7 also suggest that the proposed method provides the best quantitative evaluation results. Fig. 8 provides a clear visual comparison of the classification accuracy results obtained from the various algorithms evaluated, including k-NN, SVM, and ABC-KMeans. The proposed method showed

an improvement in accuracy of 6.45%, 3.12%, and 4.43%, respectively. It is evident that the proposed method outperforms the other methods and provides the best results. Note that the distinction between ABC-CS and ABC-KMeans is that the fitness or similarity between data points in ABC-CS is measured using cosine similarity. In contrast, the fitness in ABC-KMeans is calculated by minimizing intra-cluster distances.

TABLE II
CONFUSION MATRIX FOR K-NN, SVM, ABC-KMEANS, AND ABC-CS CLASSIFIERS

Predicted/Actual	k-NN		SVM		ABC-KMeans		ABC-CS	
	Benign	Malignant	Benign	Malignant	Benign	Malignant	Benign	Malignant
Benign	102	5	102	6	104	6	105	4
Malignant	15	53	9	58	11	54	6	60

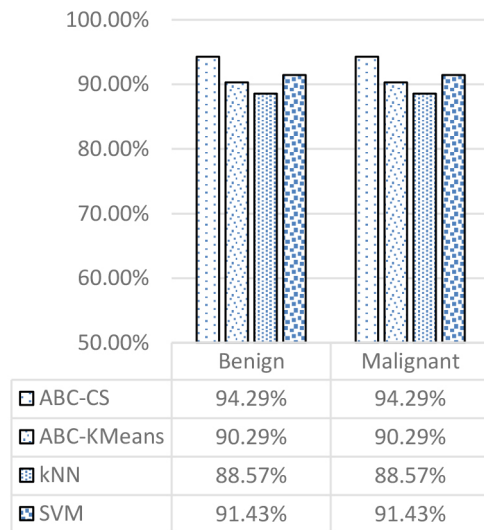


Fig. 4. The comparison of accuracy for each class

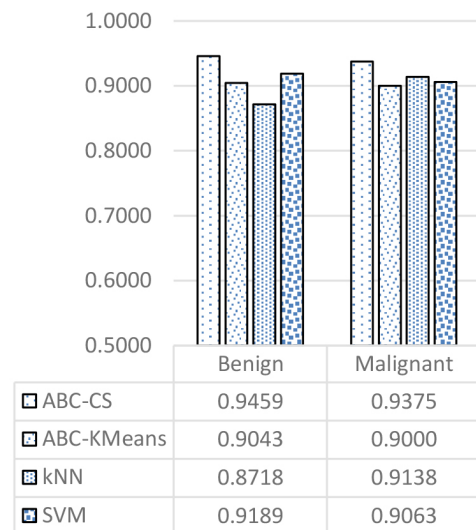


Fig. 6. The comparison of recall for each class

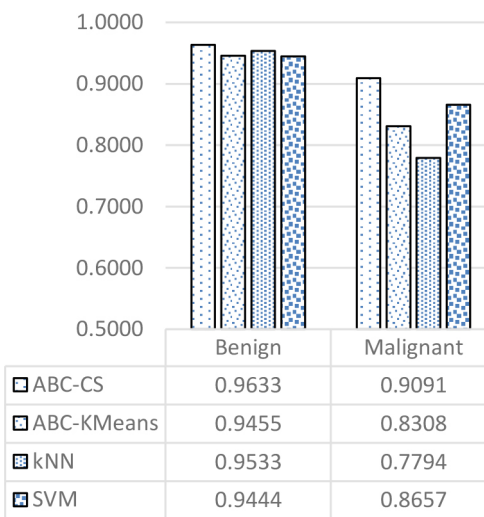


Fig. 5. The comparison of precision for each class

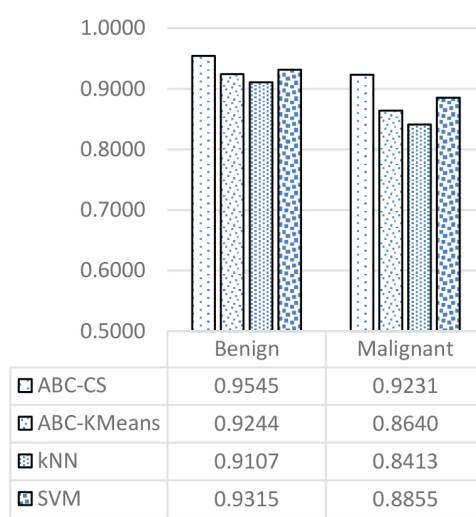


Fig. 7. The comparison of the F1-score for each class

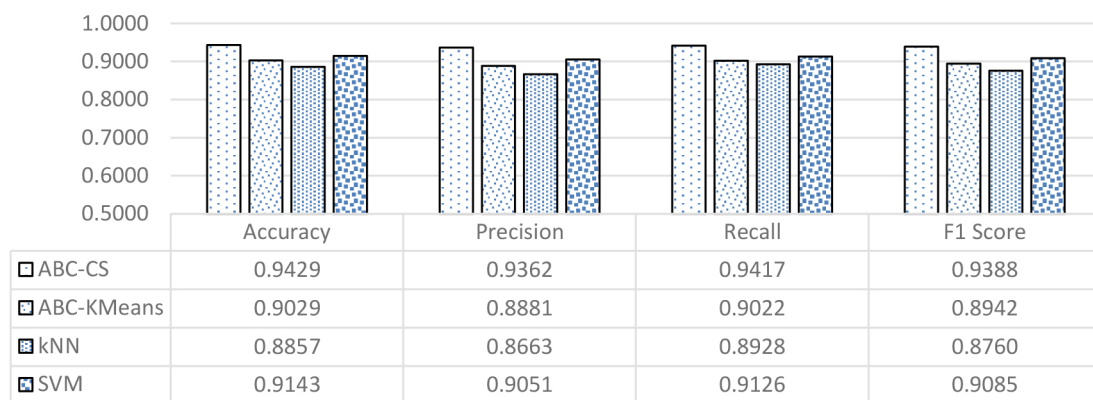


Fig. 8. The comparison of classification accuracy percentages, precision, recall, and F1 score of techniques

To evaluate the performance of our ABC-CS method on a large dataset, we used synthetic datasets by duplicating all records to produce datasets up to about 699,000 records. All methods in this experiment were programmed in C++, and all experiments were

run on a Raspberry Pi 4 Model B cluster of 3 boards, each with a quad-core CPU running at 1.8 GHz and 8 GB of memory. The operating system running on all boards was Ubuntu version 22.04.3 with MPICH2 version 4.1.2.

To assess the quality of clustering in this large dataset, we evaluated the outcomes produced by the parallel ABC-CS algorithm and compared them with those of the parallel kNN algorithm [38] and the parallel SVM algorithm [39]. The clustering quality results, reflecting the F1 score achieved by these algorithms on this dataset, are depicted in Fig. 9.

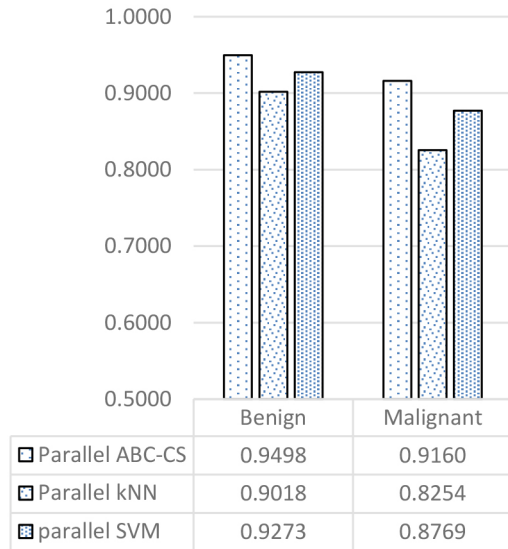


Fig. 9. The comparison of the F1 score for each class on the large dataset of parallel techniques

Fig. 9 illustrates the favorable performance of our parallel ABC-CS algorithm in the large dataset. The F1 scores obtained using the parallel ABC-CS method surpass those of the parallel kNN and parallel SVM methods, indicating that our proposed method achieves the best quantitative performance.

In the context of parallel computing [40], scalability refers to a system's ability to increase overall throughput as additional resources (processors) are added to handle increased workloads. Speedup, on the other hand, measures how much faster a parallel algorithm is than its corresponding sequential algorithm. The speedup (S_p) can be calculated using (6):

$$S_p = \frac{T}{T_N} \quad (6)$$

where the T is the running time using one processor and T_N is the running time using N processors.

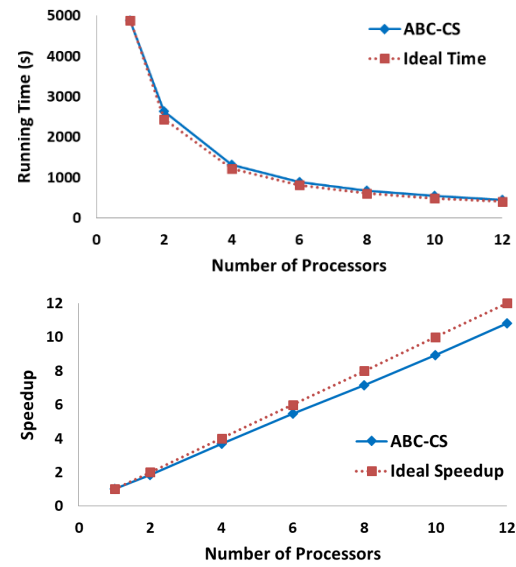


Fig. 10. Running time and speedup results of ABC-CS with 12 Raspberry Pi cluster processors

The scalability of our proposed method on the large dataset is shown in Fig. 10. As expected, increasing the number of processors used in our algorithm tends to reduce the runtime without sacrificing accuracy. Note that the ideal speedup is linear, in which the performance improvement is directly proportional to the number of processors. Thus, adding N processors to a task would reduce its execution time by a factor of N .

VI. DISCUSSION

This study demonstrates the feasibility of using the Artificial Bee Colony (ABC) algorithm with Cosine Similarity on a Raspberry Pi Cluster for medical data classification. The results show promising performance; however, the approach has limitations. Scalability with very large datasets may be constrained by the Raspberry Pi's limited processing power and memory, which can affect execution speed and efficiency.

Compared with previous studies, which primarily use high-performance computing, this work highlights the potential of low-cost, distributed hardware for medical data analysis. The integration of Cosine Similarity improved feature selection and classification

accuracy, supporting prior findings on similarity-based optimization while offering a novel application in embedded systems.

Future research could explore applications to medical imaging datasets, integration with deep learning models, and hybrid optimization techniques to enhance performance and robustness. Overall, this study provides a foundation for cost-effective and scalable medical data analysis, with clear directions for further investigation.

VII. CONCLUSIONS

This research aims to enhance medical data classification through the use of a new method called “ABC-CS”, which stands for Artificial Bee Colony with Cosine Similarity. The key contribution of this work is the development of an effective approach to optimize class centroids in a supervised manner using the ABC-CS technique. The performance of the proposed method was compared with well-established classification techniques, including k-NN, SVM, and ABC-KMeans, on the Breast Cancer Wisconsin dataset from the UCI machine learning repository. The results showed that the proposed ABC-CS method achieved superior classification accuracy compared to the other methods. Furthermore, the results obtained with the proposed method on Raspberry Pi clusters show that the parallel ABC-CS method is well-suited to handling massive amounts of data while maintaining the quality of the classification results. It can be concluded that the ABC-CS method offers a promising solution for enhancing medical data classification.

REFERENCES

- [1] H. Khan and M. Javaid, “Big data applications in medical field: A literature review,” *J. Ind. Integr. Manag.*, vol. 6, no. 1, pp. 53-69, Jul. 2021.
- [2] A. Kalantari, A. Kamsin, S. Shamshirband, A. Gani, H. Alinejad-Rokny, and A. T. Chronopoulos, “Computational intelligence approaches for classification of medical data: State-of-the-art, future challenges and research directions,” *Neurocomputing*, vol. 276, pp. 2-22, Feb. 2018.
- [3] L. D. L. Fuente-Tomas et al., “Classification of patients with bipolar disorder using k-means clustering,” *PLoS One*, vol. 14, no. 1, p. e0210314, Jan. 2019.
- [4] P. Nanglia, S. Kumar, A. N. Mahajan, P. Singh, and D. Rathee, “A hybrid algorithm for lung cancer classification using SVM and Neural Networks,” *ICT Express*, vol. 7, no. 3, pp. 335-341, Sep. 2021.
- [5] A. Hamed, A. Sobhy, and H. Nassar, “Accurate classification of COVID-19 based on incomplete heterogeneous data using a KNN variant algorithm,” *Arab. J. Sci. Eng.*, vol. 46, no. 9, pp. 8261-8272, Sep. 2021.
- [6] M. Z. Alam, M. S. Rahman, and M. S. Rahman, “A random forest based predictor for medical data classification using feature ranking,” *Inf. Med. Unlocked*, vol. 15, p. 100180, Apr. 2019.
- [7] R. A. Haraty, M. Dimishkieh, and M. Masud, “An enhanced K-means clustering algorithm for pattern discovery in healthcare data,” *Int. J. Distrib. Sens. Netw.*, vol. 11, no. 6, p. 615740, Jun. 2015.
- [8] Z. Wang, J. Na, and B. Zheng, “An improved kNN classifier for epilepsy diagnosis,” *IEEE Access*, vol. 8, pp. 100022-100030, May 2020.
- [9] J. Cervantes, F. García-Lamont, L. Rodríguez-Mazahua, and A. Lopez, “A comprehensive survey on support vector machine classification: Applications, challenges and trends,” *Neurocomputing*, vol. 408, pp. 189-215, May 2020.
- [10] X. Fan et al., “Review and classification of bio-inspired algorithms and their applications,” *J. Bionic Eng.*, vol. 17, no. 3, pp. 611-631, Aug. 2020.
- [11] A. Yadav and D. K. Vishwakarma, “A comparative study on bio-inspired algorithms for sentiment analysis,” *Clust. Comput.*, vol. 23, no. 4, pp. 2969-2989, 2020.
- [12] F. Valdez, O. Castillo, and P. Melin, “Bio-inspired algorithms and its applications for optimization in fuzzy clustering,” *Algorithms*, vol. 14, no. 4, p. 122, Dec. 2021.
- [13] V. Rajput, P. Mulay, and C. M. Mahajan, “Bio-inspired algorithms for feature engineering: Analysis, applications and future research directions,” *Inf. Discov. Deliv.*, vol. 53, no. 1, pp. 56-71, Jan. 2025.
- [14] S. S. Yadav and S. M. Jadhav, “Deep convolutional neural network based medical image classification for disease diagnosis,” *J. Big Data*, vol. 6, no. 1, pp. 1-18, Dec. 2019.
- [15] A. K. Dubey, A. Kumar, and R. Agrawal, “An efficient ACO-PSO-based framework for data classification and preprocessing in big data,” *Evol. Intell.*, vol. 14, no. 2, pp. 909-922, Jun. 2021.
- [16] R. Divya and R. S. S. Kumari, “Genetic algorithm with logistic regression feature selection for Alzheimer’s disease classification,” *Neural Comput. Appl.*, vol. 33, no. 14, pp. 8435-8444, Dec. 2021.
- [17] D. Karaboga, “An idea based on honey bee swarm for numerical optimization,” Tech. Rep. TR06, Erciyes Univ., Eng. Fac., Comput. Eng. Dept., Turkey, 2005.
- [18] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, “A comprehensive survey: Artificial Bee Colony (ABC) algorithm and applications,” *Artif. Intell. Rev.*, vol. 42, no. 1, pp. 21-57, 2014.
- [19] B. Akay and D. Karaboga, “A survey on the applications of Artificial Bee Colony in signal, image, and video processing,” *Signal Image Video Process.*, vol. 9, no. 4, pp. 967-990, Mar. 2015.
- [20] B. Akay, D. Karaboga, B. Gorkemli, and E. Kaya, “A survey on the Artificial Bee Colony algorithm variants for binary, integer and mixed integer programming problems,” *Appl. Soft Comput.*, vol. 106, p. 107351, Jul. 2021.
- [21] A. Banharnsakun, “Artificial bee colony algorithm for solving the knight’s tour problem,” in *Proc. Int. Conf. Intell. Comput. Optim.*, 2018, pp. 129-138.
- [22] A. Banharnsakun, “Artificial bee colony algorithm for content-based image retrieval,” *Comput. Intell.*, vol. 36, no. 1, pp. 351-367, Jan. 2020.
- [23] A. Banharnsakun, “Low-Light Image Enhancement with Artificial Bee Colony Method,” in *Proc. Int. Conf. Intell. Comput. Optim.*, Jan. 2021, pp. 3-13.
- [24] L. Syed, S. Jabeen, S. Manimala, and A. Alsaedi, “Smart healthcare framework for ambient assisted living using IoMT and big data analytics techniques,” *Future Gener. Comput. Syst.*, vol. 101, pp. 136-151, Jun. 2019.
- [25] P. Galetsi, K. Katsaliaki, and S. Kumar, “Big data analytics in health sector: Theoretical framework, techniques and prospects,” *Int. J. Inf. Manage.*, vol. 50, pp. 206-216, Feb. 2020.
- [26] L. Wang and C. A. Alexander, “Big data analytics in medical engineering and healthcare: Methods, advances and challenges,” *J. Med. Eng. Technol.*, vol. 44, no. 6, pp. 267-283, Jun. 2020.
- [27] M. D. Mudaliar and N. Sivakumar, “IoT-based real-time energy monitoring system using Raspberry Pi,” *Internet Things*, vol. 12, p. 100292, Dec. 2020.

- [28] D. K. Dewangan and S. P. Sahu, "Deep learning-based speed bump detection model for intelligent vehicle system using Raspberry Pi," *IEEE Sens. J.*, vol. 21, no. 3, pp. 3570-3578, Feb. 2020.
- [29] V. Gonzalez-Huitron et al., "Disease detection in tomato leaves via CNN with lightweight architectures implemented in Raspberry Pi 4," *Comput. Electron. Agric.*, vol. 181, p. 105951, Feb. 2021.
- [30] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques, Third Edition*. Massachusetts: Morgan Kaufmann Publishers, 2011, pp. 83-124.
- [31] L. Jiang and Y. Wang, "A wind power forecasting model based on data decomposition and cross-attention mechanism with cosine similarity," *Electr. Power Syst. Res.*, vol. 229, p. 110156, Jan. 2024.
- [32] M. Ahmad and M. Mazzara, "Scsnet: Sharpened cosine similarity-based neural network for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 21, pp. 1-4, Feb. 2024.
- [33] R. W. Nash, M. R. Crusoe, M. Kontak, and N. Brown, "Supercomputing with MPI meets the Common Workflow Language standards: An experience report," in *Proc. IEEE/ACM Workflows Support Large-Scale Sci.*, 2020, pp. 17-24.
- [34] R. Ehsani and F. Drabløs, "Robust distance measures for kNN classification of cancer cata," *Cancer Inform.*, vol. 19, pp. 1-9, Oct. 2020.
- [35] S. Huang et al., "Applications of Support Vector Machine (SVM) learning in cancer genomics," *Cancer Genomics Proteomics*, vol. 15, no. 1, pp. 41-51, Dec. 2018.
- [36] M. Krishnamoorthi and A. M. Natarajan, "ABK-means: An algorithm for data clustering using ABC and K-means algorithm," *Int. J. Comput. Sci. Eng.*, vol. 8, no. 4, pp. 383-391, Oct. 2013.
- [37] W. Wolberg, "Breast cancer wisconsin (original)." *UCI Machine Learning Repository*. 1990. [Online]. Available: <https://doi.org/10.24432/C5HP4Z> [Accessed: Sep. 25, 2025].
- [38] S. Athani, S. Joshi, B. A. Rao, S. Rai, and N. G. Kini, "Parallel implementation of kNN algorithm for breast cancer detection," in *Proc. Evol. Comput. Intell. FICTA 2020*, 2020, pp. 475-483.
- [39] Z. Kang, N. Xiao, Z. Chen, Y. Ou, and X. Li, "An improved parallel SVM algorithm on distributed system," in *Proc. Int. Conf. CyberC*, 2020, pp. 204-209.
- [40] D. Jaiswal and P. Kumar, "A survey on parallel computing for traditional computer vision," *Concurrency Comput.: Pract. Exp.*, vol. 34, no. 4, p. e6638, Sep. 2022.



Anan Banharnsakun is an associate professor in the Department of Computer Engineering, Kasetsart University Sriracha Campus, Thailand. He received a Ph.D. degree in Electrical and Computer Engineering from King Mongkut's University of Technology Thonburi, Thailand. His research interests include biologically inspired computational intelligence, robust optimization, pattern recognition, and high-performance computing.

A Hybrid Optimization-Simulation Approach for Supply Chain Network Design under Uncertainty and Highly Perishable Environments

Jirawat Chatchaichalermporn¹ and Navee Chiadamrong^{2*}

^{1,2}School of Manufacturing Systems and Mechanical Engineering (MSME),
Sirindhorn International Institute of Technology, Thammasat University, Pathumthani, Thailand
E-mail: m6622040910@g.siiit.tu.ac.th and navee@siiit.tu.ac.th*

Received: August 15, 2025 / Revised: October 12, 2025 / Accepted: November 10, 2025

Abstract—Designing and managing logistics and supply chain networks for perishable products requires careful consideration of product quality degradation over time due to environmental conditions and processing delays. This study proposes a Hybrid Optimization-Simulation (HOS) framework that combines a Mixed-Integer Linear Programming (MILP) model with a Discrete Event Simulation (DES) to design a cost-efficient and quality-preserving distribution network for floral products. The MILP model incorporates a Time-Temperature Sum (TTS) constraint to account for product perishability, while the simulation model evaluates the network's operational feasibility under dynamic conditions. A case study of the floriculture supply chain, involving growers, hubs, and retailers, is used to validate the proposed approach's performance. The results show that the deterministic MILP achieves the lowest total cost of \$163,418, but does not account for uncertainty. In contrast, the proposed HOS outperforms the typical Simulation-Based Optimization (SBO) by 4.78% in total cost and achieves a much shorter computation time. In addition, the HOS framework can produce logistics and supply chain network configurations that balance cost efficiency and quality assurance, providing a robust tool for supply chain design in highly perishable environments.

Index Terms—Perishable Products, Mixed Integer Linear Programming (MILP), Simulation, Hybrid Optimization Approach

I. INTRODUCTION

The design of logistics and supply chain networks for perishable products presents distinct challenges due to the rapid deterioration of product quality over time. In supply chains, handling products such as cut flowers, freshness is a critical performance measure, and quality degradation is heavily influenced by

factors such as temperature exposure and delays during transportation and processing. As these factors interact dynamically, traditional logistics planning approaches that focus solely on cost minimization often fail to maintain acceptable service levels or reduce spoilage.

A typical cut flower supply chain offers a representative example of this complexity, in which flowers are cultivated in geographically dispersed regions and routed through distribution hubs for processes such as bouquet-making before reaching retailers. These processes often require the timely combination of different flower types, further complicating logistics coordination. The perishability of flowers, typically measured in vase life, requires that strategic decisions—such as facility placement, flow routing, and process allocation—explicitly account for product quality through the supply chain.

Conventional optimization techniques such as Mixed-Integer Linear Programming (MILP) have been extensively used in supply chain network design. However, these models often rely on static approximations and do not adequately capture the dynamic nature of quality decay. In contrast, simulation models can incorporate time-dependent and uncertain elements such as transport variability and temperature fluctuations, providing more realistic evaluations of operational performance. Nonetheless, simulation alone lacks the prescriptive capabilities of optimisation in determining optimal network configurations.

To bridge this gap, Hybrid Optimization-Simulation (HOS) approaches have emerged, combining the prescriptive strength of optimization with the descriptive richness of simulation. A notable contribution in this area is the model developed by De Keizer et al. [1], who combined an MILP with DES to design logistics networks for cut flowers. Their approach used simulation feedback to iteratively adjust product-deterioration constraints in the MILP, enabling the design to reflect real-world operational conditions such as uncertain supply, processing delays, and variable transport temperatures.

Building upon this framework, the present study adopts an HOS approach to the supply chain network design. This research also emphasises the importance of incorporating time-temperature-sum quality metrics into strategic decision-making. This study also makes four additional contributions beyond previous studies. Firstly, a modified HOS approach is proposed to tradeoff between degrading product quality and the customer's service level, subject to minimising total costs. Secondly, it extends the HOS methodology by varying the levels of uncertainty in the simulation model to improve the realism and robustness of supply chain performance evaluations. Thirdly, it introduces OptQuest, an optimization tool embedded in ARENA, as a benchmarking tool to evaluate and validate the MILP and HOS solutions using metaheuristic-based search techniques. This dual-layered validation provides greater confidence in the effectiveness of the proposed network designs, especially under dynamic and uncertain conditions. Finally, this research contributes a practical implementation of the HOS approach by developing and constructing the hybrid framework in a real-world simulation platform. This demonstrates how theoretical optimization-simulation integration can be operationalized effectively, offering actionable insights for decision-makers in perishable supply chains. These advancements support the development of more resilient and empirically grounded logistics networks for perishable products.

As a result, the research contributes to the growing body of literature that seeks to enhance supply chain resilience and efficiency in environments characterized by high perishability and operational uncertainty.

II. LITERATURE REVIEW

This literature review section begins by examining prior related studies on perishable products in supply chain network design, followed by discussions of optimization, simulation, and hybrid methodologies, particularly those focused on process allocation and product quality degradation.

A. Perishable Products in Supply Chains

Perishable products such as fresh food and cut flowers are inherently vulnerable to quality degradation, which significantly complicates supply chain planning and execution. These products are highly sensitive to environmental conditions—particularly time and temperature—making quality preservation a critical objective throughout transportation, storage, and processing. Managing the flow of perishables requires not only timely delivery but also specialized handling protocols and temperature-controlled logistics. Musavi and Bozorgi-Amiri [2] emphasized the importance of freshness in their sustainable hub location-vehicle scheduling model,

which incorporated both product perishability and environmental impact through CO₂ emissions. Their approach illustrated how sustainability goals and quality preservation could be jointly addressed in strategic supply chain decisions. Similarly, Rong et al. [3] developed a framework for integrating food quality degradation into an MILP model. Their study focused on modeling the deterioration of food products over time and incorporating this into production and distribution planning. By doing so, they bridged the gap between quality control and logistical decision-making in food supply chains. Next, Pasha et al. [4] further advanced this domain by modeling quality as a function of both time and temperature in a two-echelon distribution system. Their work demonstrated how product quality could be monitored dynamically to influence both inventory control and network design decisions. Agustina et al. [5] experimented on the production-distribution planning for perishable products, emphasizing that shelf-life dynamics and cold-chain responsiveness were central to efficient network design. These contributions highlighted that perishable supply chains cannot be effectively managed without incorporating the physical decay processes into the decision-making framework.

B. Optimization in Perishable Supply Chains

A wide range of optimization techniques has been applied to enhance the performance of supply chains dealing with perishable products. Mathematical modeling is essential for its prescriptive power and ability to find globally optimal solutions under defined constraints. These mathematical models are especially useful for determining optimal facility locations, inventory policies, and routing decisions under resource and time constraints. In the context of perishables, these models must also account for the limited shelf life and quality degradation that affect the economic value of goods over time. For instance, Mohebalizadehgashti et al. [6] developed a multi-period, multi-product, multi-echelon MILP model for a green meat supply chain in Southern Ontario that simultaneously minimizes total cost and CO₂ emissions while maximizing capacity utilization, using the augmented ϵ -constraint method to obtain Pareto-optimal solutions. They further incorporated the decision tree analysis to assess the impact of demand and purchasing cost uncertainty, offering decision-makers balanced trade-offs for sustainable perishable supply chain design. Mohammadi et al. [7] addressed routing optimization in agri-food supply chains, where timely delivery was essential to minimize quality losses. Their hybrid algorithm sought to reduce both operational costs and perishability-driven losses, demonstrating the dual objectives often presented in such networks. A particularly notable contribution is by De Keizer et al. [1], who formulated an MILP to

design a distribution network for cut flowers. This model incorporated a Time-Temperature Sum (TTS) constraint to ensure product quality thresholds were met across the supply chain. Their framework enabled decision-makers to assess the trade-off between cost efficiency and product freshness, particularly in networks with multiple product types and processing stages. Collectively, these studies have reinforced the relevance and adaptability of optimization methods in addressing the unique challenges of perishable product distribution. Then, Shirzadi et al. [8] addressed optimization in perishable supply chains by developing a green inventory-routing model with simultaneous pickup and delivery that integrates postharvest biological behavior, greenhouse gas emission control, and reverse logistics. Their bipolar fuzzy optimization framework incorporated freshness and ripeness into pricing and satisfaction while managing uncertainty in quality and holding costs, offering a more comprehensive solution than previous models that considered these factors separately.

In addition, metaheuristic approaches have been widely used to handle complex, nonlinear problems that traditional solvers may struggle with, particularly under time constraints or in large solution spaces. Pan et al. [9] introduced a hybrid metaheuristic optimization for multi-echelon perishable supply chains, improving computational efficiency and robustness under uncertainty. More recently, reinforcement learning has emerged as a promising approach, enabling the learning of adaptive decision-making policies through interaction with dynamic environments, a capability that is especially useful for real-time control and routing in uncertain supply chains.

C. Simulation in Supply Chain Modeling

While MILP optimization models offer prescriptive decision-making tools, they often rely on static assumptions that may not capture the full complexity and uncertainty of real-world perishable supply chains. In contrast, simulation provides a powerful means to evaluate system performance under dynamic and stochastic conditions. Discrete Event Simulation (DES) is particularly effective in modeling time-dependent processes such as transport delays, machine breakdowns, and fluctuating demand—all of which can significantly impact product freshness and delivery reliability. For example, Nakandala et al. [10] incorporated simulation to evaluate their optimization outcomes, allowing them to assess quality performance under variable transport conditions. Their approach provided critical insights into how operational uncertainties affected quality-related metrics, which were often overlooked in deterministic models. Sel and Bilgen [11] developed a simulation-supported decision-making framework

for the beverage industry, combining MILP-based heuristics with DES to evaluate operational feasibility. Their framework was capable of adapting to machine failures and uncertain production rates using a rolling horizon planning approach. This integration allowed for more accurate forecasting and responsiveness in high-variability environments. Ásmundsson et al. [12] applied simulation modeling to analyze and improve supply chain performance, focusing on identifying bottlenecks, testing alternative policies, and evaluating operational changes in a risk-free environment. Then, Gaukler [13] demonstrated that integrating time-temperature history data into simulations significantly enhanced the accuracy of logistics decision-making for perishable foods. In summary, these studies have shown that simulation is an essential complement to optimization, particularly for perishable goods that are highly sensitive to real-time disruptions. It allows researchers and practitioners to test logistics configurations before implementation, improving both reliability and service performance.

D. Hybrid Optimization–Simulation Approaches

Hybrid Optimization–Simulation (HOS) approaches have emerged as a promising methodology for tackling the complexity of perishable supply chain design and operation. These approaches leverage the strengths of both optimization—providing optimal or near-optimal solutions under defined constraints—and simulation, which evaluates system performance under realistic, variable conditions. For example, Sel and Bilgen [11] implemented a hybrid method by combining MILP-based heuristics with simulation in the beverage industry. Their rolling-horizon approach enabled sequential decision-making while responding dynamically to changes in system status. Ding et al. [14] proposed an HOS approach using a Multi-Objective Genetic Algorithm (MOGA) and Discrete Event Simulation (DES) to design production-distribution networks under uncertainty. Their model simultaneously optimized network structure, operational strategies, and control parameters while capturing dynamic and stochastic system behavior. Ko et al. [15] developed an HOS framework for third-party logistics providers (3PLs), combining a genetic algorithm for network design with simulation to evaluate warehouse capacity and service time under uncertain demand and processing variability. Such hybrid frameworks are particularly valuable for perishable products, where operational realities like spoilage, temperature sensitivity, and time constraints play a critical role. By integrating both planning and evaluation layers, HOS approaches can offer a more comprehensive and practical solution for managing uncertainty and ensuring quality in perishable supply chains. Then, Abbasian et al. [16] developed a bi-objective hybrid optimization method incorporating location, inventory, and routing

decisions for a perishable food supply network, and addressed resiliency and sustainability under disruption-risk conditions.

Most existing optimization models for perishable supply chains focus primarily on deterministic environments and cost-minimizing objectives, neglecting the influence of real-world uncertainties such as process time variation, buffering delays, and fluctuating temperature conditions that directly affect the product freshness and the service level performance. As a result, this study tries to fill the gap by integrating a Discrete Event Simulation (DES) with a Mixed-Integer Linear Programming (MILP) model to capture the dynamic effects of uncertainty on product quality through Time-Temperature Sum (TTS) constraints.

III. PROBLEM DESCRIPTION AND OPTIMIZATION MODEL

A. Problem Description

This study focuses on the cut-flower supply chain, from growers to retailers, with floricultural clusters (such as auctions and traders) in between. These clusters are where flowers are collected, processed into bouquets, and sent to retailers. The model builds upon the conceptual framework proposed by De Keizer et al. [1], who analyzed the distribution network of cut flowers in the Netherlands.

In this supply chain network, growers deliver flowers to auction centers at scheduled times. At the auction, traders buy the flowers and arrange delivery to their facilities. After that, the flowers are processed into bouquets and delivered to retailers. Flowers that are not sold or processed in time are thrown away because they no longer meet customer requirements. Traders can also buy flowers directly from growers, skipping the auction, but the time and quality limits still apply.

A key decision in designing this network is where to do the bouquet-making. Since bouquets take up more space than single flowers, it may seem better to make them close to the retailer. However, doing all processing in one place (centralized) can reduce costs. On the other hand, bringing flowers from different locations together may cause delays. This can lead to a loss of quality if the flowers are not combined on time. So, the location of bouquet-making must balance transport costs and the risk of quality degradation.

The quality of flowers is measured by their vase life, where Tromp et al. [17] as how many days the flowers can stay fresh in a vase at room temperature (usually 20°C). Quality loss is calculated using the Time-Temperature Sum (TTS) model. It depends on

the starting vase life (A) and the Time Temperature Sum (TTS), which shows how temperature and time affect the flowers during transport and storage. The formula is: $VL = A - TTS$, where $TTS = \sum_i time_i * temperature_i$. i is the process in the supply chain. TTS adds up the effect of time and temperature at each stage in the supply chain. A higher TTS indicates a shorter vase life. As each retailer has a service level requirement, which means a certain percentage of flowers must meet a minimum vase life, the network design must find the best locations and product flows to keep costs low while meeting these quality rules.

Each grower supplies one type of flowers or green ornamentals, representing specialized production. There are five flower types (Alstroemeria, Lilium, Tulipa, Gerbera, and Rosa) and two ornamental types (Pittosporum and Ruscus). Ten retailers (see Fig. 1) are included in the model, each placing demand for bouquets that are mono-type (single flower type) or mixed-type (combination of flowers and greens). Table I presents the bill of materials for mono and mixed bouquets, specifying the required input quantities of each flower type and their available supply from growers. The differentiation between these bouquet types introduces synchronization challenges, particularly for mixed bouquets that require the timely availability of all input components. Nine potential hub locations (see Fig. 2) are included as intermediate facilities capable of storing (buffering) and performing bouquet-making processes. Hubs also differ in buffer durations and processing times, reflecting differences in operational efficiency. All nine hubs have identical unit cost parameters. Each hub requires a fixed installation cost of \$10,000, and an additional fixed cost of \$5,000 for installing the bouquet-making process. The variable processing cost is \$1 per unit. Transportation costs and temperature during transport are calculated based on time and shipment type (flowers or bouquets), with additional consideration of whether the transport is cooled or non-cooled. When the transport time is longer than 6 hours, the cooled transport (7 °C) is used in which is more expensive than non-cooled transport (14 °C) (see Table IV). There are 10 retailers (see Fig. 3), each responsible for receiving a mix of mono and mixed bouquets based on their demand and quality requirements. These retailers set specific service-level targets and required vase-life thresholds that the network must meet to ensure customer satisfaction. As a result, the network design must find the best locations and product flows to keep costs low while meeting these quality rules. The input data are shown in Tables I-IV and Figs 1-3.

TABLE I
BILL OF MATERIALS AND SUPPLY DATA

	Type of Cut Flowers (flowers)				Type of Ornamentals (flowers)		
	Alstroemeria	Lilium	Tulipa	Gerbera	Rosa	Pittosporum	Ruscus
Mono Bouquet	-	-	15	-	-	-	-
Mixed Bouquet	3	3	-	3	5	4	4
Supply	6,450	6,450	15,000	6,450	10,750	8,600	8,600

TABLE II
BUFFERING AND PROCESSING TIME DATA

Hub	Buffering Time (Hours)	Processing Time (Hours)
Hub 1	8	5
Hub 2	3	4
Hub 3	5	3
Hub 4	5	3
Hub 5	5	3
Hub 6	5	3
Hub 7	5	3
Hub 8	9	4
Hub 9	12	5

TABLE III
CUSTOMER REQUIREMENT AND DEMAND DATA

Retailer	Required Product Quality (Days)	Service Level (%)	Demand (Bouquet)	
			Mono Bouquet	Mixed Bouquet
Retailer 1	8	90%	100	215
Retailer 2	8	95%	100	215
Retailer 3	8	90%	100	215
Retailer 4	8	95%	100	215
Retailer 5	9	95%	100	215
Retailer 6	9	95%	100	215
Retailer 7	9	90%	100	215
Retailer 8	9	95%	100	215
Retailer 9	8	90%	100	215
Retailer 10	8	90%	100	215

TABLE IV
TEMPERATURE AND TRANSPORTATION COST

From/To	Temperature		Costs/km/1,000 units of flower (\$)			
			Flower		Bouquet	
	<6 Hours	>6 Hours	<6 Hours	>6 Hours	<6 Hours	>6 Hours
Grower to Hub	14 °C	7 °C	1	1.2	-	-
Hub to Hub	14 °C	7 °C	0.5	0.9	2	2.4
Hub to Retailer	14 °C	7 °C	-	-	3	6.4

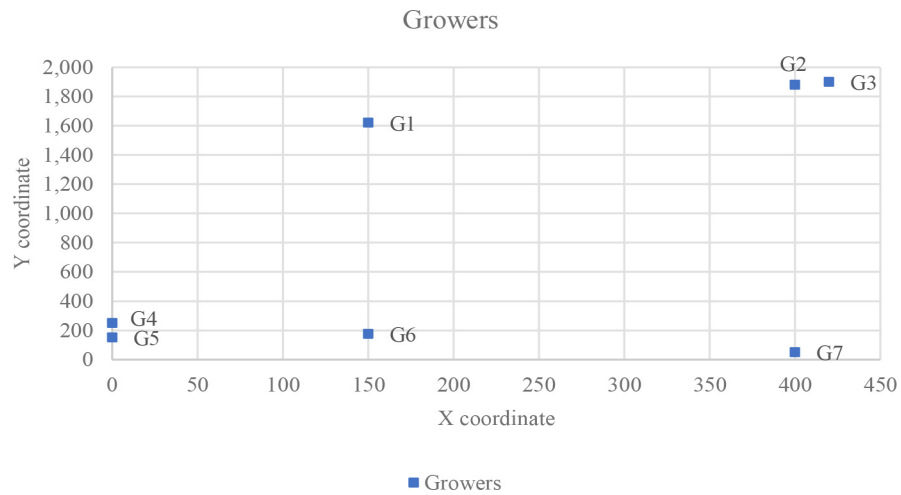


Fig. 1. Growers location coordinate

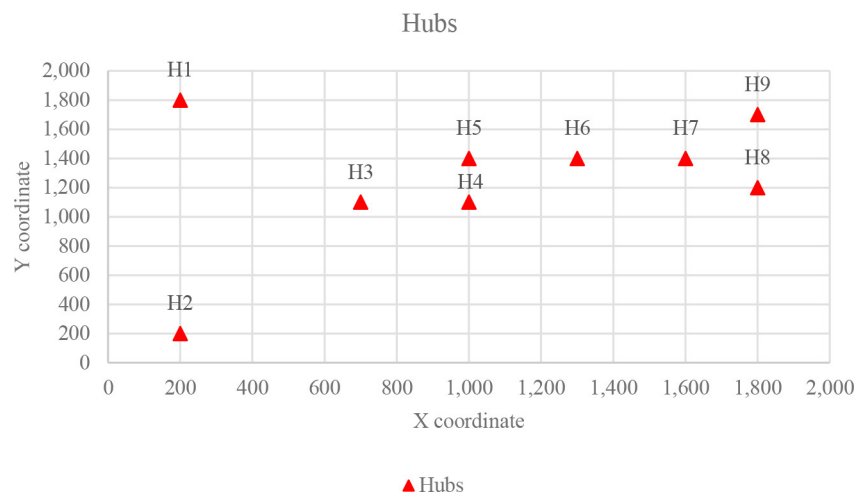


Fig. 2. Hubs location coordinate

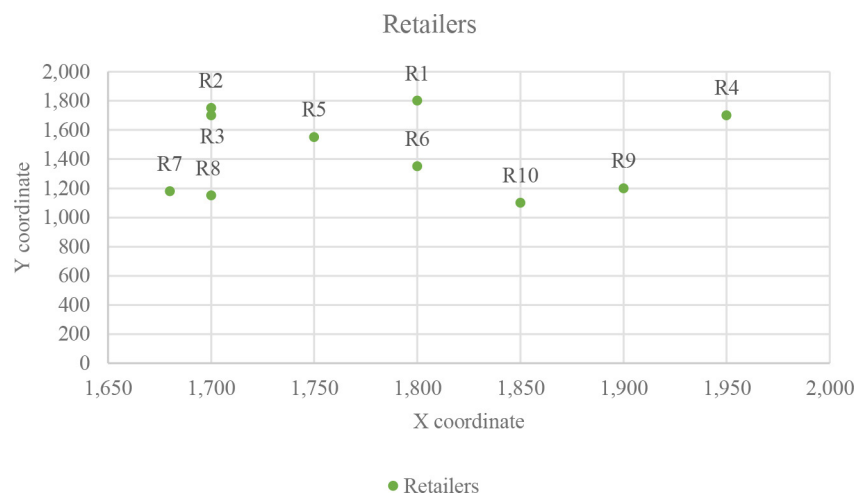


Fig. 3. Retailers' location coordinates

B. Problem Assumptions

The following are the assumptions that are imposed in the study:

- The initial vase life for each type of flower is 11 days.
- Capacities of hubs are assumed to be unlimited.
- Transporters are homogeneous with unlimited availability.
- Values of all parameters are deterministic while solving with the mathematical model.
- Values of all costs are always deterministic.
- Process time, buffering time, and temperature during the transport and processing can be uncertain in the simulation model, and they follow the normal distribution.

C. Mathematical Model

This section outlines the mathematical model developed for designing a logistics and supply chain network tailored to the distribution of perishable products. The MILP model aims to minimize the total costs of the network by determining optimal hub locations, assigning processes, and planning product flows.

Indices and sets:

$g \in G$ set of growers

$r \in R$ set of retailers

$i, j \in H$ set of hubs

$p, q \in P$ set of products (flowers and bouquets)

P_p set of input products for the process that produces product p (flowers)

Decision Variables:

Y_i hub open or not

Y_{ip} process producing product p installed at hub i or not (1 if $X_{ip} > 0$, 0 otherwise)

XG_{gip} quantity of product p transported from grower g to hub i (flowers)

XH_{ijp} quantity of product p transported from hub i to hub j (flowers/bouquets)

XR_{irp} quantity of product p transported from hub i to retailer r (bouquets)

X_{ip} quantity of product p produced at hub i (bouquets)

Auxiliary variables:

$ZH_{ijp} = TTSH_{ip} YH_{ijp}$

$ZR_{irp} = TTSH_{ip} YR_{irp}$

$Z_{iqp} = TTSH_{ip} Y_{ip}$

Parameters:

fc_i fixed costs for opening hub i (\$)

fc_{ip} fixed costs for installing the process producing product p at hub i (\$)

cg_{gip} unit transportation costs of product p from grower g to hub i (\$)

ch_{irp} unit transportation costs of product p from hub i to hub r (\$)

cr_{irp} unit transportation cost of product p from hub i to retailer r (\$)

c_{ip} unit production costs of product p at hub i (\$)

sup_{gp} supply of product by grower g to hub i (flowers)

dem_{rp} demand of product p by retailer r (flowers)

m_{pq} quantity of product p in one unit of product q (flowers)

tts_{gip} TTS for transporting product p from grower g to hub i (degree-days)

tts_{hijp} TTS for transporting product p from hub i to hub j (degree-days)

tts_{irp} TTS for transporting product p from hub i to retailer r (degree-days)

tts_{ip} TTS for producing product p at hub i (degree-days)

tts_{bip} TTS for buffer of product p at hub i (degree-days)

$maxtts_{rp}$ maximum TTS to have good quality of product p at retailer r (degree-days)

up_{ip} upper bound of $TTSH_{ip}$ (degree-days)

$$\min \sum_{i \in H} fc_i Y_i + \sum_{i \in H} \sum_{p \in P} fc_{ip} Y_{ip} + \sum_{g \in G} \sum_{i \in H} \sum_{p \in P} cg_{gip} XG_{gip} + \sum_{i \in H} \sum_{j \in H} \sum_{p \in P} ch_{ijp} XH_{ijp} + \sum_{i \in H} \sum_{r \in R} \sum_{p \in P} cr_{irp} XR_{irp} + \sum_{i \in H} \sum_{p \in P} c_{ip} X_{ip} \quad (1)$$

s.t.

$$\sum_{i \in H} XG_{gip} = sup_{gp} \quad \forall g, p \quad (2)$$

$$\sum_{i \in H} XR_{irp} = dem_{rp} \quad \forall r, p \quad (3)$$

$$\sum_{g \in G} XG_{gip} + \sum_{j \in H} XH_{jip} + X_{ip} = \sum_{j \in H} XH_{jip} + \sum_{r \in R} XR_{irp} + \sum_{q \in P} m_{gq} \quad \forall i \in H, p \in P \quad (4)$$

$$\sum_{g \in G} \sum_{p \in P} XG_{gip} + \sum_{j \in H} \sum_{p \in P} XH_{jip} + \sum_{p \in P} X_{ip} \leq [\sum_{g \in G} \sum_{p \in P} sup_{gp}] Y_i \quad \forall i \quad (5)$$

$$X_{ip} \leq \sum_{g \in G} sup_{gp} Y_{ip} \quad \forall i, p \quad (6)$$

$$Y_i \in \{0, 1\} \quad \forall i \quad (7)$$

$$Y_{ip} \in \{0, 1\} \quad \forall i, p \quad (8)$$

$$XG_{gip} \geq 0 \quad \forall g, i, p \quad (9)$$

$$XH_{ijp} \geq 0 \quad \forall i, j, p \quad (10)$$

$$XR_{irp} \geq 0 \quad \forall i, r, p \quad (11)$$

$$X_{ip} \geq 0 \quad \forall i, p \quad (12)$$

Equation (1) is to minimize total costs consisting of fixed location, process costs, transportation costs, and variable costs. Equation (2) ensures that the total amount of product p shipped from grower g to all hubs equals the grower's supply for that product. Equation (3) states that the total quantity of product p delivered from hubs to each retailer r must meet the retailer's demand. Equation (4) maintains the flow balance at each hub i for every product p . The total amount of product p entering the hub, whether from growers, other hubs, or as a result of local processing, must equal the amount leaving the hub, either to another hub, to a retailer, or to another process as an input. Equation (5) ensures that product flows through a hub only occur if the hub is opened. Equation (6) ensures that production can only take place if the

related process is installed. Equations (7) - (12) define binary and non-negative variables.

$$TTSR_{rp} = \sum_{r \in R} (TTS_{ip} + ttsr_{ip}) YR_{irp} \quad \forall i, p \quad (13)$$

Equation (13) calculates the total TTS for product p delivered to retailer r . This value represents the total exposure of a product to temperature over time from its point of origin until it reaches the retailer. Subsequently, there are 3 possible ways to get product to hub :

$$1. TTS_{ip}^{grower} = \sum_{g \in G} (tts_{gip} + ttsb_{ip}) YG_{gip} \quad \forall i, p \quad (14)$$

Equation (14) calculates the TTS of product p arriving at hub i when it is delivered directly from a grower g . This TTS includes the time-temperature exposure during transport from the grower to the hub and the buffering time at the hub.

$$2. TTS_{ip}^{hub} = \sum_{j \in G} (TTS_{jip} + tsh_{jip} + ttsb_{ip}) YH_{jip} \quad \forall i, p \quad (15)$$

Equation (15) represents the TTS of product p arriving at hub i when it is sent from another hub j . The total TTS includes the TTS already accumulated at hub j , the transport TTS between hubs j and i , and the buffer time at hub i .

$$3. TTS_{ip}^{process} = \sum_{q \in P_p} (TTS_{ip} + tsh_{ip} + ttsb_{ip}) Y_{ip} \quad \forall i, p \quad (16)$$

Equation (16) covers the case where product p is produced at hub i through a process. The TTS in this case includes the TTS of the input product q at hub i , the processing time at the hub, and the buffering time at hub i .

$$TTS_{ip} = \max[TTS_{ip}^{grower}, TTS_{ip}^{hub}, TTS_{ip}^{process}] \quad (17)$$

Equation (17) defines the TTS at hub i for product p . This value captures the maximum TTS among all possible ways product p can arrive at hub i .

$$(TTS_{ip} + ttsr_{ip}) YR_{irp} \leq \max tts_{rp} \quad \forall i, r, p \quad (18)$$

Equation (18) ensures that the total TTS for product p delivered to retailer r must be lower than or equal to the maximum allowance TTS at retailer r for product p .

$$(tts_{gip} + ttsb_{ip}) YG_{gip} \leq TTS_{ip} \quad \forall g, i, p \quad (19)$$

Equation (19) states that the sum of transport and buffering TTS from grower g to hub i is less than or equal to the maximum TTS at hub i .

$$(TTS_{jip} + tsh_{jip} + ttsb_{ip}) YH_{jip} \leq TTS_{ip} \quad \forall i, j, p \quad (20)$$

Equation (20) addresses the scenario where product p is transferred from hub j to hub i , including buffering time must be lower than or equal to the maximum TTS at hub i .

$$(TTS_{ip} + tsh_{ip} + ttsb_{ip}) Y_{ip} \leq TTS_{ip} \quad \forall i, p \quad (21)$$

Equation (21) ensures that TTS of the output product, which are TTS of input product q , processing time and buffering time at hub i should be less than or equal to maximum TTS at hub i .

Equations (18), (20) and (21) are non-linear since they contain the products of binary and bounded non-negative variables. To eliminate multiplications of binary and bounded non-negative variables (18), (20) and (21), those equations translate to (22) - (25), (26) - (29) and (30) - (33), respectively:

$$ZR_{irp} + ttsr_{ip} YR_{irp} \leq \max tts_{rp} \quad \forall i, r, p \quad (22)$$

$$ZR_{irp} \leq YR_{irp} up_{ip} \quad \forall i, r, p \quad (23)$$

$$ZR_{irp} \leq TTS_{ip} up_{ip} \quad \forall i, r, p \quad (24)$$

$$ZR_{irp} \leq TTS_{ip} - (1 - YR_{irp}) up_{ip} \quad \forall i, r, p \quad (25)$$

$$ZH_{jip} + (tsh_{jip} + ttsb_{ip}) YH_{jip} \leq TTS_{ip} \quad \forall i, j, p \quad (26)$$

$$ZH_{jip} \leq YH_{jip} up_{jp} \quad \forall i, j, p \quad (27)$$

$$ZH_{jip} \leq TTS_{jp} \quad \forall i, j, p \quad (28)$$

$$ZH_{jip} \leq TTS_{jp} - (1 - YH_{jip}) up_{jp} \quad \forall i, j, p \quad (29)$$

$$Z_{iqp} + (tts_{ip} + ttsb_{ip}) Y_{ip} \leq TTS_{ip} \quad \forall i, p, q \quad (30)$$

$$Z_{iqp} \leq Y_{ip} up_{iq} \quad \forall i, q, p \quad (31)$$

$$Z_{iqp} \leq TTS_{jq} \quad \forall i, p, q \quad (32)$$

$$Z_{iqp} \leq TTS_{jq} - (1 - Y_{ip}) up_{iq} \quad \forall i, p, q \quad (33)$$

D. Simulation Model

The Discrete Event Simulation (DES) model, which is built in ARENA, is used to evaluate whether the logistics and supply chain network design meets customer requirements. The model follows the actual flow of individual products from growers to retailers and checks their quality upon delivery. Unlike the MILP model, the simulation captures detailed dynamics such as delays and temperature variations during transport and processing. For the base scenario, all uncertainties follow a normal distribution with their estimated means as the mean and standard deviations set at 7% of their means. However, different levels of each uncertainty will be further explored in the sensitivity analysis to assess their impact. The objective of the simulation is to test the feasibility of the network design proposed by the MILP model, with a focus on product quality. It simulates the real-world operation of the supply chain, including transport, buffering, and processing activities. The scope of the model covers the entire flow from the growers to the retailers, incorporating the effects of uncertainty in supply times, process durations, transport delays, and temperature conditions. It also evaluates whether products meet the minimum vase life required by each retailer.

The simulation model operates by tracing each product unit through the network. Each product is assigned key attributes such as current vase life, temperature exposure, and time spent at each stage. The model begins with growers generating daily supplies of flower types, which are transported to assigned hubs. At these hubs, products may be stored temporarily in buffers or processed into final forms, such as mixed bouquets. Bouquet-making processes

at the hubs begin only after all required input flows have arrived, in accordance with the proportions defined in the MILP solution. After processing, products are sent to retailers.

FIFO (First-In-First-Out) logic is used for buffering, and the model applies randomness to process durations, transport delays, and temperature conditions. It dynamically updates product quality using the degree-days model at each step. Each retailer specifies a minimum vase life and a target service level, which defines the percentage of deliveries that must meet the vase life requirement. At the end of a simulation run, the achieved service level for each retailer-product pair is computed by comparing actual delivery quality against these thresholds. The model is designed to be a terminating system, with 10 replications representing a 45-day operational cycle each. Across 10 replications, the variation in the objective value (total costs) is 3.68% relative to its mean.

E. Hybrid Optimization-Simulation Approach

This section introduces the HOS approach, which connects the MILP model with the simulation model. In this framework, as shown in Fig. 3, the MILP model is first used to generate an optimal network configuration that satisfies cost and quality constraints, including Time–Temperature–Sum (TTS) thresholds. The resulting solution—specifying facility locations, flow allocations, and process assignments—is then tested in the DES model to assess whether it meets the required service levels under operational uncertainty. The simulation captures real-world variability, such as delays and

temperature fluctuations, and calculates the actual vase life of delivered products. If the simulation indicates that quality targets are not met, the feedback is used to iteratively adjust the maximum TTS constraints in the MILP model based on required quality (β_{rp}) and realized quality (γ_{rp}): $\maxtts_{rp} * \gamma_{rp} / \beta_{rp}$. For example, if the realized quality, which is the actual service level obtained from the simulation is, 87% in an iteration, and the required quality (which is the required service level of the retailer is 95%) and $\maxtts_{rp}$ (which is the maximum TTS of product p at retailer r) is 100 degree-days and the Vase Life (VL) of 6 days, the new $\maxtts_{rp}$ would be $\frac{0.87}{0.95} * \maxtts_{rp} = 91.58$ degree-days. From $VL = A - \frac{1}{20} TTS$, A is the initial vase life (which is set at 11 days), so the new tightening VL would be 6.42 days. This VL duration exceeds the maximum of 6 days. Such an adjustment enables the system to accept higher-quality products by expanding the acceptance level for Vase Life (VL). As a result, the system will respond by finding an alternative route that is quicker and can accommodate a longer VL, but the cost could be higher than the route from the previous iteration.

This loop continues until the proposed network design satisfies both cost-efficiency and customer requirements under dynamic conditions. Although the MILP model ensures feasibility based on predefined constraints, the solution may become infeasible. A key reason for this infeasibility is that no available route or network configuration in the current solution can keep the TTS below the required maximum. This integration allows the strengths of both methods to be leveraged: MILP provides optimal planning, while DES ensures feasibility and robustness in execution.

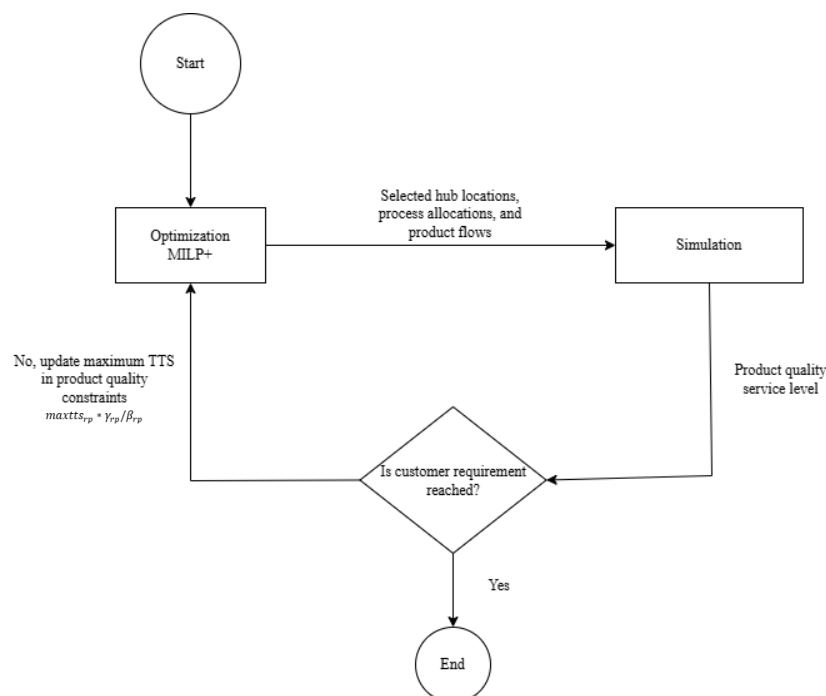


Fig. 3. HOS approach

F. Simulation-Based Optimization with OptQuest (SBO)

Simulation-Based Optimization (SBO) with OptQuest is a technique that combines DES with metaheuristic search algorithms to optimize decision variables within a stochastic environment. In this study, OptQuest—integrated within the ARENA simulation platform—is used to identify network configurations that minimize total costs while maintaining required service levels for customer requirements. The approach evaluates the performance of each solution through simulation runs, capturing the effects of randomness in processing times, buffering durations, and temperature conditions, which directly influence product freshness. OptQuest iteratively explores different combinations of hub selections and allocations of bouquet-making processes using heuristic methods. This method is particularly useful in complex systems where analytical models may not fully capture the operational dynamics, making it a practical tool for improving supply chain performance under real-world variability. In this research, SBO is used as a benchmark to assess and validate the performance of other proposed solutions.

IV RESULTS AND DISCUSSION

This section presents the findings from the experiments using the MILP, Simulation-Based Optimization (SBO), and Hybrid Optimization–Simulation (HOS) approach. The goal is to evaluate how each method performs under uncertainty, especially in terms of cost, quality, and service level. The results are divided into three parts. First, the base scenario shows how the HOS model improves through multiple iterations. Then, a comparison is made between HOS, MILP (ideal case), and SBO to show their differences in both financial performance and the network structure. Finally, a sensitivity analysis is conducted to understand how different levels of uncertainty affect the network design and total costs.

A. Base Scenario's Results

This section presents the outcomes from applying three approaches—MILP, SBO, and Hybrid Optimization–Simulation (HOS)—under uncertainty set in the base case, where all uncertain parameters are set to fluctuate with a standard deviation at 7% from their means, normally distributed. Uncertainty is introduced in three key areas: processing time, buffering time, and temperature, all of which directly affect the freshness and quality of perishable products. The evaluation focuses on total cost performance while maintaining product quality and service-level requirements over a simulated 45-day operational period. The cost results and obtained service levels are summarized in Tables V–VII.

In the first iteration of HOS (see Table V), the total cost is \$186,827, comprising \$62,300 in production cost, \$84,527 in transportation cost, \$30,000 in hub fixed cost, and \$10,000 in process fixed cost. However, despite satisfying the mathematical constraints of the MILP model, this solution does not meet the required service levels in the simulation due to variability in process delay and temperature exposure affecting the vase life of flowers (see Tables VI–VII).

In response, the HOS framework is executed iteratively to refine the MILP solution by adjusting the Time-Temperature Sum (TTS) constraints. Over five iterations (see Table V), the total costs gradually increase—from \$186,827 in the first iteration to \$191,547 in the fifth iteration—as the model compensates for uncertainty by selecting more robust hub configurations and adjusting the transportation flows. While production and process costs remained constant across all iterations, changes in hub selection and route assignments led to variations in transportation and fixed hub costs. Notably, the hub fixed cost drops to \$20,000 during Iterations 3 and 4 due to fewer hubs being activated, though this is counterbalanced by higher transportation costs during those iterations.

TABLE V
COST OF THE HYBRID APPROACH IN THE BASE SCENARIO

Value (\$)	Iteration 1	Iteration 2	Iteration 3	Iteration 4	Iteration 5
Total Costs	186,827	188,173	189,645	189,645	191,547
Production Cost	62,300	62,300	62,300	62,300	62,300
Transportation Cost	84,527	85,873	97,345	97,345	89,247
Hub Fixed Cost	30,000	30,000	20,000	20,000	30,000
Process Fixed Cost	10,000	10,000	10,000	10,000	10,000

TABLE VI
ACTUAL VS. REQUIRED SERVICE LEVEL (MONO BOUQUET)

Retailer	Required Service Level (%)	Actual Service Level (%)				
		Iteration 1	Iteration 2	Iteration 3	Iteration 4	Iteration 5
Retailer 1	90%	92.4%	94.6%	95.5%	95.5%	96.8%
Retailer 2	95%	91.6%	93.2%	96.8%	96.8%	97.2%
Retailer 3	90%	92.2%	94.3%	95.1%	95.1%	97.9%
Retailer 4	95%	91.4%	93.1%	95.4%	95.4%	96.1%
Retailer 5	95%	91.8%	92.7%	94.6%	94.6%	95.4%
Retailer 6	95%	92.7%	93.5%	95.7%	95.7%	96.2%
Retailer 7	90%	89.7%	91.6%	93.4%	93.4%	95.4%
Retailer 8	95%	90.1%	91.5%	93.7%	93.7%	95.3%
Retailer 9	90%	92.5%	94.8%	96.1%	96.1%	96.7%
Retailer 10	90%	91.9%	93.5%	96.7%	96.7%	97.6%

Note: Highlighted results show the unsatisfied service level at the retailers

TABLE VII
ACTUAL VS. REQUIRED SERVICE LEVEL (MIXED BOUQUET)

Retailer	Required Service Level (%)	Actual Service Level (%)				
		Iteration 1	Iteration 2	Iteration 3	Iteration 4	Iteration 5
Retailer 1	90%	90.7%	91.2%	93.7%	93.7%	96.4%
Retailer 2	95%	89.4%	92.1%	94.5%	94.5%	96.1%
Retailer 3	90%	90.1%	91.9%	92.2%	92.2%	95.2%
Retailer 4	95%	84.3%	87.9%	92.6%	92.6%	95.5%
Retailer 5	95%	81.2%	85.3%	89.1%	89.1%	95.1%
Retailer 6	95%	83.8%	89.7%	92.5%	92.5%	95.6%
Retailer 7	90%	86.7%	89.4%	91.3%	91.3%	96.1%
Retailer 8	95%	89.8%	91.6%	94.5%	94.5%	96.6%
Retailer 9	90%	91.2%	92.4%	92.7%	92.7%	94.8%
Retailer 10	90%	87.1%	90.2%	93.5%	93.5%	94.7%

Note: Highlighted results show the unsatisfied service level at the retailers

B. Comparison of Hybrid Approach with MILP (ideal case) and SBO as a Benchmark

After analyzing the base scenario using the HOS approach, it is important to compare its performance with other commonly used methods. This helps assess whether the additional complexity and effort of the HOS framework yield better solutions for managing a perishable supply chain. Two other approaches are used for comparison: the MILP model under ideal conditions (without uncertainty) and SBO, which searches for an optimal or near-optimal solution under uncertainty but does not explicitly include perishability constraints. By comparing these three methods, we can better understand the trade-offs between cost, solution quality, and reliability in uncertain environments.

Table VIII shows the total costs and cost components for the MILP (ideal case), SBO, and Hybrid Optimization–Simulation (HOS) approaches. As expected, the MILP model produces the lowest total costs at \$163,418, with the production cost fixed at \$62,300, the transportation cost at \$61,118, the hub fixed cost at \$30,000, and the process fixed cost at \$10,000. However, this solution is created without considering any uncertainty. The SBO approach yields the highest total cost of \$201,172. While the production and process costs remain the same as the other methods, the transportation cost increased to \$98,872. In contrast, the HOS method shows total costs of \$191,547. This solution keeps the production and process costs constant at \$62,300 and \$10,000, respectively, and returns to using three hubs, resulting

in a hub fixed cost of \$30,000. However, the transportation cost increases to \$89,247, reflecting changes made to handle uncertainty while maintaining quality. Overall, Table VIII highlights how each approach allocates costs differently under uncertainty and how HOS offers a more balanced trade-off between cost and quality performance.

Table IX presents the selected hub and process locations from each method. Both the MILP and the HOS solution choose the same configuration: Hubs 1, 2, and 7 are opened, and bouquet-making processes are installed at hubs 1 and 2. This configuration reflects a cost-efficient and quality-aware strategy. As shown in Table IV, the transportation cost for one bouquet is lower than transporting the equivalent volume of unassembled flowers (see Table IV). For example, a mixed bouquet requires multiple flower types, and transporting those individually incurs higher cumulative costs. In contrast, once these flowers are assembled into a single bouquet, they can be shipped as a single unit at a lower rate per bouquet. Therefore, the model favors performing the bouquet-making process early in the network—at hubs 1 and 2—near the growers. This reduces the total transportation cost by avoiding multiple flower shipments to downstream hubs. After assembly, the finished bouquets are sent to hub 7, which acts as a distribution hub. Table IX

indicates that the HOS method can maintain the original structure from the MILP model while still meeting customer requirements under uncertainty, through TTS refinement and simulation feedback. In contrast, the SBO solution activates hub 6 instead of hub 7 to reduce delivery cost and temperature exposure.

Table X shows the computation times from the three methods: MILP (ideal case), Simulation-Based Optimization (SBO), and Hybrid Optimization–Simulation (HOS). The MILP model requires only 5 seconds to compute because it operates under ideal, deterministic conditions and does not account for uncertainty or simulation feedback. In contrast, the SBO approach takes 5,842 seconds due to its reliance on multiple simulation runs and a metaheuristic search process to evaluate solutions under uncertainty. The HOS approach, which integrates optimization and simulation in an iterative feedback loop, requires 3,981 seconds—slightly less than SBO—because it begins with a MILP solution and refines it based on the simulation results. This makes HOS more computationally efficient than SBO while still addressing the uncertainty, offering a balanced trade-off between accuracy and efficiency in designing resilient supply chain networks for perishable products.

TABLE VIII
COST OF MILP, SBO, AND HYBRID APPROACH

Value (\$)	MILP (Ideal Case)	SBO	Hybrid Approach
Total Costs	163,418	201,172	191,547
Production Cost	62,300	62,300	62,300
Transportation Cost	61,118	98,872	89,247
Hub Fixed Cost	30,000	30,000	30,000
Process Fixed Cost	10,000	10,000	10,000

TABLE IX
OPTIMAL HUB AND PROCESS LOCATION OF MILP, SBO, AND HYBRID APPROACH

Location	MILP (Ideal Case)	SBO	Hybrid Approach
Hub Location	1,2,7	1,2,6	1,2,7
Process Location	1,2	1,2	1,2

TABLE X
COMPUTATION TIMES OF MILP, SBO, AND HYBRID APPROACH

Computation Performance	MILP (Ideal Case)	SBO	Hybrid Approach
Computation Time (Seconds)	5	5,842	3,981
Number of Iterations	-	22	5

Note: All models were run on a laptop with 13th Gen Intel(R) Core (TM) i7-13620H (2.40 GHz)

C. Sensitivity Analysis on the Uncertainty

This section looks into how different levels of uncertainty affect the results of the Hybrid Optimization–Simulation (HOS) approach. Five levels of uncertainty are tested: 0%, 3%, 7% (base scenario), 10%, and 15%, which are fluctuated from the means of all processing time, buffering time, and temperature under the normal distribution. Table XI and Fig. 4 show the total costs and cost breakdown at each level. When there is no uncertainty (0%), the total costs are \$163,418. As uncertainty increases, the total costs also increase. At a 3% uncertainty, the cost rises to \$183,873, and at a 7% uncertainty, it increases to \$191,547. When uncertainty reaches 10%, the cost is \$202,878. This increase is mostly due to a higher transportation cost, which goes from \$61,118 at 0% to \$90,578 at 10%. This happens because more effort is needed to keep products fresh and meet delivery times under uncertain conditions. Other cost components also change. The hub fixed cost increases from \$30,000 to \$40,000 when more hubs are opened to reduce travel distance and protect quality. Additionally, the number of iterations required by the HOS approach also increases with the uncertainty, as more feedback cycles between optimization and simulation are needed to adjust the TTS constraints and find a feasible solution that meets quality and service level requirements. At the 15% uncertainty level, the model becomes infeasible due to the absence of any viable routing configuration that can maintain

the TTS within the maximum allowable threshold. This indicates that, under such high variability in processing times, buffering durations, and temperature conditions, the network is unable to ensure product quality standards, as no available path satisfies the TTS constraints required to preserve the minimum vase life at the retailer.

Table XII shows the hub and process locations for each level of uncertainty. For 0%, 3%, and 7% levels, the selected hubs are 1, 2, and 7, with processes at hubs 1 and 2. However, when the uncertainty level rises to 10%, the model adds hub 6 to the network, resulting in four active hubs: 1, 2, 6, and 7, while still keeping the bouquet-making processes limited to hubs 1 and 2. As uncertainty increases, longer processing and buffering times, along with more variable temperatures, lead to higher TTS values, increasing the risk of product deterioration. By adding hub 6, the model reduces transportation distances for certain routes and distributes product flows across more facilities. This shows that as the uncertainty increases, the model adds more hubs to handle the extra delays and risks.

In summary, the HOS model adjusts the supply chain setup to deal with higher uncertainty. When there is more variation in process time, buffering, and temperature, the model changes the network by opening more hubs. These changes help protect product quality and service levels, even though they increase the total costs.

TABLE XI
RESULT OF THE SENSITIVITY ANALYSIS ON THE UNCERTAINTY

Value (\$)	Standard Deviation of Uncertainty				
	0%	3%	7% (base scenario)	10%	15%
Total Costs	163,418	183,873	191,547	202,878	Infeasible
Production Cost	62,300	62,300	62,300	62,300	-
Transportation Cost	61,118	81,573	89,247	90,578	-
Hub Fixed Cost	30,000	30,000	30,000	40,000	-
Process Fixed Cost	10,000	10,000	10,000	10,000	-
Number of Iterations	-	5	5	9	-

TABLE XII
OPTIMAL HUB AND PROCESS LOCATION ON VARIES UNCERTAINTY

Location	Standard Deviation of Uncertainty				
	0%	3%	7% (Base Scenario)	10%	15%
Hub Location	1,2,7	1,2,7	1,2,7	1,2,6,7	Infeasible
Process Location	1,2	1,2	1,2	1,2	Infeasible

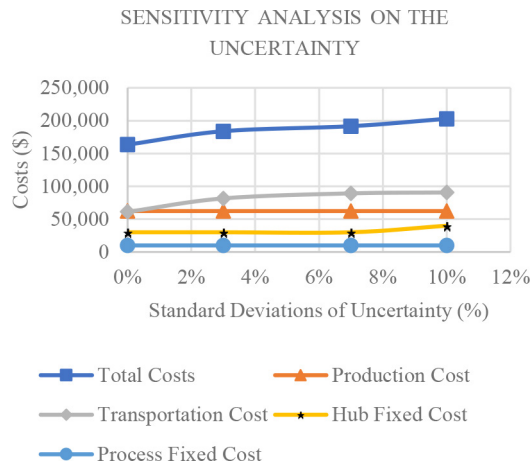


Fig. 4. Sensitivity analysis on the uncertainty

V. MANAGERIAL IMPLICATIONS

This study offered several important insights for supply chain managers handling perishable products such as cut flowers. Firstly, the proposed HOS approach demonstrated how integrating optimization with simulation could produce more resilient and quality-preserving logistics networks under real-world uncertainty. Managers can use this framework to balance cost efficiency with the required service level and product freshness, particularly when traditional optimization models fail to capture dynamic variability in processing time, buffering delays, and temperature fluctuations.

Secondly, the results highlighted the importance of carefully selecting hub locations and process allocations. When uncertainty is within manageable levels, the HOS approach can adjust the network configuration to preserve product quality while minimizing costs. However, when uncertainty exceeds a certain threshold—as observed at the 15% level in the sensitivity analysis—the model fails to find a feasible solution that satisfies quality and service level requirements. This outcome is particularly valuable for managers, as it signals the operational limits of the current supply chain design.

Finally, by comparing the HOS approach with MILP and SBO solutions, managers are better equipped to understand the trade-offs between cost, quality, and operational flexibility. This supports more informed strategic planning in perishable supply chains, especially in environments characterized by high uncertainty and tight quality requirements.

VI. CONCLUSION AND FURTHER STUDY

The study proposed a Hybrid Optimization–Simulation (HOS) approach for designing a logistics network for perishable products, focusing on a fresh floricultural supply chain. The HOS method combined a MILP model, which includes Time-Temperature Sum (TTS) constraints, with a DES model that tests

whether the network can maintain product freshness and meet the service level targets under uncertainty. The findings revealed that while the initial MILP solution could meet cost and quality requirements in a deterministic setting, it failed to maintain service levels when subjected to variability in processing time, buffering delays, and temperature exposure. Through iterative refinement, the HOS approach successfully adjusted TTS constraints and reconfigured network decisions—such as hub selection and product routing—to restore feasibility and ensure product freshness across all retailers. Compared to other methods, the HOS framework demonstrated a more balanced performance. While MILP could provide the lowest cost, it did not include any uncertainty. The SBO method accounted for uncertainty but required more computational effort and iteration cycles. In contrast, the HOS approach leveraged the strengths of both optimization and simulation, producing a feasible and quality-assured solution while maintaining reasonable computational efficiency.

The sensitivity analysis on different levels of uncertainty confirmed that as uncertainty increased, the total costs and complexity of the network also increased. The model responded by opening more hubs to keep the supply chain flexible and protect product freshness. Additionally, higher levels of uncertainty led to an increase in the number of required iterations within the HOS framework, as more adjustment cycles were needed to refine the Time-Temperature Sum (TTS) constraints and meet the quality and service level requirements. This iterative process allowed the model to gradually adapt the network configuration in response to dynamic conditions, demonstrating the capability of the HOS framework to handle uncertainty through repeated optimization and simulation feedback. However, beyond a certain level of uncertainty, the model could no longer generate feasible solutions, signaling the operational limits of the current supply chain design under extreme variability. These findings highlighted the usefulness of HOS for planning perishable supply chains in uncertain environments.

However, the proposed HOS model still has some limitations. It assumed unlimited hub capacity and homogeneous transporters, which might not reflect real-world operational constraints. In addition, the uncertainty in the simulation was only limited to the normal distribution, whereas real perishable supply chains may face non-normal or correlated variations. Furthermore, this framework focused solely on minimizing total costs without explicitly considering other objectives, such as waste reduction, environmental impact, or service responsiveness. Once these objectives are considered simultaneously, the problem becomes a multi-objective optimization problem.

Overall, the HOS framework proved effective in designing resilient supply chain networks that account

for both cost and quality under realistic conditions. It enabled decision-makers to plan logistics strategies that were not only efficient but also responsive to uncertainty. For further study, future work could explore other types of hybrid approaches to improve solution quality and computational performance. For example, combining metaheuristic algorithms with simulation or integrating reinforcement learning with optimization may help the model adapt faster to changing conditions. In addition, incorporating multi-objective optimization—such as minimizing both cost and number of disposed products—can offer a more complete view of trade-offs in decision-making. These improvements would enhance the flexibility and applicability of the framework in real-world perishable supply chains across different industries.

ACKNOWLEDGEMENT

This work was supported by the Faculty's Quota scholarship awarded by Sirindhorn International Institute of Technology, Thammasat University.

REFERENCES

- [1] M. De Keizer, R. Haijema, J. M. Bloemhof, and J. G. A. J. van der Vorst, "Hybrid optimization and simulation to design a logistics network for distributing perishable products," *Comput. Ind. Eng.*, vol. 88, pp. 26-38, Aug. 2015.
- [2] M. Musavi and A. Bozorgi-Amiri, "A multi-objective sustainable hub location-scheduling problem for perishable food supply chain," *Comput. Ind. Eng.*, vol. 113, pp. 766-778, Nov. 2017.
- [3] A. Rong, R. Akkerman, and M. Grunow, "An optimization approach for managing fresh food quality throughout the supply chain," *Int. J. Prod. Econ.*, vol. 131, no. 1, pp. 421-429, Aug. 2011.
- [4] J. Pasha and P. Ghasemi, "A robust possibilistic programming approach for green closed-loop supply chain network design under hybrid uncertainty," *Comput. Ind. Eng.*, vol. 160, no. 107609, Oct. 2021.
- [5] D. Agustina, C. K. M. Lee, and R. Piplani, "Managing perishability in production-distribution planning: A discussion and review," *Flexible Serv. Manuf. J.*, vol. 25, no. 3, pp. 389-413, Sep. 2013.
- [6] F. Mohebalizadehgashti, H. Zolfagharinia, and S. H. Amin, "Designing a green meat supply chain network: A multi-objective approach," *Int. J. Prod. Econ.*, vol. 219, pp. 312-327, Jan. 2020.
- [7] Z. Mohammadi, F. Barzinpour, and E. Teimoury, "A location-inventory model for the sustainable supply chain of perishable products based on pricing and replenishment decisions: A case study," *PLoS One*, vol. 18, no. 7, no. e0288915, Jul. 2023.
- [8] S. Shirzadi, V. Ghezavati, R. Tavakkoli-Moghaddam, and S. Ebrahimnejad, "Developing a green and bipolar fuzzy inventory-routing model in agri-food reverse logistics with postharvest behavior," *Environ. Sci. Pollut. Res.*, vol. 28, no. 38, pp. 53103-53125, Mar. 2021.
- [9] L. Pan, M. Shan, and L. Li, "Optimizing perishable product supply chain network using hybrid metaheuristic algorithms," *Sustainability*, vol. 15, no. 13, no. 10711, Jul. 2023.
- [10] D. Nakandala, H. Lau, and J. Zhang, "Cost-optimization modelling for fresh food quality and transportation," *Ind. Manag. Data Syst.*, vol. 116, no. 3, pp. 564-583, Apr. 2016.
- [11] C. Sel and B. Bilgen, "A mixed integer linear programming model for the design of the capacitated multi-echelon, multi-product closed-loop supply chain," *J. Cleaner Prod.*, vol. 79, pp. 116-135, Aug. 2014.
- [12] Á. J. Ásmundsson, R. J. Linn, R. A. Jacobs, and E. L. Moore, "A practical approach to linking tactical and operational planning in the supply chain," *Decis. Sci.*, vol. 37, no. 4, pp. 559-582, Feb. 2006.
- [13] G. M. Gaukler, "The value of time and temperature history information for perishable food in logistics," *Int. J. Prod. Econ.*, vol. 259, no. 108885, May 2023.
- [14] H. Ding, L. Benyoucef, and X. Xie, "A simulation optimization method for supplier selection and order allocation," *Int. J. Prod. Econ.*, vol. 120, no. 1, pp. 171-182, Mar. 2009.
- [15] H. J. Ko, C. S. Ko, and T. Kim, "A hybrid optimization/simulation approach for a distribution network design of 3PLs," *Comput. Ind. Eng.*, vol. 50, no. 4, pp. 440-449, Aug. 2006.
- [16] M. Abbasian, Z. Sazvar, and M. M. Mohammadisiahroudi, "A hybrid optimization method to design a sustainable resilient supply chain in a perishable food industry," *Environ. Sci. Pollut. Res.*, vol. 30, pp. 6080-6103, Feb. 2023.
- [17] S. -O. Tromp, R. G. M. van der Sman, H. M. Vollebregt, and E. J. Woltering, "On the prediction of the remaining vase life of cut roses," *Postharvest Biol. Technol.*, vol. 70, pp. 42-50, Aug. 2012.



Jirawat Chatchaichalermporn

is currently a student in the Master of Engineering program in Logistics and Supply Chain System Engineering at the School of Manufacturing Systems and Mechanical Engineering, Sirindhorn International Institute of Technology, Thammasat University, Thailand. His research interests are in the area of optimization and supply chain network design.



Navee Chiadamrong

is an Associate Professor from the School of Manufacturing Systems and Mechanical Engineering, Sirindhorn International Institute of Technology, Thammasat University, Thailand, where he teaches and researches in the area of production planning and control methods and supply chain management. He received his MSc in Engineering Business Management from the University of Warwick and his Ph.D. in Manufacturing Engineering and Operations Management from the University of Nottingham, UK. Some of his recent articles have appeared in the International Journal of Production Economics, Computers and Industrial Engineering, Journal of Simulation, and TQM & Business Excellence.

A Study of Road Tracking Immunity for Autonomous Driving

Chuanxiang Bi¹ and Jian Qu^{2*}

^{1,2}Faculty of Engineering and Technology, Panyapiwat Institute of Management,
Nonthaburi, Thailand

E-mail: 6572100057@stu.pim.ac.th, jianqu@pim.ac.th*

Received: August 15, 2025 / Revised: October 12, 2025 / Accepted: November 10, 2025

Abstract—With the advancement of deep learning research, autonomous driving technology has become increasingly mature. However, the accompanying issue is that environmental interference may pose safety hazards for autonomous driving, presenting a significant threat. Therefore, this paper aims to evaluate the resilience of autonomous vehicles to environmental interference by studying the fundamental road-tracking task of autonomous driving. Following the model of real autonomous driving vehicles, we constructed a 1:20-scale intelligent model car, Jetracer, and used it to simulate the road-tracking task of autonomous driving. We designed four different environments, first collecting varying numbers of image data in the original environment for comparative analysis. Then, we selected ResNet18 and ResNet34 as models for training and loaded them onto Jetracer for testing in the four different environments. The experimental results indicate that as the number of images in the dataset increases, the effectiveness of road tracking also gradually improves. Meanwhile, we found that ResNet18 is more suitable as a training model compared to ResNet34. Additionally, Jetracer demonstrates a certain degree of interference resistance, where slight or localized environmental changes do not significantly affect its road tracking performance. However, if there are significant overall environmental changes or new environments are introduced, Jetracer's road-tracking performance is severely impacted.

Index Terms—Deep Learning, Autonomous Driving, Environmental Interference, Road-Tracking, ResNet18, ResNet34, Jetracer

I. INTRODUCTION

Automatic driving [1], [2] technology has been a key area of automotive innovation in recent years. However, intelligent vehicles capable of multitasking face challenges in handling various complex road conditions. Among these challenges, road tracking

is considered a primary task for autonomous driving. Cosmas et al. [3] employed a laser-based approach for road boundary detection, achieving road tracking by spatially processing data from laser sensors. Meanwhile, Kumar et al. [4] and Li and Qu [5] proposed traditional lane-keeping visual methods, detecting lanes through image processing to analyze the deviation of intelligent vehicles from the road, ensuring that vehicles travel on the correct trajectory. In addition, various other methods are available. Qi et al. [6] utilized reinforcement learning to train intelligent vehicles in virtual environments and then tested them in real environments. While experimental results in virtual environments show promising prospects, successfully translating them to real environments faces challenges. Real-world environments are dynamic and constantly changing, influenced by factors such as lighting, shadows, and other sources of noise, which are often insufficiently accounted for in virtual environments.

To address this, Peng et al. [7] proposed an end-to-end road-tracking architecture based on deep reinforcement learning and evaluated this method in an open racing simulator. Although some success has been achieved in road tracking through the above studies, the road tracking task still faces many challenges due to the variable and complex road conditions that the intelligent car is facing, such as artificial graffiti lines on the road or placing obstacles, so in this paper, we train the intelligent car in a real environment that has already been constructed, and then through the intentional change of the real environment, which is divided into six environments to study the smart car's resistance to different road disturbances in order to continuously improve its ability to adapt to various road conditions.

II. LITERATURE REVIEW

Current research indicates that many intelligent vehicles still rely on externally connected computers and cannot independently complete tasks involving deep learning or reinforcement learning, thus limiting the autonomy of artificial intelligence. For instance,

Yunyong and Qu [8] utilized Arduino as the mainboard for the vehicle to execute reinforcement learning tasks. However, due to Arduino's limited performance, it cannot handle complex computational problems, necessitating a Bluetooth connection to a computer as the backend. Arduino, as a microcontroller-based hardware platform, can only execute programs preprogrammed into the chip and is therefore only suitable for simple scenarios. To make intelligent vehicles an independent unit, some researchers have turned to microprocessor-based hardware platforms, such as the Raspberry Pi, which can handle complex applications like network communication and image processing. For example, Du et al. [9] implemented road-tracking for intelligent vehicles using cameras and radar. Although many studies use the Raspberry Pi as a development board, we found that the deep neural networks in these studies typically have fewer layers and simpler architectures. Therefore, in this work, we chose the Jetson Nano as the mainboard to enable intelligent vehicles to perform tasks independently. For example, Karni et al. [10] and Bae et al. [11] have explored the possibilities of using Jetson Nano. Although research using toy cars is cost-effective, it is difficult to replicate real-world driving conditions due to significant differences between toy cars and real vehicles. In Karni's study and that of Lee and Lam [12], researchers combined cameras and ultrasonic sensors. Meanwhile, Du et al. [9] and Zang et al. [13] utilized various cameras and LiDAR. Despite the adoption of multiple sensors in current research, human drivers typically rely only on vision and hearing to operate vehicles. Therefore, this paper selected a 1:20-scale intelligent car and combined it with the Jetson Nano [14] as the mainboard to investigate the road-tracking task for autonomous driving. The intelligent car's steering and driving control capabilities can, to some extent, simulate the driving systems of real vehicles.

The end-to-end control [15], [16] method is a typical approach in the field of vision-based autonomous driving. In this method, vehicles directly collect sensor data on pedestrians, obstacles, and traffic lights, and then uniformly input these signals into a neural network. The output of the neural network system is closely related to vehicle control, guiding the vehicle's next actions. As early as 1989, Pomerleau utilized neural networks to predict steering commands based on simulated road images. In 2016, NVIDIA unveiled its end-to-end deep learning technology for autonomous vehicles. This technology trains Convolutional Neural Networks (CNNs) to directly map the raw pixels from a single front-facing camera to steering commands. The end-to-end control system can optimize all processing steps, thus having wide-ranging application prospects.

In this paper, the intelligent vehicle we used adopts the end-to-end control method.

Convolutional Neural Networks [17] (ConvNets or CNNs) have emerged as one of the most commonly used neural network architectures in autonomous driving, rooted in the concept of Multilayer Perceptrons (MLPs) and inspired by Hubel and Wiesel's studies of the visual cortex in cats. CNNs offer optimization advantages by reducing the number of parameters, lowering model complexity, and mitigating overfitting risks, particularly evident when processing image inputs. CNNs directly ingest image data, eliminating the need for manual feature extraction and data reconstruction processes inherent in traditional recognition algorithms. Specifically adept at handling two-dimensional images, CNNs autonomously extract features such as color, texture, and shape, granting them a significant advantage in two-dimensional image processing and computational efficiency. In autonomous driving, CNNs are widely used for tasks such as road tracking. For instance, Rausch et al. [15] proposed a CNN model comprising three convolutional layers, two pooling layers, and one fully connected layer, whereas Lin et al. [18] presented a more complex model with five convolutional layers and four fully connected layers. These models underwent testing in computer-simulated virtual environments, successfully achieving road tracking, albeit with disparities compared to real-world conditions. Further research utilizes miniature intelligent driving platforms and custom-built tracks for testing. Studies by Bechte et al. [19] and Do et al. [20] also employ similar CNN models, effectively achieving road tracking on custom-built tracks. Consequently, Convolutional Neural Networks play a pivotal role in autonomous driving, and this research provides insights into their key architectural components, proposes end-to-end autonomous driving methodologies, and discusses road-tracking implementation strategies.

III. METHODOLOGY

A. Hardware Designs

This paper selects Jetracer as the research subject to closely simulate real-world driving scenarios. Unlike many toy cars used by researchers, Jetracer features adjustable parameters that enhance vehicle performance. In contrast to toy cars that can only accept binary signals (1 or 0), Jetracer's design permits adjustments to parameters such as steering gain, thereby preventing deviation from the normal trajectory. Unlike other sensors that solely rely on a single camera to perceive the surrounding environment, Jetracer, akin to human drivers, utilizes both vision and hearing.

After comparing three embedded mainboards, Arduino, Raspberry Pi, and Jetson Nano [2], [21], this paper ultimately selects Jetson Nano as the host board for Jetracer. As shown in Fig. 1 and Table I, the Jetson Nano boasts powerful performance and rich functionality, enabling it to connect to cameras and control their operation. The camera serves as the information-perception sensor, responsible for collecting and transmitting data to the Jetson Nano. The control board acts as the command transmission medium, relaying instructions from Jetson Nano to the motors to control the vehicle's propulsion. With this design, we establish an autonomous driving platform using the Jetracer hardware architecture and the hardware framework illustrated in Fig. 2.

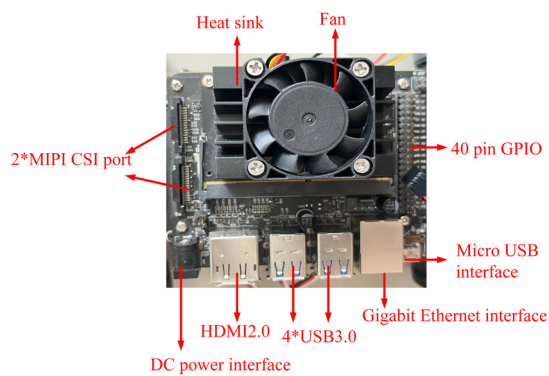


Fig. 1. Jetson Nano construction diagram

TABLE I
JETSON NANO SPECIFIC INFORMATION

Project	NVIDIA Jetson Nano
CPU	Quad-Core ARM Cortex-A57
GPU	NVIDIA Maxwell
Memory	4GB LPDDR4
Networking	Gigabit Ethernet/M.2Key E
Video Encode	H.264/H.265(4Kp30)
Camera	MIPI CSI port
USB	4*USB3.0,USB2.0 Micro-B
Display	HDMI 2.0 and eDPI.4
Other	40-pin GPIO
Storage	Micro-SD

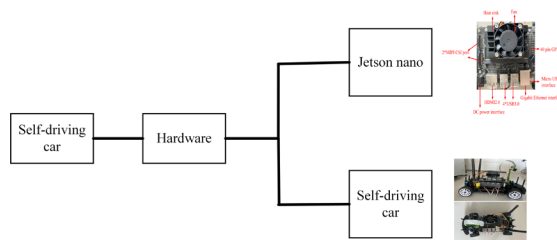


Fig. 2. Jetracer hardware framework diagram

B. Environment Construction

Regarding the driving environment of the intelligent vehicle, we utilize a designed map. Environment 1, as depicted in Fig. 3, is a scenario we designed. It consists of an elliptical track, with yellow lines representing the road edges and white dashed lines representing the lane centerlines. The main objectives of this environment are twofold: data collection and validation of the intelligent vehicle's road-tracking capabilities in the training environment.



Fig. 3. Environment 1

Environment 2 (as shown in Fig. 4): We replaced the white dotted middle lane line with a solid black line based on Environment 1, taking into account that the road environment changes the middle lane line color, to test whether a change in the middle lane line makes the smart car less capable of performing the road tracking task.



Fig. 4. Environment 2

Environment 3 (as shown in Fig. 5): We unfolded two additional curved paths on the white dotted line in the middle into a three-way intersection on one of the road segments in Environment 1, considering whether someone maliciously modified the lane lines, to test whether it would interfere with the normal driving of the smart car.



Fig. 5. Environment 3

Environment 4 (as shown in Fig 6): We built on Environment 1 by paving the road with stickers of various lengths and colors, considering whether a crash occurs when the road is full of obstacles. The aim is to test whether the smart car will not be able to perform the task due to a drastic change in the environment.



Fig. 6. Environment 4

Environment 5 (as shown in Fig. 7): We drew an identical median next to the lane median based on Environment 1, with the aim of testing whether the smart car would sway from side to side due to the two lane medians.



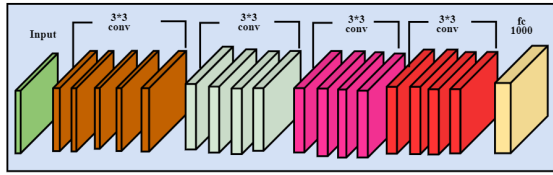
Fig. 7. Environment 5

Environment 6 (as shown in Fig. 8): We change the original lane median to yellow on the basis of Environment 5, with the purpose of testing whether the intelligent vehicle will change its path because the original lane median is changed.



Fig. 8. Environment 6

Regarding the models used in the research, most existing studies have employed simple Convolutional Neural Network (CNN) models. For example, Victor et al. [22], used a 6-layer CNN neural network in their study, while Truong-Dong Do et al. [23] utilized a nine-layer CNN neural network. However, shallow neural networks may encounter two problems: Insufficient feature extraction and the need for large image datasets. However, He et al. proposed ResNet18 in 2016 [24] (as shown in Fig. 9). They significantly increased the network depth by introducing residual connections, thereby alleviating network degradation. ResNet is a deep residual network that effectively avoids the vanishing gradient problem by skipping layers via residual connections, enabling it to train very deep structures. ResNet uses a large number of residual blocks as the base blocks. The design of ResNet18 consists of a 7x7 convolutional layer, followed by a 3x3 max-pooling layer for feature extraction. Subsequently, multiple residual blocks of the same structure are used for feature extraction, each containing two 3x3 convolutional layers and using residual concatenation. ResNet18 contains a total of four composite blocks, with the number of channels in the convolutional layers within each block increasing in the order 64, 64, 128, and 256, and the number of channels varied by cross-scaling the residual blocks. Finally, the average pooling layer and softmax classifier are picked up. ResNet18 has fewer but more significant parameters and possesses strong representation-learning capability. At the same time, the computational overhead is relatively small and well-suited to the task at hand. Therefore, we chose ResNet18 as our model.



ResNet18

Fig. 9. ResNet18 construction diagram

IV. EXPERIMENTAL SETUP

A. Dataset Collection

Data collection is a crucial step in intelligent vehicle research. We utilize the intelligent vehicle to traverse along the map and capture images for collecting image data. To ensure data quality, we close curtains and doors to maintain stable lighting conditions, preventing sunlight from affecting the environment and reducing sunlight noise. All data are collected and tested under such lighting conditions. The neural networks used in this paper take images with a resolution of 224x224 pixels as input.

B. Model Training

During the model training phase, we leverage the Colab platform's powerful computational resources to train efficiently. By configuring an environment that meets deep learning requirements, we train the model on the Colab platform. The environment is configured as in Table II. We divide the collected dataset into two parts: the training set and the test set. The training set comprises 80% of the dataset and is used to train the model, while the test set comprises 20% and is used to validate the model's accuracy.

TABLE II
ENVIRONMENT CONFIGURATION

Environment Configuration				
Request	TensorRT	Pytorch	CUDA	CUDNN
Version	6.0.1.10	1.4.0	10.0.326	7.6.3.28

C. Loss functions

Loss functions are used in deep learning to measure the difference between a model's predictions and the actual values. Since this paper implements a simple road-tracking task, it uses an existing cross-entropy loss function rather than designing a new one. In this paper, we choose the Mean Squared Error (MSE). MSE is a common loss function because it requires no parameters, is computationally efficient, and has a clear physical meaning. Its formula is shown below.

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

We employed a consistent hyperparameter configuration scheme for training all models. Considering the memory constraints of the embedded

device's mainboard, we selected appropriate mini-batch sizes and mini-epochs within the Jetson Nano's memory allowance for model training. Here, an epoch refers to the number of times all data are processed during training. Our study aimed to identify the minimum loss value and the most efficient training epochs. We implemented an early stopping strategy during model training. When the training loss ceases to decrease, we conduct an additional five training iterations. If the training loss does not decrease after these five iterations, training is automatically terminated. Furthermore, we opted for the widely adopted Adam optimizer due to its simplicity, computational efficiency, and relatively modest memory requirements.

In our experiments, the model car's speed is fixed. This simplifies the control logic, which only needs to adjust the steering angle. During training, the model learns to keep the vehicle in the center of the road by predicting the appropriate steering angle based on the input image. The trained model is deployed to an autonomous driving platform and tested and validated in a real-world environment. The steering angle output from the model will be used to control the vehicle's direction when the vehicle speed is fixed. By processing camera input images in real time, the model can continuously output the steering angle to maintain road tracking and drive steadily.

D. Experiment

In Environment 1, we captured 400, 600, 800, and 1000 images for road tracking using the intelligent vehicle, naming these four datasets Dataset A, Dataset B, Dataset C, and Dataset D. We trained ResNet18 models separately on each dataset. Subsequently, we employed the trained models to perform road-tracking tasks in Environments 1, 2, 3, and 4, recording data to assess the models' resistance to interference. Due to the limited number of images in the collected dataset, setting the epoch count too low may result in underfitting, while setting it too high may lead to overfitting. Therefore, we conducted experiments with several parameter combinations and determined to utilize an epoch of 40 and a batch size of 16 based on our evaluations. As shown in Table III, subsequent training sessions were also conducted using this configuration.

TABLE III
HYPERPARAMETER SETTING

Hyperparameter		
Optimizer	Batch Size	Epoch
Adam	16	40

For comparison, we selected ResNet34 as another model. Similarly, we trained ResNet34 models with the aforementioned four datasets individually and

utilized the trained models to execute road-tracking tasks in Environments 1, 2, 3, and 4. This was done to compare the road-tracking performance with the previous ResNet18 models. A diagram of the experimental framework is shown in Fig. 10.

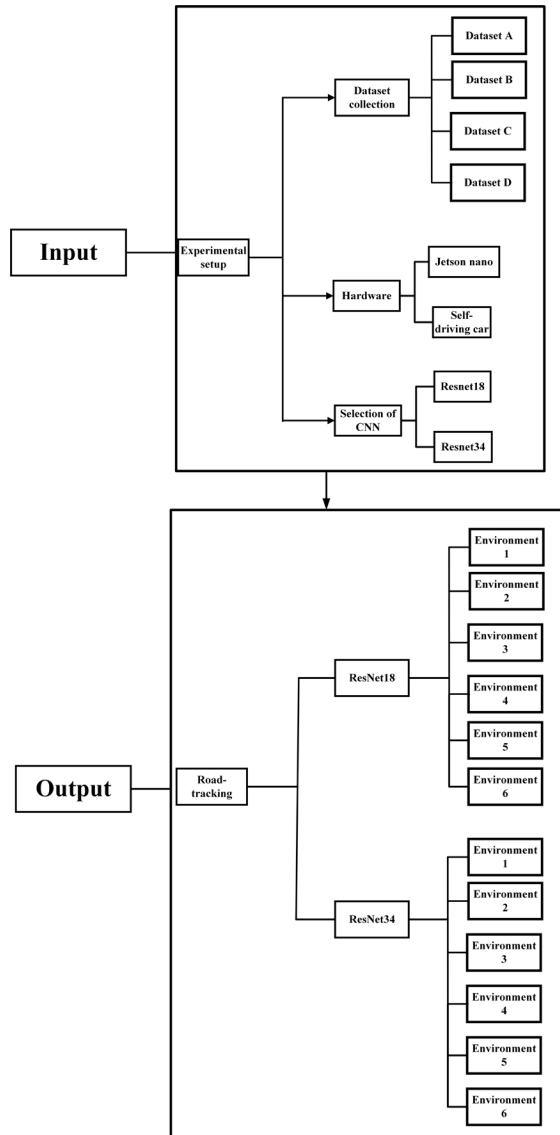


Fig. 10. Experimental framework diagram

V. RESULT

A. Evaluation Methodology

To evaluate the intelligent vehicle's road-tracking performance, we propose a new method for assessing autonomous driving. The vehicle drives five laps in the environment, with each completed lap earning 2 points, for a total score of 10. The scoring rules are as follows:

(1) In each lap, if the intelligent vehicle touches the inner or outer lane line once, one point is deducted; if it touches more than twice, the score for that lap is zero.

(2) If the intelligent vehicle runs off the map boundary during a lap, the score for that lap is zero, and subsequent lap scores are also zero.

The total score can be calculated by adding up the scores from the five laps.

B. Discussions

Based on the data in Table IV, we observe the following: as the number of images in the dataset increases, the intelligent vehicle's score in Environment 1 gradually improves. This indicates that as data volume increases, the intelligent vehicle's performance in road-tracking tasks becomes more stable. Environment 2 has a relatively minor impact on the intelligent vehicle's road tracking, but it does interfere with the vehicle's travel, making it more prone to touching the inner and outer lane lines. The influence of Environment 3 on the road-tracking of the intelligent vehicle is almost negligible, indicating that changing the center lane line has minimal impact on the vehicle. In Environment 4, the intelligent vehicle's score is the lowest, indicating that its performance on road-tracking tasks is the worst. In most cases, the intelligent vehicle is directly disturbed and runs off the map, making it unable to continue the task. This suggests that creating large-scale disturbances on the ground can cause autonomous driving to fail at road-tracking tasks.

TABLE IV
TABLE ON ROAD-TRACKING SCORES FOR EACH OF
THE FOUR MODELS OF RESNET18 FOR EACH OF
THE FOUR ENVIRONMENTS

Dataset	Quantities	Model	Environment	score
Dataset A	400	ResNet18	Environment 1	5
			Environment 2	3
			Environment 3	4
			Environment 4	0
			Environment 5	6
			Environment 6	2
Dataset B	600	ResNet18	Environment 1	7
			Environment 2	5
			Environment 3	6
			Environment 4	0
			Environment 5	7
			Environment 6	4
Dataset C	800	ResNet18	Environment 1	10
			Environment 2	8
			Environment 3	9
			Environment 4	2
			Environment 5	10
			Environment 6	6
Dataset D	1000	ResNet18	Environment 1	10
			Environment 2	9
			Environment 3	10
			Environment 4	2
			Environment 5	10
			Environment 6	6

Based on the training and testing results of the ResNet34 model, compared with the previous use of the ResNet18 model, as shown in Table II, we found that the road tracking scores across all environments were lower than those of the ResNet18 model. Since ResNet34 is more complex than ResNet18 and contains more layers. However, in the simple road-tracking task in this paper, this extra complexity does not improve performance and may even lead to overfitting. In this case, it may not lead to a significant performance improvement; instead, it may cause training difficulties and reduced generalisation due to an overparameterised model. Overfitting occurs when the model performs well on training data but poorly on test data. Whereas the simpler model, ResNet18, may be sufficient to express the information needed for the task. Suggesting that in a specific task (road tracking), ResNet18 may perform better due to its lower complexity, faster inference, and better generalization.

This suggests that ResNet18 is more suitable than ResNet34 for this autonomous driving task. However, as with the ResNet18 model, the scores for environments 2 and 3 did not differ much from those for environment 1. This suggests that these environments have relatively little effect on the performance of the ResNet34 model. In contrast, in environment 4, the ResNet34 model also exhibits a lower score profile, indicating that this environment has a greater impact on the road-tracking task, consistent with previous observations.

TABLE V
THE TABLE ON ROAD-TRACKING SCORES FOR EACH OF
THE FOUR MODELS OF RESNET34 FOR EACH OF
THE FOUR ENVIRONMENTS

Dataset	Quantities	Model	Environment	Score
Dataset A	400	ResNet34	Environment 1	3
			Environment 2	0
			Environment 3	2
			Environment 4	0
			Environment 5	3
			Environment 6	1
Dataset B	600	ResNet34	Environment 1	6
			Environment 2	4
			Environment 3	6
			Environment 4	0
			Environment 5	8
			Environment 6	4
Dataset C	800	ResNet34	Environment 1	8
			Environment 2	5
			Environment 3	6
			Environment 4	1
			Environment 5	8
			Environment 6	6
Dataset D	1000	ResNet34	Environment 1	10
			Environment 2	7
			Environment 3	10
			Environment 4	2
			Environment 5	10
			Environment 6	7

Regarding the reasons why the experimental model car above crashed into the outer or inner lane, we conclude the following.

(1) In terms of image processing, noise in the image (shadows, halos) may interfere with the model's judgment, leading to incorrect lane tracking. Low-resolution images may also fail to capture details, resulting in the model's inability to accurately identify lane lines and other road features.

(2) In terms of environmental factors and external interferences, this paper sets up road interferences, such as lane lines of other colours and obstacles of different lengths on the road, and the training data does not contain sufficiently diverse map data, which may prevent the model from accurately identifying them.

(3) In terms of hardware processing capability, the model car's ability to process images in real time is not very strong, and processing delays may lead to incorrect judgments.

(4) In terms of sensor and hardware limitations, a single sensor (e.g., using only a camera) may not be sufficient to cope with complex environments, and a combination of multiple sensors is needed to improve accuracy.

VI. CONCLUSION

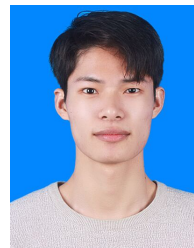
Achieving road tracking is the most fundamental and critical task in autonomous driving, and it is also the focus of the current experiments in this paper. By establishing an autonomous driving platform - Jetracer, to replace real autonomous driving vehicles, we have successfully conducted research on the road tracking task. Our experimental results demonstrate that the intelligent vehicle can perform road-tracking tasks using a single camera. In addition, we conducted three experiments: first, varying the degree of environmental interference; second, comparing the performance of different models; and third, analysing the impact of the number of images in the dataset on road-tracking effectiveness. The results show that the effectiveness of road tracking improves as the number of images in the training dataset increases. However, once the number of images reaches a certain threshold, further increasing the dataset size does not significantly improve effectiveness. Furthermore, the autonomous driving vehicle exhibits good anti-interference capability against minor environmental disturbances, but when facing severe environmental interference, its road-tracking effectiveness is significantly degraded. Finally, by comparing the performance of two models, we conclude that ResNet18 outperforms ResNet34 in road-tracking tasks and is better suited to autonomous driving.

ACKNOWLEDGMENT

The first author received scholarship support from CP ALL to conduct this research at PIM. The last author is the corresponding author.

REFERENCES

- [1] J. Qu, "Multi-task in autonomous driving through RDNet18-CA with LiSHTL-S loss function," *ECTI Trans. Comput. Inf. Technol.*, vol. 18, no. 2, pp. 158-173, Apr. 2024.
- [2] S. Ding and J. Qu, "Research on multi-tasking smart cars based on autonomous driving systems," *SN Comput. Sci.*, vol. 4, no. 3, p. 292, Mar. 2023.
- [3] K. Tsiakas, I. Kostavelis, D. Giakoumis, and D. Tzovaras, "Road tracking in semi-structured environments using spatial distribution of LiDAR data," in *Artificial Intelligence Applications and Innovations*, Cham, Switzerland: Springer, 2021, ch. 32, pp. 432-445.
- [4] A. Kumar, T. Saini, P. B. Pandey, and A. Agarwal, "Vision-based outdoor navigation of self-driving car using lane detection," *Int. J. Inf. Technol.*, vol. 14, no. 1, pp. 215-227, Aug. 2022.
- [5] Y. Li and J. Qu, "Intelligent road tracking and real-time acceleration-deceleration for autonomous driving using modified convolutional neural networks," *Curr. Appl. Sci. Technol.*, vol. 22, no. 6, pp. 1-26, Mar. 2022.
- [6] Q. Zhang, T. Du, and C. Tian, "Self-driving scale car trained by deep reinforcement learning," *arXiv*, 2019. [Online]. Available: <https://arxiv.org/abs/1909.03467> [Accessed: Aug. 15, 2025].
- [7] B. Peng et al., "End-to-end omous driving through dueling double deep Q-network," *Automot. Innov.*, vol. 4, pp. 328-337, Jun. 2021.
- [8] S. Yuenyong and Q. Jian, "Generating synthetic training images for deep reinforcement learning of a mobile robot," *J. Intell. Inform. Smart Technol.*, vol. 2, no. 3, pp. 16-20, 2017.
- [9] X. Du, M. H. Ang, and D. Rus, "Car detection for autonomous vehicle: LiDAR and vision fusion approach through deep learning framework," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 749-754.
- [10] U. Karni, S. S. Ramachandran, K. Sivaraman, and A. K. Veeragahaven, "Development of autonomous downscaled model car using neural networks and machine learning," in *Proc. Int. Conf. Comput. Methodol. Commun. (ICCMC)*, 2019, pp. 1089-1094.
- [11] Y. Bae et al., "Prototyping a system of cost-effective autonomous guided vehicles," in *Proc. Annu. Gen. Donald R. Keith Memorial Conf.*, 2021, pp. 138-143.
- [12] K. L. Lee and H. Y. Lam, "Development of deep learning autonomous car using Raspberry Pi," *Prog. Eng. Appl. Technol.*, vol. 2, no. 1, pp. 534-548, Jun. 2021.
- [13] S. Zang et al., "The impact of adverse weather conditions on autonomous vehicles: How rain, snow, fog, and hail affect the performance of a self-driving car," *IEEE Veh. Technol. Mag.*, vol. 14, no. 2, pp. 103-111, Mar. 2019.
- [14] S. Valladares et al., "Design and implementation of an autonomous vehicle prototype," in *Proc. 4th Int. Conf. Intell. Human Syst. Integr. (IHSI)*, 2021, pp. 343-349.
- [15] V. Rausch et al., "Learning a deep neural net policy for end-to-end control of autonomous vehicles," in *Proc. Am. Control Conf. (ACC)*, 2017, pp. 491-4919.
- [16] Z. Chen and X. Huang, "End-to-end learning for lane keeping of self-driving cars," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, 2017, pp. 1856-1860.
- [17] C. J. Willmott and K. Matsuura, "Advantages of the Mean Absolute Error (MAE) over the Root Mean Square Error (RMSE) in assessing average model performance," *Clim. Res.*, vol. 30, no. 1, pp. 79-82, 2005.
- [18] W. -Y. Lin, W. -H. Hsu, and Y. -Y. Chiang, "A combination of feedback control and vision-based deep learning mechanism for guiding self-driving cars," in *Proc. IEEE Int. Conf. Artif. Intell. Virtual Reality (AIVR)*, 2018, pp. 262-266.
- [19] M. G. Bechtel et al., "Deeppicar: A low-cost deep neural network-based autonomous car," in *Proc. IEEE Int. Conf. Embedded Real-Time Comput. Syst. Appl. (RTCSA)*, 2018, pp. 11-21.
- [20] T. -D. Do et al., "Real-time self-driving car navigation using deep neural network," in *Proc. Int. Conf. Green Technol. Sustain. Dev. (GTSD)*, 2018, pp. 7-12.
- [21] Z. Nie and J. Qu, "Multi-task autonomous driving based on improved convolutional neural network and ST loss in MTS and MOD modes," *Curr. Appl. Sci. Technol.*, vol. 23, no. 3, pp. 1-26, Jun. 2023.
- [22] B. Goldfain et al., "Autorially: An open platform for aggressive autonomous driving," *IEEE Control Syst. Mag.*, vol. 39, no. 1, pp. 26-55, Jun. 2019.
- [23] S. Mehta, C. Paunwala, and B. Vaidya, "CNN based traffic sign classification using Adam optimizer," in *Proc. Int. Conf. Intell. Comput. Control Syst. (ICCS)*, 2019, pp. 1293-1298.
- [24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 770-778.



Chuanxiang Bi is currently pursuing the Master of Engineering Technology at the Faculty of Engineering and Technology, Panyapiwat Institute of Management, Thailand. He received B.B.A from Nanjing Tech University Pujiang Institute, China, in 2022. His research interests are artificial intelligence, image processing, and autonomous driving.



Jian Qu is a full-time lecturer at the Faculty of Engineering and Technology, Panyapiwat Institute of Management. He received a Ph.D. with an Outstanding Performance award from the Japan Advanced Institute of Science and Technology, Japan, in 2013. He received B.B.A with Summa Cum Laude honours from the Institute of International Studies of Ramkhamhaeng University, Thailand, in 2006, and M.S.I.T from Sirindhorn International Institute of Technology, Thammasat University, Thailand, in 2010. His research interests are natural language processing, intelligent algorithms, machine learning, machine translation, information retrieval, and image processing.

A Review on DonkeyCar Autonomous Driving Platform Using Neural Network

Yangyang Li¹ and Jian Qu^{2*}

^{1,2}Faculty of Engineering and Technology, Panyapiwat Institute of Management, Nonthaburi, Thailand

E-mail: 6672100032@stu.pim.ac.th, jianqu@pim.ac.th*

Received: September 26, 2024 / Revised: March 23, 2025 / Accepted: May 28, 2025

Abstract—DonkeyCar is a DIY autonomous vehicle platform for exploring AI-driven self-driving technology. Using the DonkeyCar Unity simulator, models can be trained and tested in a risk-free environment before real-world deployment. This study examines the impact of training data and track complexity on model performance. Results show that increasing the number of training images from 1,346 to 5,351 reduced the error rate from 36% to 3%, thereby improving driving stability. Models trained on complex tracks adapted better but required longer training. However, challenges such as sensor noise, environmental uncertainties, and limitations in obstacle avoidance suggest the need for data augmentation, reinforcement learning, and multi-sensor fusion to enhance model robustness. These findings contribute to optimizing low-cost, vision-based autonomous driving systems.

Index Terms—Autonomous Driving, Deep Learning

I. INTRODUCTION

DonkeyCar is an open-source DIY platform designed for developing and training small-scale autonomous vehicles, primarily using RC cars as a base. It seamlessly integrates hardware and software, providing a user-friendly environment for experimenting with machine learning and autonomous driving technologies [1]. Whether for beginners or experienced developers, DonkeyCar offers a flexible framework for exploring autonomous vehicle development.

At the core of DonkeyCar is autonomous driving powered by machine learning? The development process begins with a user manually controlling the vehicle to collect driving data, including images and corresponding steering/acceleration commands. This dataset is then used to train a neural network model, which learns to predict the vehicle's control actions in unseen environments, ultimately enabling autonomous navigation [2].

Deep learning, a subfield of machine learning, relies on artificial neural networks to simulate the human brain's cognitive processes. It is particularly effective in handling unstructured data such as images and speech. Through multi-layer neural networks (deep networks), computers can automatically extract meaningful patterns and features from large datasets. In the DonkeyCar project [3], deep learning plays a crucial role in visual perception and decision-making. The vehicle captures road images through an onboard camera, which are then fed into a trained neural network. The model processes these images to determine appropriate steering and acceleration commands, facilitating autonomous driving.

Currently, most DonkeyCar projects rely primarily on cameras rather than additional sensors for three key reasons:

A. Simple and Efficient Visual Perception

Cameras provide a straightforward and effective means of capturing detailed environmental information, offering a direct representation of the car's surroundings. Visual perception is widely used in autonomous driving for its ability to detect lane markings, road signs, obstacles, and curves. Compared to other sensors, camera data is more intuitive and integrates seamlessly with deep learning models for real-time decision-making.

B. Low Cost and Accessibility

Cameras are cost-effective and widely available, making them an economical choice for DIY projects. Compared to expensive LIDAR or high-precision radar sensors, cameras provide a budget-friendly alternative that is familiar and easy to configure. Additionally, hardware platforms such as Raspberry Pi natively support camera modules, further simplifying system setup and reducing overall development costs [4].

C. Optimized for Deep Learning Models

The DonkeyCar platform heavily relies on deep learning models, particularly Convolutional Neural Networks (CNNs), which excel at processing visual

data. Cameras naturally complement CNN-based architectures, allowing for efficient end-to-end learning of driving behaviors. This approach streamlines the development process, enabling users to focus on training and optimizing image-based models without the added complexity of fusing data from multiple sensors [5].

D. Research Objective

This study aims to evaluate the impact of training data size and track complexity on the performance of vision-based autonomous driving models using DonkeyCar. By analyzing model accuracy, error rates, and generalization capabilities in different simulated environments, this work provides insights into optimizing low-cost, camera-based self-driving platforms. The findings contribute to improving autonomous navigation in real-world scenarios and advancing the development of AI-driven vehicle control systems.

II. RESEARCH OBJECTIVE

The primary objective of this research is to develop an autonomous vehicle following system based on the DonkeyCar platform. Specifically, this study aims to enhance vehicle

Perception and decision-making using deep learning-based vision systems. By leveraging Convolutional Neural Networks (CNNs), the proposed system will enable a trailing vehicle to detect and follow a leading vehicle in real-time, without relying on additional sensors such as LiDAR or radar.

To achieve this, the study will focus on the following key objectives:

(1) Data Collection and Model Training:

Develop a dataset by capturing real-time images from the onboard camera while manually driving the leading vehicle.

Train a deep learning model capable of recognizing the leading vehicle and estimating its position relative to the trailing vehicle.

(2) Autonomous Following Algorithm:

Implement a real-time object detection mechanism to identify the leading vehicle using CNN-based models.

Design a control strategy that allows the trailing vehicle to adjust its speed and steering angle based on the detected position of the leading vehicle.

(3) Simulation and Real-World Validation:

Test the proposed system in the DonkeyCar simulator to evaluate its performance in a controlled virtual environment.

Deploy the model on a physical DonkeyCar and conduct real-world experiments to assess robustness and adaptability under varying conditions.

The expected outcome of this research is a vision-based vehicle following system that operates

effectively in both simulated and real-world environments. The findings will help advance low-cost, camera-based autonomous driving solutions and provide valuable insights for future developments in small-scale autonomous vehicles [6].

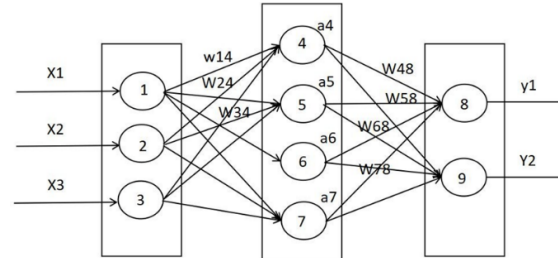


Fig. 1. Fully connected neural network (DNN)

III. RELATED WORKS

Autonomous driving has been a highly active research area, leveraging various Artificial Intelligence (AI) techniques to enhance perception, decision-making, and control. In particular, deep learning and Reinforcement Learning (RL) have demonstrated remarkable capabilities in enabling self-driving vehicles to navigate complex environments with minimal human intervention. In this section, we review the role of deep learning in autonomous driving, the application of reinforcement learning for adaptation and control, and the training workflow within DonkeyCar simulations.

A. Perception in Autonomous Driving

Perception is a critical component of autonomous driving, enabling vehicles to sense and interpret their surroundings. Deep learning models, particularly Convolutional Neural Networks (CNNs), have become the cornerstone of object detection, lane recognition, and traffic signal interpretation [7]. CNNs extract hierarchical spatial features from images, allowing vehicles to distinguish pedestrians, traffic lights, and other road elements.

Advanced object detection architectures, such as YOLO (You Only Look Once) and Faster R-CNN, have significantly improved real-time detection performance, making them suitable for embedded systems in autonomous vehicles [8]. These models help identify obstacles and make split-second navigation decisions, a crucial aspect of real-world deployment.

B. Deep Learning for Control and Decision Making

Deep learning is not only essential for perception but also plays a significant role in vehicle control. End-to-end learning approaches, where Deep Neural Networks (DNNs) map raw sensor inputs directly to steering commands, have shown promising results [9]. For instance, NVIDIA's DAVE-2 system trained a deep network using human driving data, successfully replicating human-like driving behaviors [10].

However, end-to-end learning often suffers from limited interpretability and robustness issues, making hybrid approaches more favorable. A combination of deep learning with traditional rule-based or model-based planning techniques provides a more reliable and scalable framework for autonomous navigation.

C. Reinforcement Learning for Adaptive Driving

Reinforcement Learning (RL) has emerged as a powerful paradigm for training self-driving cars in dynamic and uncertain environments. Unlike supervised learning, where models learn from labeled data, RL enables agents to learn optimal policies through trial and error, receiving rewards for desirable actions [11].

Proximal Policy Optimization (PPO) and Deep Q-Networks (DQN) are widely used RL algorithms for autonomous driving [12]. These methods allow self-driving cars to refine their control strategies, adapt to varying road conditions, and optimize fuel efficiency.

D. DonkeyCar Training Workflow

DonkeyCar provides a lightweight and accessible platform for training autonomous driving models in simulation before deploying them on real-world robotic vehicles. The training pipeline follows a structured workflow:

(1) Data Collection:

The DonkeyCar vehicle is manually driven in a simulated environment to collect training images and sensor data.

(2) Preprocessing and Augmentation:

The collected data is cleaned and augmented to enhance model robustness. Techniques such as brightness adjustment, cropping, and flipping help the model generalize better.

(3) Model Training:

A CNN-based model is trained to map images to steering and throttle commands. This model learns to navigate autonomously by imitating human driving patterns.

(4) Evaluation and Fine-Tuning:

The trained model is tested in the simulator to evaluate its performance. If necessary, additional training data is collected, and hyperparameters are tuned to improve accuracy.

(5) Deployment:

Once the model achieves satisfactory performance, it is deployed on a physical DonkeyCar, allowing it to navigate autonomously in real-world scenarios.

Empirical results indicate that increasing the number of training samples significantly enhances model accuracy and driving stability. As shown in Table I, a model trained on 5,351 images exhibits a

considerably lower error rate than one trained on only 1,346 images [13].

TABLE I
MODEL COMPARISON

Model	Training Images	Collisions	Error Rate
Model 1	1,346	36	36%
Model 2	5,351	3	3%

E. Challenges and Future Directions

Despite significant advancements, DonkeyCar-based autonomous driving still faces several challenges:

- **Limited Computational Power:**

The embedded systems used in DonkeyCar have lower processing capabilities than those of full-scale autonomous vehicles, limiting the complexity of deployable models.

- **Generalization to Real-World Scenarios:**

While models trained in simulation perform well within controlled environments, their transferability to real-world conditions remains an open challenge.

- **Automated Labeling and Data Augmentation:**

Current training methodologies rely heavily on manually collected data. Future work could explore automated labeling techniques and synthetic data generation to improve scalability.

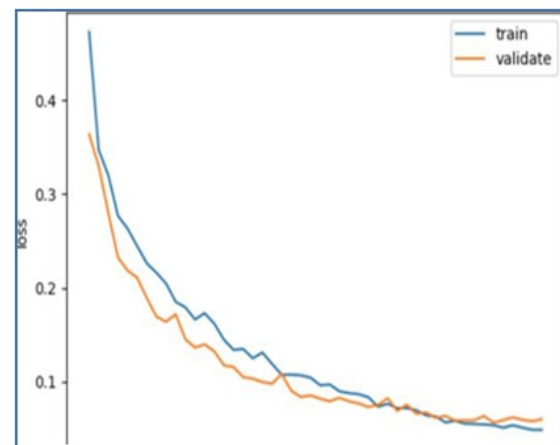


Fig. 2. Model loss picture

As illustrated in the figure above, the blue line represents the model's training curve, while the validation curve serves as a comparison. The close alignment between the two curves suggests that the model generalizes well. The DonkeyCar simulator was used for training [14], and, owing to the extensive dataset collected, the model exhibits strong autonomous driving capabilities with minimal lane deviations.

part	max	min	avg	50%	90%	99%	99.9%
DonkeyGymEnv	0.89	0.01	0.03	0.03	0.04	0.08	0.19
LocalWebController	3.45	0.01	0.09	0.04	0.18	0.75	2.48
Pipe	0.32	0.00	0.01	0.01	0.02	0.04	0.07
ExplodeDict	0.19	0.00	0.01	0.01	0.01	0.03	0.07
ThrottleFilter	0.36	0.00	0.01	0.01	0.01	0.03	0.07
UserPilotCondition	0.38	0.00	0.01	0.01	0.02	0.04	0.09
RecordTracker	4.91	0.00	0.02	0.01	0.01	0.24	1.00
AltLaunch	0.41	0.00	0.01	0.01	0.01	0.03	0.08
DriveMode	0.30	0.00	0.01	0.01	0.02	0.04	0.10
ToggleRecording	6.28	0.00	0.10	0.01	0.34	1.67	4.07
TubWriter	43.94	0.58	4.61	4.48	7.00	13.78	40.29

Fig. 3. Training model diagram

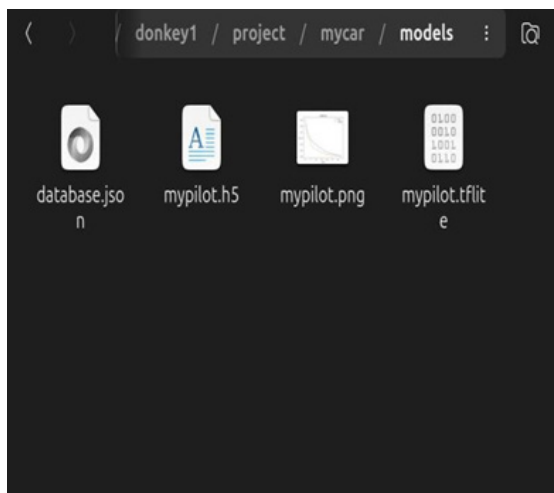


Fig. 4. Training model diagram

After training with DonkeyCar, a series of essential files are generated, including mypilot.h5, database.json, mypilot.png, and mypilot.tflite. These files collectively form the core data of the model, ensuring its usability, reproducibility, and smooth deployment.

- **mypilot.h5 File:** This is the primary output file from DonkeyCar training, which stores the neural network's trained weights. It records how the model learns to associate input images from the car's camera with user-provided control signals (steering and throttle). For example, when enabling autonomous driving, DonkeyCar loads the mypilot.h5 file, allowing the vehicle to drive autonomously based on the training data [15].

- **database.json File:** This file stores metadata about the trained model, such as network architecture, training parameters, vehicle information, speed, model name, and local control records. It plays a crucial role in reloading the model, inspecting its configuration, or further training. Additionally, this file describes

the neural network architecture used for training, including the number of layers, activation functions, and other custom components, ensuring that the model structure can be accurately recreated for inference or retraining [16].

- **mypilot.tflite File:**

This is a TensorFlow Lite model optimized for lightweight inference. It is particularly useful for deployment on resource-constrained devices like Raspberry Pi, enabling efficient execution of the trained model on low-power hardware.

- **mypilot.png File:**

This file typically contains visualized data generated during training, such as the model's loss curve or driving trajectory, helping analyze and optimize training performance.

These files collectively form the complete model output after DonkeyCar training, enabling easy reloading, inspection, and efficient deployment across different devices, ensuring the reproducibility and reliability of the trained model [17].

```

/home/tyangyang/mhcondas/envs/donkey/lib/python3.11/site-packages/keras/src/engine/training.py:3103: UserWarning: You are saving your model as an HDF5 file via 'model.save()'. This file format is considered legacy. We recommend using instead the native Keras format, e.g. 'model.save('my_model.keras')'.
  saving_api.save_model(
12/12 [=====] - 12s 827ms/step - loss: 0.3681 - n_outputs0_loss: 0.3142 - n_outputs1_loss: 0.0539 - val_loss: 0.3484 - val_n_outputs0_loss: 0.3098 - val_n_outputs1_loss: 0.0387
Epoch 2/100
1/12 [=>.....] - ETA: 6s - loss: 0.3105 - n_outputs0_loss: 0.2683 - n_outputs1_loss: 0.03283
2/12 [====>...] - ETA: 6s - loss: 0.3283 - n_outputs0_loss: 0.2830 - n_outputs1_loss: 0.03330
3/12 [=====>...] - ETA: 6s - loss: 0.3330 - n_outputs0_loss: 0.2776 - n_outputs1_loss: 0.03269
4/12 [=====] - ETA: 5s - loss: 0.3276 - n_outputs0_loss: 0.3269 - n_outputs1_loss: 0.03313
5/12 [=====] - ETA: 5s - loss: 0.3269 - n_outputs0_loss: 0.3313 - n_outputs1_loss: 0.03330
6/12 [=====] - ETA: 4s - loss: 0.3313 - n_outputs0_loss: 0.3330 - n_outputs1_loss: 0.03330
7/12 [=====] - ETA: 3s - loss: 0.3330 - n_outputs0_loss: 0.3330 - n_outputs1_loss: 0.03330

```

Fig. 5. Train Model

Formula:

$$MSE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

Description: Mean Squared Error (MSE) is the most commonly used loss function in DonkeyCar for regression tasks. It measures the average squared difference between predicted and actual values, penalizing larger errors more heavily [18].

Use Case: MSE is ideal for predicting continuous values, such as steering angles and throttle, to ensure smooth and precise vehicle control.

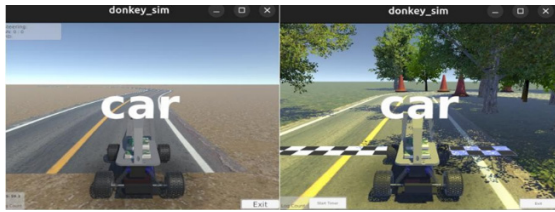


Fig. 6. Comparison of two tracks

Fig. 6 illustrates two distinct tracks within the simulator: **generated_track** and **generated_road**. This experiment aims to analyze differences in model training across the two tracks, assess the impact of training data on model performance, and explore optimization strategies to enhance the trained model's effectiveness for autonomous driving.

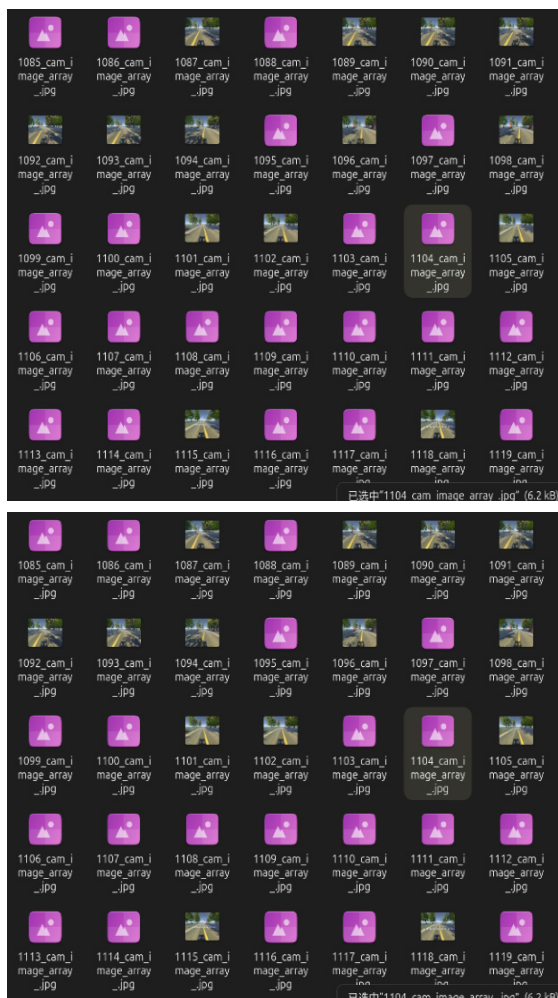


Fig. 7. Generated_road data

The two images above show the second map used for training, **generated_track**. Initially, 1,346 images were used, and the results showed 36 collisions, with the model unable to complete a full lap. This indicates insufficient data, shorter training time, and a more complex map with more obstacles.

As shown in Fig. 10, when the data increased to 5,351 images, the model's performance improved significantly. It was able to complete autonomous driving on the main road, and the error rate dropped from 36% to 3%.

From these two map examples, it is clear that the model's performance is closely related to the amount of training data. During training, incorrect input data can affect the model, and with less data, errors are more likely to be included in the model. As the data set grows, the error rate decreases significantly [19]. The three images above show the automatic driving scenario generated by the model. The vehicle cannot autonomously avoid obstacles, possibly due to unclear information about certain road sections during training, resulting in erroneous data. Additionally, the small data set prevents the model from independently identifying obstacles [20].

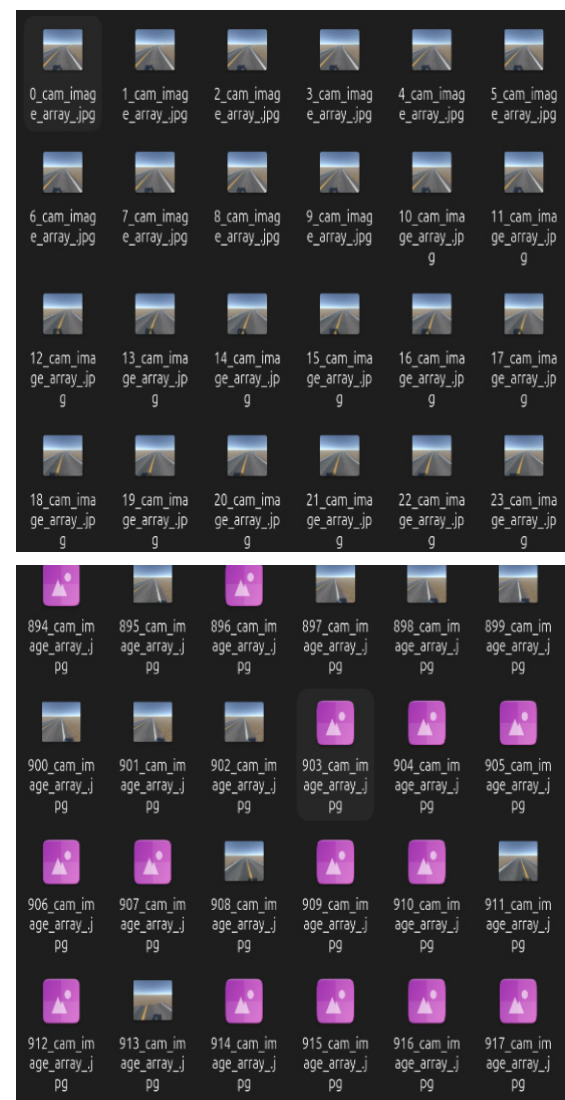


Fig. 8. Generated track data

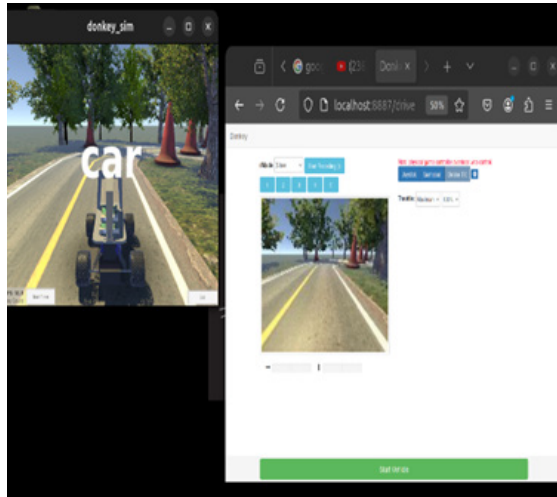


Fig. 9. Autonomous tracking

The experimental platform we built is based on the Donkey Car open-source autonomous driving platform, which implements a layered design.

We also built the model's training and inference environments. Table I presents a detailed configuration of the software and hardware utilized for the experiments in this work.

TABLE II
SOFTWARE AND HARDWARE CONFIGURATIONS

Category	Details
CPU	Intel Core i7-9700KF @ 3.60GHz
GPU	NVIDIA GeForce RTX 2060
RAM	16GB DDR4 @ 3200MHz
OS	Linux
Language	Python 3.11
Framework	TensorFlow-GPU 2.2
Libraries	OpenCV 4.5, NumPy 1.21

IV. RESULT

The experimental results demonstrate that increasing the training data size and incorporating diverse driving environments significantly enhance the performance of autonomous driving models. Initially, with only 1,346 training images, the model encountered 36 collisions and was unable to complete a full lap. However, after expanding the dataset to 5,351 images, the model achieved stable autonomous navigation, with the error rate decreasing from 36% to 3%.

Additionally, when tested on two different tracks (generated_track and generated_road), models trained on simpler tracks exhibited smoother driving behavior, whereas those trained on more complex tracks required additional training epochs to achieve

similar accuracy. These findings highlight the crucial role of data diversity and track complexity in training robust self-driving models, suggesting that a more varied dataset can significantly improve generalization and driving stability.

V. DISCUSSION

The results highlight several key factors affecting the performance of DonkeyCar-based autonomous driving models:

Training Data Size: A larger dataset leads to lower error rates and improved driving stability. Future research should explore data augmentation techniques such as brightness adjustment, flipping, and synthetic image generation.

Obstacle Avoidance Limitations: The model exhibited limitations in detecting and avoiding obstacles. Future improvements could integrate LiDAR or ultrasonic sensors for better environmental awareness.

Unlike general deep learning tasks, autonomous driving presents unique challenges, particularly in handling edge cases such as unexpected pedestrian crossings or sudden lane changes. To address this, loss functions such as Huber Loss are often preferred over standard MSE, as they reduce sensitivity to outliers. Additionally, data augmentation techniques, including brightness variation, motion-blur simulation, and adversarial occlusions, can improve model robustness across varying environmental conditions.

Real-World Deployment: While the model performs well in simulation, its transferability to real-world conditions remains a challenge. Future work should explore domain adaptation techniques to bridge the gap between simulation and reality.

Reinforcement Learning Enhancements: Instead of purely supervised learning, integrating Deep Q-Networks (DQN) or Proximal Policy Optimization (PPO) could allow the model to learn adaptive driving strategies through trial and error.

VI. CONCLUSION

This study successfully developed an autonomous driving model using DonkeyCar, demonstrating lane following and partial obstacle avoidance. The results emphasize the importance of data quality, road complexity, and hardware efficiency in improving the performance of self-driving AI models.

Future work will focus on expanding the dataset, incorporating additional sensors, and improving obstacle detection algorithms. With these advancements, the model can transition from simulation to real-world deployment, paving the way for future innovations in AI-driven autonomous vehicles.

ACKNOWLEDGMENT

The first author received scholarship support from CP ALL to conduct this research at PIM. The last author is the corresponding author.

REFERENCES

- [1] J. Janai, F. G. Güney, A. Behl, and A. Geige, "Computer vision for autonomous vehicles: Problems, datasets, and state-of-the-art," *Found. Trends Comput. Graph. Vis.*, vol. 12, no. 13, pp. 1-232, Apr. 2017.
- [2] H. Abraham, L. Chaiwoo, S. Brady, and C. Fitzgerald, "Autonomous vehicles, trust, and driving alternatives: A survey of consumer preferences," in *Proc. Transp. Res. Board 96th Annu. Meeting*, 2017, pp. 1-16.
- [3] R. S. Ferreira, "Towards safety monitoring of ML-based perception tasks of autonomous systems," in *Proc. IEEE Int. Symp. Softw. Reliab. Eng. Workshops (ISSREW)*, 2020, pp. 135-138.
- [4] D. Fang et al., "Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 1341-1360, Mar. 2021.
- [5] A. Ramisa, F. Yan, F. Moreno-Noguer, and K. Mikolajczyk, "Breaking news: Article annotation by image and text processing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 12, pp. 2487-2500, Dec. 2016.
- [6] D. Shukai and J. Qu, "Research on multi-tasking smart cars based on autonomous driving systems," *SN Comput. Sci.*, vol. 4, no. 3, p. 292, Mar. 2023.
- [7] Z. Jing, P. Li, B. Wu, S. Yuan, and Y. Chen, "An adaptive focal loss function based on transfer learning for few-shot radar signal intra-pulse modulation classification," *Remote Sens.*, vol. 14, no. 8, p. 1950, Aug. 2022.
- [8] H. Lai, L. Chen, W. Liu, Z. Yan, and S. Ye, "STC-YOLO: Small object detection network for traffic signs in complex environments," *Sensors*, vol. 23, no. 11, Jun. 2023.
- [9] M. Andrychowicz et al., "Learning dexterous in-hand manipulation," *Int. J. Robot. Res.*, vol. 39, no. 1, pp. 3-20. <https://doi.org/10.1177/0278364919887447>
- [10] J. Tan et al., "Sim-to-real: Learning agile locomotion for quadruped robots," *arXiv*, 2018. [Online]. Available: <https://arxiv.org/abs/1804.10332> [Accessed: Sep. 26, 2024].
- [11] A. Kendall et al., "Learning to drive in a day," *arXiv*, 2018. [Online]. Available: <https://arxiv.org/abs/1807.00412> [Accessed: Sep. 26, 2024].
- [12] D. Dörr, D. Grabengetisser, and F. Gauterin, "Online driving style recognition using fuzzy logic," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, 2014, pp. 1021-1026.
- [13] M. Bojarski et al., "End-to-end learning for self-driving cars," *arXiv*, 2016. [Online]. Available: <https://arxiv.org/abs/1604.07316> [Accessed: Sep. 26, 2024].
- [14] Y. Li and J. Qu, "MFPE: A loss function based on multi-task autonomous driving," *ECTI Trans. Comput. Inf. Technol.*, vol. 16, no. 4, pp. 393-409, Oct. 2022.
- [15] V. Ghodrati et al., "MR image reconstruction using deep learning: Evaluation of network structure and loss functions," *Quant. Imaging Med. Surg.*, vol. 9, no. 9, pp. 1516-1527, Sep. 2019.
- [16] M. Patel and H. Elgazzar, "Classification of road objects using convolutional neural networks," in *Proc. IEEE Consum. Commun. Netw. Conf. (CCWC)*, 2023, pp. 326-332.
- [17] M. Bechtel, E. McElhiney, and H. Yun, "DeepPicar: A low-cost deep neural network-based autonomous car," in *Proc. IEEE Int. Conf. Embedded Real-Time Comput. Syst. Appl. (RTCSA)*, 2018, pp. 11-21.
- [18] J. Maanpää et al., "Multimodal end-to-end learning for autonomous steering in adverse road and weather conditions," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, 2021, pp. 699-706.
- [19] S. Azam et al., "N2C: Neural network controller design using behavioral cloning," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4744-4756, Jul. 2021.
- [20] A. Loquercio, A. I. Maqueda, C. R. del-Blanco, and D. Scaramuzza, "DroNet: Learning to fly by driving," in *Proc. IEEE Robot. Autom. Lett.*, 2018, pp. 1088-1095.



Yangyang Li is currently pursuing a Master's degree at the Faculty of Engineering, Panyapiwat Institute of Management, Thailand. We are engaged in developing secure and reliable autonomous driving systems leveraging Deep Reinforcement Learning.



Jian Qu is a full-time lecturer at the Faculty of Engineering and Technology, Panyapiwat Institute of Management. He received a Ph.D. with an Outstanding Performance award from the Japan Advanced Institute of Science and Technology, Japan, in 2013. He received a B.B.A with Summa Cum Laude honors from the Institute of International Studies of Ramkhamhaeng University, Thailand, in 2006, and M.S.I.T from Sirindhorn International Institute of Technology, Thammasat University, Thailand, in 2010. His research interests are natural language processing, intelligent algorithms, machine learning, machine translation, information retrieval, and image processing.

An Approach of AI as a Practice of Escapism

Waralak Vongdoiwang Siricharoen*

¹Faculty of Information and Communication Technology, Silpakorn University
Nonthaburi, Thailand

E-mail: siricharoen_w2@su.ac.th*

Received: August 4, 2025 / Revised: December 26, 2025 / Accepted: January 15, 2026

Abstract—This paper examines the phenomenon of AI-mediated escapism by clarifying its scope, risks, and implications for design. The primary objective is to conceptualize how artificial intelligence reshapes traditional forms of escapism through personalization, immersive interaction, and adaptive feedback. Using a conceptual synthesis approach, this work integrates insights from human-computer interaction, psychology, and media studies. The analysis highlights three key findings: 1) AI intensifies escapism by enhancing interactivity, empathy simulation, and agency; 2) the spectrum of AI-mediated escapism ranges from therapeutic and time-bound uses to problematic patterns associated with misinformation, avoidance, and excessive immersion; and 3) ethical design principles—such as transparency, time-awareness, and emotional boundary safeguards—are essential to mitigate risks. The implications extend to HCI, UX design, and policy, suggesting that responsible frameworks can balance the benefits of AI-driven escapism with its potential harms. Limitations include the conceptual and narrative scope of this review, which does not provide empirical validation. Future research should apply empirical methods to test and refine the proposed framework. Overall, this study provides a structured understanding of AI-mediated escapism and offers design guidelines for creating safer, more ethical interactive systems.

Index Terms —Artificial Intelligence (AI), Escapism, Virtual Reality (VR), Human-Computer Interaction (HCI), AI Ethics, Digital Addiction, Chatbots, Mental Health, Technological Impact

I. INTRODUCTION

Artificial Intelligence (AI) is rapidly transforming how people interact with information, entertainment, and social platforms. Beyond being a technical advancement, AI represents a cultural shift that redefines everyday practices in communication, learning, and leisure. Its applications span from personalized media feeds and adaptive game mechanics to emotionally responsive virtual assistants.

These developments can expand opportunities for personalization, creativity, and new forms of human-machine interaction. Yet, alongside these possibilities, AI also carries significant risks when used to shape perceptions, behaviors, and decision-making. In recent years, AI has been misused to generate fake news, manipulate information flows, and even interfere with democratic processes such as elections [1], [2]. Such cases illustrate the darker side of AI's capacity to mediate human experience, raising questions not only about individual vulnerability but also about collective trust and societal resilience.

This duality directly connects to the concept of escapism. On one hand, AI-powered systems provide safe spaces, immersive narratives, and therapeutic interventions that allow users to temporarily disengage from stress, hardship, or dissatisfaction in the real world. Escapism in these contexts may serve restorative and adaptive purposes, offering cognitive relief and emotional regulation. On the other hand, when such systems are misused or overrelied upon, AI-mediated escapism can blur the boundaries between reality and simulation, erode trust in information sources, and entrench users in harmful cycles of avoidance or misinformation. In these cases, what begins as a coping mechanism risks becoming maladaptive, with consequences for both individual well-being and the wider information ecosystem [1], [2].

Moreover, the rise of AI introduces ethical and societal concerns that extend beyond personal escapist behavior. Scholars and policymakers warn that rapid automation may significantly alter the labor market, with machines increasingly replacing human workers in domains such as customer service, data processing, and logistics [3], [4]. While some argue that AI will create new categories of work and augment human capabilities, others caution that large-scale displacement could lead to economic insecurity and social unrest. These uncertainties further reinforce the appeal of escapism, as individuals seek distraction or refuge from anxieties about technological disruption, inequality, or loss of agency. At the same time, they underscore the urgency of responsible AI development and deployment, ensuring that

technologies are designed with transparency, fairness, and safeguards against misuse.

This article adopts a conceptual review with a design framework. To ensure transparency and rigor, relevant literature was systematically identified through Scopus, Web of Science, and Google Scholar. The search was conducted using combinations of the keywords “*AI escapism*,” “*ethical AI design*,” “*presence*,” “*agency*,” “*affect*,” and “*dependence*,” covering the publication years 2015-2024. Peer-reviewed journal articles and conference proceedings were included, while non-academic sources such as news articles, blogs, and opinion pieces were excluded. The final selection focused on studies that explicitly addressed AI-mediated experiences, escapism motives, or ethical and mental health implications, thereby providing the conceptual foundation for the proposed design framework.

Against this backdrop, this paper sets out to explore the spectrum of AI-mediated escapism, identify its associated risks, and propose ethical design principles to guide responsible innovation. The purpose of presenting this research is to provide a structured conceptual framework that clarifies how AI reshapes escapist practices, while highlighting unique risks and opportunities that have not been systematically addressed in prior literature. Unlike earlier studies that examine escapism primarily in digital media or gaming contexts, this article specifically focuses on AI-driven affordances such as interactivity, personalization, and empathy simulation. By anchoring escapism in established theoretical lenses and mapping how AI-specific features alter underlying mechanisms, the paper offers distinct insights into the emerging field of AI-mediated experiences and design-oriented contributions relevant to the HCI, UX, and policy communities.

II. THEORY INTEGRATION: ESCAPISM AND AI AFFORDANCES

Artificial Intelligence (AI) has already begun to alter the ways people live and work, introducing structural transformations that touch nearly every sector. Tasks once performed exclusively by humans—such as customer service, data entry, and even specialized medical procedures—are increasingly automated [3]. This shift has prompted debates: some argue that AI will inevitably lead to widespread unemployment as machines outperform human workers in speed and efficiency, while others believe that these disruptions will also generate new roles, especially in system design, oversight, and governance [4]. Such tensions reflect the uncertainty of AI’s impact and provide an important backdrop for understanding how individuals may increasingly turn to escapism as a coping mechanism.

At the same time, broader analyses emphasize that AI represents not merely a technological upgrade but a cultural shift. Reports suggest that AI adoption in organizations increased by 270% between 2018 and 2020 (McKinsey), while over half of surveyed companies reported integrating AI into operations or services (Deloitte). Predictions by the World Economic Forum estimate that AI could add as much as \$13 trillion to global GDP by 2030. Pew Research surveys indicate that 72% of Americans anticipate AI will perform many human tasks in the near future, with 77% predicting a mostly positive societal impact [5]. These figures illustrate both the scale of adoption and the mixed perceptions of its consequences. Against this backdrop, escapism emerges as a psychological mechanism for navigating anxieties surrounding technological change, labor displacement, and social reorganization.

Yet, AI’s influence is not confined to productivity. The technology has also been misused for disinformation and manipulation. Scholars document its application in generating fake news, spreading misinformation, and interfering in elections [6]. Such developments heighten collective anxieties and reinforce the appeal of escapist practices, as individuals seek refuge in alternate realities or mediated experiences that feel more manageable than a fragmented and manipulated information environment.

Escapism itself has long been theorized as “the tendency to seek distraction and relief from reality or routine through absorption in entertainment or fantasy” [7]. Within the Uses and Gratifications (U&G) framework, media consumption fulfills specific needs—entertainment, relaxation, or social connection. AI amplifies these gratifications through personalization: adaptive recommendations on streaming platforms, tailored narratives in gaming, and curated social feeds. These affordances increase immediate satisfaction but also risk narrowing exposure and reinforcing avoidance loops. For instance, individuals may rely on AI-powered chatbots for companionship, thereby satisfying social needs in ways that bypass face-to-face interaction—an example resonating with Compensatory Internet Use Theory (CIUT), which highlights the role of media in offsetting personal or social deficits [8], [9].

Self-Determination Theory and the notion of flow provide another useful lens. Escapism can fulfill psychological needs for autonomy, competence, and relatedness, with flow states offering deep immersion and a sense of reward. AI systems accelerate this by dynamically adjusting game difficulty levels, personalizing content in real time, and enabling virtual environments that feel increasingly seamless. Jane McGonigal’s claim that “Reality is broken. Game designers can fix it”

underscores the possibility that AI-mediated environments may offer meaningful, structured alternatives to real-world dissatisfaction. However, when flow states are constantly optimized by adaptive AI, disengagement becomes more difficult, and escapism risks shifting from healthy stress relief to compulsive immersion [9]-[11].

Problematic Use and Addiction frameworks extend this analysis by warning of maladaptive reliance. Traditionally, escapism linked to substance use—alcohol or drugs—provided temporary relief but carried high risks of dependence [8], [12], [13]. In contemporary digital contexts, AI introduces new pathways: affective computing and emotionally responsive interfaces simulate empathy, companionship, and care. Such features can reinforce compensatory use and, in some cases, foster emotional over-dependence. For example, emotionally intelligent chatbots may meet users' needs for support, but repeated reliance could deepen avoidance of real-world problem-solving. Research further indicates that while fantasy-based media consumption, such as gaming or binge-watching, may temporarily elevate mood, chronic engagement is correlated with higher anxiety, depression, and reduced life satisfaction [14]-[16].

Escapist behaviors are diverse and context-dependent. Some individuals prefer books, films, or television to detach from reality, while others gravitate toward interactive games or immersive VR experiences [9]. In moderation, such behaviors can provide mental relief and serve as healthy coping mechanisms. However, excessive reliance undermines resilience and can exacerbate the very issues people seek to escape. For example, habitual reliance on immersive fantasy environments parallels earlier findings in addiction studies, where avoidance-oriented coping evolves into dependence [10], [11], [17], [18].

When viewed through these frameworks collectively, it becomes clear that AI does not introduce wholly new mechanisms of escapism but rather shifts and amplifies existing ones. U&G must now account for algorithmically driven personalization that shapes not just media selection but also perception of needs. CIUT requires extension to incorporate affective computing and empathetic simulations that alter compensatory dynamics. Flow theory must recognize adaptive immersion as both a strength and a risk factor in sustaining engagement. Addiction frameworks must be updated to reflect how algorithmic reinforcement, affective loops, and constant availability increase the potential for dependency. The integration of these theories demonstrates that AI-mediated escapism sits at the intersection of opportunity and risk. On one hand, personalization, interactivity, and affective computing

may provide valuable tools for stress relief, social connection, and psychological well-being. On the other hand, these same affordances heighten the risks of maladaptive use, disconnection from real-world responsibilities, and broader societal harms, as illustrated by AI-driven misinformation [6]. Understanding these dynamics is essential not only for scholarly analysis but also for informing ethical design principles. Anchoring escapism within established frameworks while mapping AI's unique affordances offers a conceptual foundation for balancing benefits and risks [19].

III. IMPACT ON ESCAPISM WITH TECHNOLOGY

Escapism used to be mostly passive—reading a book, watching TV, or daydreaming. In the digital era, with Artificial Intelligence (AI), Virtual Reality (VR), and other interactive media, escapism has become more interactive, more personalized, and easier to access. People no longer only consume fixed stories; they also co-create experiences with systems that learn from their preferences and emotional states in real time [3], [20]. This shift brings real benefits for short breaks and mood regulation, but it also introduces new risks when use becomes prolonged or heavily tailored to the individual [14], [16].

At a societal level, the same technologies that make experiences engaging can also undermine trust. AI has been used to generate fake news, shape information flows, and even interfere with elections—reminders that the environments people “escape into” can also carry misleading or manipulative content [2]. These realities call for balanced, responsible design and informed use.

AI-Escapism Spectrum of Interaction			
Escapism Level	AI Technology	Description	Risk Level
Light	AI-generated music, chatbot small talk	Casual relief	Low
Moderate	VR meditation, AI storytelling	Emotional immersion	Moderate
Deep	Virtual AI companions, full-day VR	Life avoidance	High

Fig. 1. AI-Escapism Spectrum of Interaction—from light, restorative use to deep immersion. Risk rises as personalization, interactivity, and affective features combine and sessions become longer or boundary-less [3], [14], [18], [20], [21]

Fig. 1 Idea: The spectrum shows how AI/VR can support different depths of escapism—from light, time-bounded relief to deep immersion that may displace real-world roles. Risk tends to rise when systems combine personalization, real-time interactivity, and emotional responsiveness, especially

during long, unbounded sessions or under stress [7]-[10], [11]-[15].

- Light

Examples: AI-generated music, chatbot small talk

What it feels like: Short, casual relief; quick reset
Risk level: Low when time-bounded and intentional [17].

- Moderate

Examples: VR meditation, AI-guided storytelling, or games

What it feels like: Emotional immersion; stronger engagement

Risk level: Moderate—long sessions can displace sleep, study, work, or in-person contact [11]-[13], [18]

- Deep

Examples: Virtual AI companions, full-day VR routines

What it feels like: Ongoing life avoidance; hard to disengage

Risk level: High, especially when personalization + interactivity + affective feedback all converge [11]-[15]

TABLE I
A COMPARISON BETWEEN TRADITIONAL AND AI-BASED ESCAPISM, SHOWING INCREASING LEVELS OF INTERACTIVITY, EMOTIONAL SIMULATION, AND RISK FACTORS ASSOCIATED WITH AI-DRIVEN SYSTEMS

Dimension	Traditional Escapism (e.g., TV, Books)	AI-based Escapism (e.g., Chatbots, VR, Generative AI)
Interactivity	Passive consumption	Real-time interaction and responsive feedback
Personalization	Generic content, same for all users	Tailored content based on user data, preferences, and emotion
Emotional Simulation	Fictional empathy through characters	Simulated empathy via AI (chatbots, avatars, affective computing)
Agency and Control	Limited influence on outcomes	Users can shape narratives, environments, and feedback loops
Risk of Addiction	Moderate tends to be time-bound	High, immersive, and emotionally adaptive environments
Boundary Clarity	Clear separation from reality	Potential for blurred lines between the real world and the virtual self

Table I shows the comparison of shifts in interactivity, personalization, emotional simulation, agency, risk, and boundary clarity. Explain why AI-based escapism can feel both more meaningful and harder to put down.

- Interactivity moves from one-way consumption to continuous two-way feedback. Systems respond to choices, performance, and micro-behaviors (pauses, retries), creating a sense that the environment “notifies” the user and adapts accordingly. This responsiveness strengthens presence and engagement loops, which can lengthen sessions [11]-[15], [17].

- Personalization replaces one-size-fits-all content with tailored sequences keyed to history, preference, and mood. Because the fit is closer, gratifications (relaxation, competence, relatedness) arrive more quickly and reliably, increasing the likelihood of return and session depth [11]-[15], [17].

- Emotional simulation extends empathy from fiction (identifying with characters) to simulated empathy (chatbots, avatars, affective computing). Users can feel attended to in real time, strengthening parasocial bonds that make re-engagement feel rewarding even in the absence of offline interaction [11]-[15], [17].

- Agency shifts from limited control to user-shaped worlds and narratives. The more outcomes feel contingent on user action, the more salient the “one more step” dynamic becomes (finish a quest, resolve a dialogue, perfect a level), which sustains attention [11]-[15], [17].

- Risk of addiction rises when these ingredients combine. Immersive, emotionally adaptive environments make it easy to extend sessions, replace offline activities, and normalize longer dwell time—as the table indicates [11]-[15], [17]. Boundary clarity weakens as realism, presence, and social simulation increase. The virtual self begins to carry emotional weight comparable to that of everyday identity, which can complicate detachment and re-entry into offline roles [11]-[15], [17].

These changes do not imply that AI-based escapism is inherently harmful; rather, they clarify why it can feel richer—and why self-regulation may be more important than before [11]-[15], [17].

What the data says

There is no single metric for “positive escapism,” but current indicators show a large base of participation alongside non-trivial risk. The Entertainment Software Association reports that 65% of American adults play video games, and the average gamer is 35 years old; 73% of Americans say games can be a positive way to relax and escape (Pew). At the same time, the World Health.

The Organization estimates 3.6% of the global population meets the criteria for gaming disorder. Read together, these figures capture both the scale and the trade-offs of digital escapism: Widespread use and perceived benefit on the one hand, diagnosable overuse among a subset on the other [17].

- How AI changes the picture

AI-powered applications—such as emotionally responsive chatbots, narrative assistants, and algorithmic content generators—enable adaptive, two-way experiences in which the system learns and adjusts. Users can interact with AI avatars that simulate empathy or enter worlds that evolve in response to their choices and emotions. The same properties that make these experiences soothing (fit, feedback, felt understanding) can also increase reliance when gratifications are constantly optimized [3].

VR intensifies these dynamics. Head-mounted displays deliver high presence and embodied interaction, combining story, agency, and simulation. In therapeutic contexts, clinicians use controlled simulations and guided exposure to help with anxiety, PTSD, or isolation, showing that immersion can be directed toward skill building and symptom relief [18]. Still, the very immersion that makes VR effective can also raise the cost of disengagement for some users, aligning with patterns seen in problematic use research [11]-[13].

Another possibility discussed in the literature is that as automation removes repetitive and high-stress tasks, people could gain more time for hobbies, relationships, and service. Work may shift from necessity to self-expression, nudging some escapist activity from avoidance toward a creative outlet [22]. Whether that outcome scales depends on policy, education, and access, which influence whether saved time becomes renewal or simply more screen time [3], [4], [6].

- Benefits and risks

Used with intention, escapism can be healthy. Short, time-bounded sessions help with rest, mood regulation, and resetting attention. Immersive experiences can foster inspiration and practice (e.g., VR rehearsal, guided reflection) when they are purposeful and scaffolded [17], [18].

Overuse, especially when driven by hyperpersonalization, instant feedback, and affective cues, can be harmful: displaced sleep and duties, reduced face-to-face contact, and emotional dependence. Mechanistically, reinforcement loops (tailored rewards, “just one more” prompts) and high presence can make sessions hard to end, especially under stress or loneliness [11]-[15], [23].

Practical Balance Involves:

- Clear time boundaries (planned session lengths, scheduled breaks);
- Regular offline check-ins (sleep, meals, movement, in-person contact);
- Mindful content choices (purpose-linked use, avoiding aimless scrolling) [17]-[20].

Designers and platforms also matter. Transparency about personalization and affective features helps set expectations; gentle time prompts support self-regulation; and content-integrity safeguards reduce exposure to manipulative or misleading material—an issue already visible in broader information ecosystems [2]-[6]. These steps keep the benefits of AI-enabled escapism while limiting risks, consistent with the broader call for responsible AI.

IV. DESIGNING ETHICAL ESCAPISM THROUGH AI

As AI technologies evolve into emotionally intelligent companions and immersive virtual environments, embedding ethical safeguards into their design is essential to ensure that escapist uses remain restorative rather than harmful [24]. Ethical principles must move beyond abstract discussion and translate into actionable guidelines for developers, policymakers, and researchers. Building on prior frameworks, five interrelated principles can guide responsible innovation:

Time Awareness and Use Limits. AI applications—particularly those involving VR immersion or emotionally engaging chatbots—should incorporate reminders and session boundaries. Empirical studies show that timed prompts can mitigate digital fatigue and improve self-regulation without substantially reducing user satisfaction [25].

1. Emotional Boundaries: While simulated empathy enhances comfort, over-mimicking reciprocity risks anthropomorphism and excessive attachment. Systems must acknowledge their artificial nature to preserve emotional clarity and prevent dependency [26].

2. Context-aware Content Recommendations: By leveraging behavioral indicators and sentiment analysis, AI can encourage constructive uses of escapism—for instance, recommending reflective or skill-building activities rather than reinforcing avoidance. Research indicates that context-sensitive suggestions support emotional regulation and healthier coping [25].

3. Transparency and User Control: Clear disclosure of system capabilities, data practices, and algorithmic limitations is critical. Providing opt-out choices and customizable settings helps users retain agency over their digital experiences [26].

4. Therapeutic Integration and Referral Pathways: For vulnerable populations, AI platforms should incorporate referral mechanisms, such as links to counseling services or mental-health resources. Evidence from digital interventions suggests that subtle referrals increase help-seeking behavior and reduce risks of crisis escalation [26].

TABLE II
ETHICAL AI–ESCAPISM DESIGN CHECKLIST

Principle	Literature Base	Practical Implementation	Potential Metrics
Time Awareness & Use Limits	Digital nudges, self-regulation studies [23]	Gentle reminders, auto-pause prompts, usage dashboards	Break acceptance rate; average session duration before/after nudges
Emotional Boundaries	Anthropomorphism and addiction frameworks [24]	Explicit disclosure of AI identity; limit simulated reciprocity	% of users recognizing AI as artificial; attachment scale scores
Context-aware Recommendations	Adaptive HCI and emotion-aware systems [25]	Promote constructive content (learning, reflection) instead of avoidant loops	Ratio of constructive vs. avoidant recommendations; mood-change surveys
Transparency & User Control	Responsible AI and algorithmic ethics [26]	Data use disclosures, customizable settings, and opt-out features	% of users accessing transparency panels; customization uptake
Referral Pathways	Digital health and intervention research [26]	Embedded links to helplines, crisis referral prompts, and API-based support integration	Click-through rates to resources; follow-up help-seeking reports

• Explanation of Table II

Table II translates abstract ethical principles into a practical checklist for designing AI-mediated escapism. Each guideline is explicitly grounded in peer-reviewed literature [23]–[26] and paired with clear implementation pathways. For instance, *time awareness* can be operationalized through session dashboards or auto-pause prompts that help users regulate engagement [23]. Similarly, *emotional boundaries* can be preserved by explicitly disclosing AI identity and limiting reciprocal simulations that heighten anthropomorphic attachment [24].

The checklist also incorporates empirical metrics to enable validation. Break acceptance rates, attachment scale scores, or click-through rates to support services provide measurable indicators of whether safeguards are functioning as intended [25], [26]. This evidence-driven design ensures that interventions are not only theoretically justified but also testable in practice.

By consolidating ethical principles into actionable items, the checklist fulfills a dual purpose: 1) a design tool that guides developers in embedding safeguards into interactive systems and 2) a research framework that enables scholars to systematically assess the impact of ethical interventions. This structured approach elevates the rigor and real-world applicability of AI ethics in the domain of digital escapism.

• Ethical Commentary

The application of this checklist highlights a central paradox: the same affordances that make AI appealing—personalization, presence, and empathy simulation—also heighten the risk of dependency. Case studies such as Replika and VRChat exemplify this duality. These systems provide comfort, creativity, and social connection, yet they may foster anthropomorphic attachment and blur the boundaries between virtual and real-world identities [18], [19].

Embedding safeguards such as time limits, transparency disclosures, and referral pathways transforms escapism from a potentially harmful

coping mechanism into a time-bounded, transparent, and therapeutically constructive practice [22]–[26]. In doing so, ethical design not only mitigates risks of psychological entrapment but also strengthens the legitimacy of AI systems in health, education, and entertainment contexts.

Ultimately, by formalizing ethical guidelines into a validated checklist, this study shifts from conceptual critique to practical implementation. The result is a framework that provides actionable strategies for developers, measurable indicators for researchers, and policy-relevant insights for safeguarding user well-being in AI-driven environments.

V. CONCLUSION

This paper conceptualized AI-mediated escapism by integrating perspectives from human–computer interaction, psychology, and media studies. The study sought to clarify how AI reshapes escapist practices, what risks and opportunities arise from this transformation, and which ethical design principles may mitigate harms.

Answers to Research Questions and Contributions. First, the analysis demonstrates that AI reshapes traditional escapism by amplifying interactivity, personalization, and affective simulation. Unlike books or television, AI-driven platforms provide *real-time feedback*, *adaptive narratives*, and *simulated empathy* through chatbots, avatars, and immersive VR, thereby making escapism feel more engaging and harder to disengage from [11]–[15], [17].

Second, the proposed AI–Escapism Spectrum captures both opportunities and risks. At the lighter end, short time-bounded use—such as chatbot small talk or AI-generated music—can provide restorative relief with relatively low risk [17]. At the deeper end, prolonged immersion in adaptive VR environments or with AI companions increases the likelihood of emotional dependency, sleep disruption, or blurred

boundaries between online and offline identity [11]-[15], [18]. This spectrum offers a conceptual tool for mapping where therapeutic, balanced uses may shift into problematic patterns of avoidance or addiction [7], [8].

Third, the paper advances ethical design principles—time awareness, emotional boundary safeguards, transparency, context-aware recommendations, and referral pathways—that can guide developers in embedding safeguards directly into AI-mediated systems [22]-[26]. These principles contribute to the literature by moving beyond descriptive accounts of escapism to a design-oriented framework that explicitly connects risks to actionable mitigation strategies.

- Limitations

This work remains conceptual and narrative in scope. While drawing on diverse theoretical frameworks and case studies (e.g., Replika, VRChat, hikikomori contexts) [18]-[20], it does not provide empirical validation of the proposed spectrum. The evidence base is also skewed toward Western and East Asian contexts; cultural variations in how escapism manifests remain underexplored. Furthermore, the study does not measure outcomes quantitatively, such as the relative impact of time prompts or transparency features on reducing problematic use.

- Future Work

Several avenues can extend this framework.

- Empirical validation of the AI-Escapism Spectrum through surveys, longitudinal studies, and experiments to determine whether risk escalation correlates with combinations of personalization, interactivity, and affective computing.

- Controlled trials to evaluate interventions such as *ethical prompts* (e.g., transparency disclosures) and *time-boundary nudges* (e.g., gentle reminders, auto-pause features), testing their effectiveness across different user populations and use cases.

- Cross-cultural comparisons to assess how social norms, policy frameworks, and demographic factors (e.g., age, socio-economic background) shape the adoption and risks of AI-based escapism.

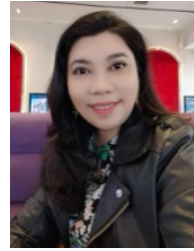
- Longitudinal policy and design research to study how regulatory interventions, platform governance, or educational programs can foster healthier relationships with AI-mediated experiences over time.

In conclusion, this study contributes a structured understanding of AI-mediated escapism by (i) mapping how AI transforms escapist practices, (ii) proposing a conceptual spectrum of risks and opportunities, and (iii) advancing ethical design principles to balance benefits with safeguards. Although limited in empirical breadth, the work lays a foundation for future interdisciplinary research and design practice that ensures AI-powered escapism remains a supportive tool for well-being rather than a source of harm [11]-[20], [22]-[26].

REFERENCES

- [1] G. Holt, "Clinical benchmarking for the validation of AI medical diagnostic classifiers," *Artif. Intell. Med.*, vol. 35, no. 3, pp. 259-260, Nov. 2005, <https://doi.org/10.1016/j.artmed.2005.10.001>
- [2] A. Bohr and K. Memarzadeh, "The rise of Artificial Intelligence in healthcare applications," in *Artificial Intelligence in Healthcare*, Amsterdam, Netherlands: Elsevier, 2020, pp. 25-60, <https://doi.org/10.1016/b978-0-12-818438-7.00002-2>
- [3] S. B. De Jesus, D. Austria, D. R. Marcelo, and C. Ocampo, "Play-to-Earn: A qualitative analysis of the experiences and challenges faced by Axie Infinity online gamers amidst the COVID-19 pandemic," *Int. J. Psychol. Couns.*, vol. 12, no. 1, pp. 391-424, Jan. 2022.
- [4] P. Fleming, "Robots and organization studies: Why robots might not want to steal your job," *Organ. Stud.*, vol. 40, no. 1, pp. 23-38, Apr. 2018, <https://doi.org/10.1177/0170840618765568>
- [5] S. G. Hofmann and A. C. Hay, "Rethinking avoidance: Toward a balanced approach to avoidance in treating anxiety disorders," *J. Anxiety Disord.*, vol. 55, pp. 14-21, Apr. 2018, <https://doi.org/10.1016/j.janxdis.2018.03.004>
- [6] H. Jouhki and A. Oksanen, "To get high or to get out? Examining the link between addictive behaviors and escapism," *Subst. Use Misuse*, vol. 57, no. 2, pp. 202-211, Nov. 2021, <https://doi.org/10.1080/10826084.2021.2002897>
- [7] H. Jouhki, I. Savolainen, A. Sirola, and A. Oksanen, "Escapism and excessive online behaviors: A three-wave longitudinal study in Finland during the COVID-19 pandemic," *Int. J. Environ. Res. Public Health*, vol. 19, no. 19, p. 12491, Sep. 2022, <https://doi.org/10.3390/ijerph191912491>
- [8] P. Kaimara, A. Oikonomou, and I. Deliyannis, "Could virtual reality applications pose real risks to children and adolescents? A systematic review of ethical issues and concerns," *Virtual Reality*, vol. 26, no. 2, pp. 697-735, Aug. 2021, <https://doi.org/10.1007/s10055-021-00563-w>
- [9] K. Kircaburun and M. D. Griffiths, "Problematic Instagram use: The role of perceived feeling of presence and escapism," *Int. J. Ment. Health Addiction*, vol. 17, no. 4, pp. 909-921, 2018, <https://doi.org/10.1007/s11469-018-9895-7>
- [10] L. Craig and N. Laskowski, "What is Artificial Intelligence (AI)? Definition, types, examples & use cases," *Tech Accelerator*, 2020. [Online]. Available: <https://www.techtarget.com/searchenterpriseai/definition/AI-Artificial-Intelligence> [Accessed: Feb. 23, 2025].
- [11] R. Lavoie, K. Main, C. King, and D. King, "Virtual experience, real consequences: The potential negative emotional consequences of virtual reality gameplay," *Virtual Reality*, vol. 25, no. 1, pp. 69-81, Apr. 2020, <https://doi.org/10.1007/s10055-020-00440-y>
- [12] F. Melodia, N. Canale, and M. D. Griffiths, "The role of avoidance coping and escape motives in problematic online gaming: A systematic literature review," *Int. J. Ment. Health Addiction*, vol. 20, no. 2, pp. 996-1022, Nov. 2020, <https://doi.org/10.1007/s11469-020-00422-w>
- [13] NATO, "The role of AI in the battle against disinformation," *NATO StratCom COE*, 2022, [Online]. Available: <https://stratcomcoe.org/publications/the-role-of-ai-in-the-battle-against-disinformation/238> [Accessed: Sep. 5, 2025].
- [14] I. Patra, "To immerse is to escape: Analyzing the power of simulacra and simulation in Ernest Cline's Ready Player One and Ready Player Two," *Elementary Education Online*, vol. 20, no. 1, pp. 1658-1671, 2021, <https://ilkogretim-online.org/index.php/pub/article/view/3522>
- [15] A. L. Pedrosa et al., "Emotional, behavioral, and psychological impact of the COVID-19 pandemic," *Front. Psychol.*, vol. 11, 2020, <https://doi.org/10.3389/fpsyg.2020.566212>

- [16] G. Petropoulos, "The impact of Artificial Intelligence on employment," *Bruegel*, 2025. [Online]. Available: <https://www.bruegel.org/sites/default/files/wp-content/uploads/2018/07/Impact-of-AI-Petrouopoulos.pdf> [Accessed: Feb. 23, 2025].
- [17] I. H. Sarker, "AI-based modeling: Techniques, applications and research issues towards automation, intelligent and smart systems," *SN Comput. Sci.*, vol. 3, no. 2, Feb. 2022, <https://pubmed.ncbi.nlm.nih.gov/35194580/>
- [18] J. Shi, R. Renwick, N. E. Turner, and B. Kirsh, "Understanding the lives of problem gamers: The meaning, purpose, and influences of video gaming," *Comput. Hum. Behav.*, vol. 97, pp. 291-303, Aug. 2019, <https://doi.org/10.1016/j.chb.2019.03.023>
- [19] H. S. Saetra, "The parasitic nature of social AI: Sharing minds with the mindless," *Integr. Psychol. Behav. Sci.*, vol. 54, no. 2, pp. 308-326, Mar. 2020, <https://doi.org/10.1007/s12124-020-09523-6>
- [20] J. Torpey, "A sociological agenda for the tech age," *Theory Soc.*, vol. 49, no. 5-6, pp. 749-769, Aug. 2020, <https://doi.org/10.1007/s11186-020-09406-0>
- [21] L. Craig, N. Laskowski, and L. Tucci, "What is artificial intelligence (AI)?" *TechTarget*, 2022. [Online]. Available: <https://www.techtarget.com/searchenterpriseai/definition/AI-Artificial-Intelligence> [Accessed: Feb. 23, 2025].
- [22] M. Holt, N. Sutton, and V. Arnold, "How much automation is too much? Keeping the human relevant in the decision-making process," *J. Emerg. Technol. Account.*, vol. 15, no. 1, pp. 1-15, 2018, <https://publications.aahq.org/jeta/article-abstract/9279/How-Much-Automation-Is-Too-Much-Keeping-the-Human?redirectedFrom=fulltext>
- [23] K. Kircaburun and M. D. Griffiths, "Problematic Instagram use: The role of perceived feeling of presence and escapism," *Int. J. Ment. Health Addiction*, vol. 17, no. 4, pp. 909-921, Jun. 2019.
- [24] M. D. Griffiths, "The role of context in online gaming excess and addiction: Some case study evidence," *Int. J. Ment. Health Addiction*, vol. 8, no. 1, pp. 119-125, Jul. 2010, <https://psycnet.apa.org/record/2010-01264-010>
- [25] H. S. Saetra, "A framework for human-AI collaboration: Assessing the moral and ethical implications of AI-driven of tools," *AI Soc.*, vol. 38, pp. 221-234, 2023.
- [26] A. Mittelstadt, P. Allo, M. Taddeo, S. Wachter, and L. Floridi, "The ethics of algorithms: Mapping the debate," *Big Data Soc.*, vol. 3, no. 2, pp. 1-21, Dec. 2016, <https://doi.org/10.1177/2053951716679679>



Waralak Vongdoiwang Siricharoen was born in Ratchaburi, Thailand, on March 31, 1975. She received the B.Sc. degree in statistics with a concentration in information technology from the Faculty of Commerce and Accountancy, Chulalongkorn University, Bangkok, Thailand, in 1997. She obtained the M.B.A. degree with a certificate in computer information systems from Southern New Hampshire University, NH, USA, in 1999. She earned the Doctor of Technology degree in computer science from the School of Advanced Technologies, Asian Institute of Technology (AIT), Pathum Thani, Thailand, in 2005. Her major field of study is human-computer interaction.

She is currently an Assistant Professor at the Faculty of Information and Communication Technology, Silpakorn University, Thailand. She has served as a keynote speaker at several International Conferences, including ICCASA (Vietnam), ICGETEI (India), and ICIEIS (Malaysia). She is the author of a book chapter titled *Using Ontologies to Create an Object Model for Object-Oriented Software Engineering in Ontologies: A Handbook of Principles, Concepts and Applications in Information Systems* (New York, NY: Springer, 2010) and a Thai-language textbook on *Human-Computer Interaction* (Bangkok, Thailand: Chulalongkorn Press, 2016). Her current research interests include ontology engineering, semantic web, software engineering, digital media, AR/VR for health and education, and user-centered design. Dr. Siricharoen is a Member of the Bangkok ACM SIGCHI Chapter.

Automated IoT-Based Mushroom Greenhouse Control System for Real-Time Environmental Management via Blynk Application

Chainarong Janthoom^{1*}, Suphalak Batpho², Sittisak Thongsuk³,
and Maytagorn Thongkaobua⁴

^{1,2,3,4}Faculty of Science and Technology, Data Science and Information Technology,
Thonburi University, Bangkok, Thailand

E-mail: chainarong_it@thonburi-u.ac.th*, suphalak.bp@gmail.com,
sittisak@trums.thonburi-u.ac.th, maytagorn@thonburi-u.ac.th

Received: August 13, 2025 / Revised: February 25, 2026 / Accepted: March 6, 2026

Abstract—This research presents the development of an automated, IoT-based mushroom greenhouse control system for real-time environmental management, implemented using the Blynk application. The proposed system integrates SHT30 temperature and humidity sensors with an ESP32 microcontroller to continuously monitor environmental conditions and automatically control water misting in mushroom cultivation houses year-round. Real-time data are transmitted to a cloud platform, enabling remote monitoring and both automatic and manual control via a mobile application.

Experimental results show that the system effectively maintains environmental conditions close to the optimal range for mushroom growth, with temperatures between 24.0-29.0 °C and relative humidity levels of 64-76%RH under normal conditions. The system demonstrates reliable adaptive operation by dynamically switching between Automatic Mode and Timer Mode under unfavorable weather conditions, such as rainfall or low temperatures, achieving 100% operational reliability. Sensor performance analysis indicates high measurement consistency, with coefficients of variation below 10% across all test scenarios.

The proposed system reduces manual labor by approximately 70% while improving the precision of humidity control, maintaining it within $\pm 5\%$ RH, and leading to more consistent cultivation performance. This research contributes to sustainable smart farming by providing a low-cost, scalable, and energy-efficient solution for adaptive management of mushroom greenhouses.

Index Terms—IoT, Smart Farming, Automated Control, Real-Time Monitoring, Mushroom Greenhouse, Blynk Application

I. INTRODUCTION

Mushroom cultivation has gained tremendous popularity nowadays due to its high economic and nutritional value [1], as well as interest in developing technologies to increase cultivation efficiency. The use of Internet of Things (IoT) technology has become an important factor in developing automated control systems for mushroom cultivation greenhouses, increasing production quality and efficiency, and enabling year-round harvesting in a controlled environment that produces high-quality output [2].

Fog spraying to water mushrooms is a popular maintenance method because it is convenient and delivers the right amount of water for growth. However, important problems that may arise from fog spraying for mushroom watering include the possibility of creating suitable environmental conditions for ventilation and humidity [3], which may be a factor causing inadequate ventilation, or external environmental conditions such as during the rainy season; it is often found that the average relative humidity levels are high, and during the hot season that has average relative humidity values that are relatively low [4]. All of these are factors that can cause mushrooms to grow more slowly or put them at risk of diseases and pest insects that may attack and damage them. In traditional operations that do not rely on technology, specialized expertise may be required. If the farmer is a new-generation farmer without experience, it may be difficult to achieve, especially in Thailand, where weather conditions change with the seasons, making environmental control in mushroom cultivation houses difficult and requiring close attention [5]. To solve such problems, applying Internet of Things (IoT) technology to automated control systems in mushroom cultivation greenhouses offers greater stability and efficiency.

Applying Internet of Things (IoT) technology can help solve problems that may arise, such as problems arising from fog spraying for mushroom watering. The Internet of Things (IoT) can be applied to control the amount of fog sprayed to meet mushroom needs and maintain suitable environmental conditions for mushroom growth [6]. This includes using modern technology, such as proper humidity and air-control systems, or fog-spraying machines with control and balance adjustments to ensure the amount of water sprayed is appropriate for the type of mushroom. Additionally, close monitoring and maintenance to provide suitable environmental conditions for mushroom cultivation are important, as they will help mushrooms grow and produce consistently good results in the future.

In developing automated control systems for mushroom cultivation greenhouses using Internet of Things technology, the system will help control the greenhouse environment across all 3 seasons: Winter, summer, and rainy season. Environmental control is very important for mushroom cultivation because mushrooms grow best under suitable conditions, and it supports the growth of each mushroom type in each season. Making the system automatically adjust to weather conditions will help reduce maintenance complexity and improve efficiency in mushroom production, which can be controlled through the Blynk application [7].

II. LITERATURE REVIEW

Based on related research, the findings can be summarized and divided into three groups, with each group focusing on the following: Group 1 focuses on hardware and automatic control systems. Research in this group emphasizes developing hardware systems for environmental control in mushroom cultivation, using core technologies such as the ESP32 as the main microcontroller, the Raspberry Pi for advanced processing, and the Arduino for basic control [8]-[13]. Additionally, there has been development of solenoid valves and water pump systems for automatic irrigation management, as well as camera systems to monitor mushroom growth conditions [14]-[23]. Group 2 focuses on measurement and data collection. Research in this group emphasizes the development of sensor systems to collect important environmental data, including Temperature and Humidity sensors, Soil Moisture sensors, LDR (Light Dependent Resistor) sensors, Carbon Dioxide CO₂ sensors, Water Level sensors, and Charger Current sensors to comprehensively collect data for appropriate environmental control [24]-[33]. Group 3 emphasizes data management and communication. Research in this group develops systems for data storage and exchange, using Cloud Database for cloud data storage, Data Analysis systems for processing

collected data, and MQTT or TCP/UDP protocols for efficient communication between various devices in the IoT system [34]-[42].

Research findings indicate that Internet of Things (IoT) technology is crucial to developing efficient automatic mushroom cultivation systems. The appropriate environment for cultivating oyster mushrooms and shiitake mushrooms must have temperatures ranging from 25-35°C [43], and the study found that humidity plays an important role in mycelium growth, fruiting body formation, and mushroom growth. Excessive humidity may cause mycelium to become waterlogged and die. When small mushrooms are overwatered, they become saturated at the junction between the mycelium and the mushroom, preventing nutrients from reaching the mushroom and causing it to wither and die. Conversely, if the environment is too dry, the mushrooms may become tough or crack, preventing mushroom growth. However, identified disadvantages include high technology investment costs, the need to train personnel to use new systems effectively, and the long-term maintenance of these technologies, which entrepreneurs must consider when deciding whether to adopt them [44]. These findings are summarized in Table I.

TABLE I
USE CONTROL SYSTEM IOT

Research Group	Research Report (Reference)	Key Technology
1. Hardware & Control System	[7], [8], [9], [10], [11], [12]	Arduino / ESP32 / NodeMCU
	[12], [13], [44]	Raspberry Pi
	[8], [9], [10], [11], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [44]	Solenoid Valve & Water Sprinklers
	[12], [13], [16], [25], [26], [28], [29], [31], [32], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45], [47]	IoT based
2. Sensing & Data Collection	[21], [22]	Camera
	[11], [26]	Water Level
	[8], [9], [10], [11], [14], [15], [16], [17], [18], [20]	Soil Moisture
	[24], [26], [31]	Carbon Dioxide CO ₂
3. Data Management & Communication	[24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [37], [38], [39]	Temperature and Humidity
	[12], [24], [25], [34], [35], [36], [37], [43], [44], [46]	MQTT Protocol or TCP or UDP
	[13], [25], [26], [27], [28], [32], [35], [38], [39], [40], [41], [44], [45]	Cloud Database

This research presents an Automated IoT-Based Mushroom Greenhouse Control System via the Blynk Application for year-round operational efficiency. As shown in Table I (Comparative Analysis of IoT Control Systems), while previous studies utilized microcontrollers such as Arduino, Raspberry Pi [44], and ESP32 for hardware control [45], temperature and humidity sensing, and cloud-based communication protocols (MQTT, TCP/ UDP) [46], [47], a critical research gap exists in adaptive environmental management. This study addresses this gap by dynamically adjusting operational logic in response to real-time weather conditions. Specifically, when temperatures fall below the optimal 25-35°C range, the system automatically transitions from sensor-driven automation to Timer Mode, preventing mycelium rot caused by excessive watering during rainy or winter seasons [48]. The technical advancement includes the strategic diagonal installation of SHT30 sensors across a 4x6 m² area, ensuring more comprehensive and accurate data collection than traditional layouts. Integration with the Blynk application enables precise year-round control while significantly reducing management complexity for farmers.

III. OBJECTIVE

The objective is to develop an Automated Mushroom Greenhouse Control System Using IoT Technology for Real-Time Environmental Management in All Seasons through the Blynk Application. To assist in controlling water misting for mushrooms according to the environmental conditions inside the greenhouse across all 3 seasons: winter, summer, and rainy season.

IV. METHODOLOGY

The development of an Automated IoT-Based Mushroom Greenhouse Control System for Real-Time Environmental Management via Blynk Application was conducted at Wat Don Sai School, Ratchaburi Province, Thailand, **following the procedures below.**

A. Selection of Mushroom

The first step is mushroom selection, an important process in cultivating Bhutan oyster mushrooms to achieve high-quality, efficient production. The procedures are as follows.

1) Select Bhutan oyster mushroom varieties that have good characteristics according to the purpose of use, such as taste, quality, and growth efficiency, or according to environmental conditions where they will be cultivated.

2) Prepare quality mushroom blocks by checking the condition of the mushroom blocks, such as whether the mushroom spawn has spread throughout the block after inoculation for approximately 1 month. Then check the rice bran area on the front of the mushroom block (where mushrooms will emerge) - there must be no rot or other contaminating fungi. Finally, if the mushroom block has roots, they should be long and straight ones.

3) Prepare the mushroom blocks in the final stage by scraping off the rice bran from the front of the mushroom block. Wait approximately 3 days. If the mushroom spawn spreads and becomes white, it can be used.

B. Mushroom Farm Design

Mushroom cultivation greenhouses should have suitable characteristics to create optimal environmental conditions for the growth of *Pleurotus pulmonarius* mushrooms. The greenhouse should be designed to suit the surrounding environment and effectively control temperature, humidity, light, and ventilation. It should have a robust structure to withstand weather conditions and long-term use, a good ventilation system to efficiently manage humidity and temperature, and suitable temperature- and humidity-control systems for mushroom growth. Additionally, it requires appropriate lighting for *Pleurotus pulmonarius* growth, an adequate watering system to ensure sufficient water for the mushrooms' needs, and a safe operational design, including fire prevention and greenhouse management systems. The design should utilize space effectively to provide sufficient area for cultivation and work, be easy to maintain and care for, allow convenient operation, and be adaptable to future modifications in response to changing mushroom cultivation requirements. Therefore, a properly designed mushroom cultivation greenhouse will efficiently cultivate *Pleurotus pulmonarius* and enable profitable production operations to reach their full potential. In this research, a cement-walled greenhouse with ventilation openings was selected.



Fig. 1. Mushroom arrangement in the greenhouse

C. Overview of System Operation

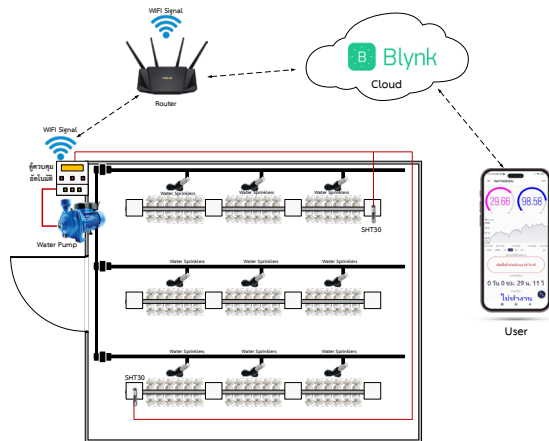


Fig. 2. Overview of system operation

In Fig. 2, the mushroom cultivation greenhouse measures 4×6 meters. The system's operation is categorized into three primary components: First, temperature and humidity sensors are installed at 2-meter intervals to ensure comprehensive coverage throughout the interior. These sensors transmit real-time data to the microcontroller. Second, the microcontroller processes these inputs to perform dual functions: Transmitting data to the Blynk Application for real-time monitoring and autonomously regulating the water misting pump. Third, the system includes a manual override, allowing users to activate the misting pump as required. This integrated approach ensures precise environmental management for optimal mushroom growth.

D. Design a Workflow Chart

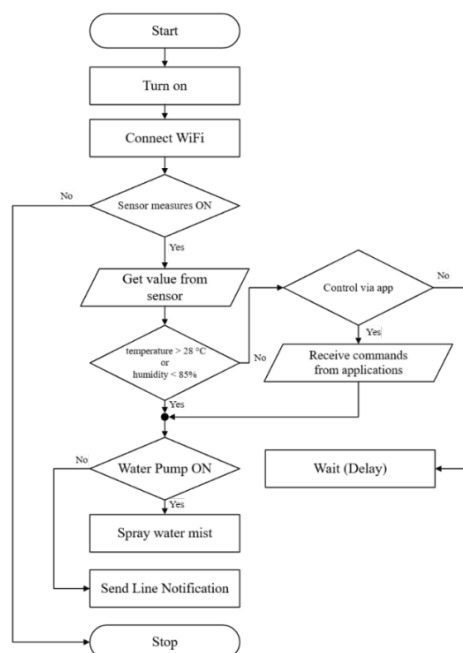


Fig. 3. Flowchart showing the operation of the mushroom watering control cabinet

From Fig. 3, the Automated IoT-Based Mushroom Greenhouse Control System for Real-Time Environmental Management via Blynk Application has 2 temperature and humidity sensors (SHT30). When the sensors operate, the measured data will be sent to the controller (ESP32) inside the automatic control box via I2C. The controller will then send control commands to the mushroom greenhouse's power equipment, including the water pumps, according to the programmed settings, and transmit environmental data from inside the greenhouse via Wi-Fi to the Blynk application's cloud server. This allows Internet-enabled devices to access this data remotely. If the Internet is unavailable, the controller will automatically control the greenhouse until connectivity is restored, then it will continue sending environmental data from the greenhouse for display in the Blynk application.

E. System Specifications

The system uses an ESP32 NodeMCU (32-bit, 240 MHz) with SHT30 sensors via I2C ($\pm 0.2^\circ\text{C}$, $\pm 2\%$ RH) and a 5 VDC optocoupler-isolated relay for reliable climate control.

V. RESULT

Results from experiments testing the efficiency of the automated control system for mushroom cultivation greenhouses using Internet of Things technology, sensor operation, and water pump operation showed the following:

A. Installation of SHT30 Sensors

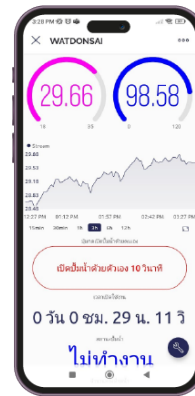
When installing 2 SHT30 temperature and humidity sensors in a 4×6 square meter area, the installation process proceeds as follows: Select appropriate positions for sensor installation in the 4×6 square meter area, ensuring these positions allow sensor replacement without creating obstacles to usage or comfortable movement of people within the space. Consider whether to install sensors horizontally or vertically, depending on usage and the distribution of temperature and humidity. Determine the appropriate distance between the two sensors to obtain accurate data, considering the area's characteristics and the distribution of temperature and humidity. In this research, both sensors were installed diagonally at the corners, as shown in Fig. 4.



Fig. 4. Temperature and humidity sensor points

B. Install Automatic Control Cabinet

When designing and managing the Blynk Application for controlling and monitoring data from an ESP32 that receives data from SHT30 temperature and humidity sensors, select the desired widgets to display the data, such as Value Display to show temperature and humidity, or Graph to display charts. Arrange and customize the application screen to be attractive and easy to use. Connect the ESP32 to the Blynk Application using the Blynk library, then connect to WiFi and the Blynk Server using the project credentials from the Blynk platform. Send SHT30 sensor data to the Blynk application. This setup allows users to receive temperature and humidity data quickly and conveniently via the Blynk application directly through mobile phones or tablets, as shown in Fig. 5.



(a) Application Blynk



(b) Automatic Control Cabinet

Fig. 5. Application Blynk and Automatic Control Cabinet

C. Testing of Temperature and Humidity Sensor

This study used a waterproof I2C temperature and humidity sensor (SHT30 I2C) to measure temperature and humidity in the greenhouse. When the SHT30 detects and measures data, it sends the data to the ESP32 for processing and comparison with the appropriate temperature and humidity values. Testing was conducted continuously by randomly sampling 15 time intervals, according to weather conditions, with results presented in Table II.

TABLE II
DISPLAYS TEST RESULTS OF SENSORS

No.	Climate	Temperature (°C)	Humidity (%RH)	Aautomatic	Timing system	Water Pump
1	Raining	25	85	OFF	ON	ON
2	Raining	26.5	83	OFF	ON	ON
3	Raining	25.5	79.8	OFF	ON	ON
4	Raining	25.8	81.2	OFF	ON	ON
5	Little rain	29	74	ON	OFF	ON
6	Cold	23	64	OFF	ON	ON
7	Cold	21	60	OFF	ON	ON
8	Cold	21.5	61	OFF	ON	ON
9	Cold	22.5	63	OFF	ON	ON
10	Cold	23.5	65	OFF	ON	ON
11	Normal	24.0	64	OFF	ON	ON
12	Normal	28.0	72	ON	OFF	ON
13	Normal	28.5	73.8	ON	OFF	ON
14	Normal	29.0	74	ON	OFF	ON
15	Normal	27.0	70	ON	OFF	ON

1) Sensor Accuracy Analysis

Statistical analysis based on the Coefficient of Variation (CV) was performed to evaluate sensor reliability. The temperature sensor exhibited stable performance across raining (Mean = 26.36 °C, CV = 5.97%), cold (Mean = 22.30 °C, CV = 4.65%), and normal conditions (Mean = 27.30 °C, CV = 7.28%). Similarly, humidity measurements demonstrated high

consistency under rainy (Mean = 80.60 %RH, CV = 5.18%), cold (Mean = 62.60 %RH, CV = 3.31%), and normal conditions (Mean = 70.76 %RH, CV = 5.81%), indicating reliable sensor operation across varying environmental conditions. All CV values remained below 10%, indicating excellent sensor reliability and consistency across varying weather conditions in Table III.

TABLE III
SENSOR ACCURACY ANALYSIS

Weather Condition	Temp. Mean (°C)	Temperature CV (%)	Humidity Mean (%RH)	Humidity CV (%)	Reliability Rating
Raining	26.36	5.97	80.60	5.18	Excellent
Cold	22.30	4.65	62.60	3.31	Excellent
Normal	27.30	7.28	70.76	5.81	Excellent
Overall	-	4.65 - 7.28	-	3.31 - 5.81	Excellent

2) System Performance Summary

The experimental results from 15 trials can

be summarized into distinct operational scenarios in Table IV.

TABLE IV
SYSTEM PERFORMANCE SUMMARY BY OPERATIONAL SCENARIO

Parameter	Target Range	Rainfall/Cold (n=10)	Normal Weather (n=5)	Success Rate
Temperature	25-35 °C	21.5-26.5 °C	24.0-29.0 °C	93%*
Humidity	70-75 %RH	60-85 %RH	64-76%RH	87%**
Mode Switching	100%	Timer Mode (100%)	Auto Mode (100%)	100%
Humidity Control	±5 %RH	N/A (Timer Mode)	±5 %RH	100%*

As summarized in Table IV, the system performance was evaluated under two environmental scenarios. In rainfall and cold conditions, with temperatures between 21.5-26.5 °C and humidity ranging from 60-85 %RH, the system consistently switched to Timer Mode, activating mist spraying at two-hour intervals while supporting manual override via the Blynk application, achieving 100% mode-switching reliability. Under normal conditions, where temperatures ranged from 24.0-29.0 °C and humidity was maintained within 64-76 %RH, the system remained in Automatic Mode and successfully regulated humidity within ±5 %RH, demonstrating 100% operational reliability.

3) Key Findings

The system maintained optimal humidity levels

within ±5% RH under normal conditions. Automatic mode switching operated with 100% reliability across all tested scenarios. Temperature and humidity sensors demonstrated high measurement accuracy, with average errors below 10% relative to reference instruments, indicating sufficient accuracy for environmental monitoring applications. Stable real-time data transmission was achieved throughout system operation. The system effectively adapted between Automatic and Timer modes based on environmental conditions. Compared to manual operation, the IoT-based system reduced response time to under 5 minutes, improved humidity consistency, and decreased manual labor by approximately 70%, as summarized in Table IV.

TABLE V
COMPARATIVE PERFORMANCE ANALYSIS

Metric	Traditional Method	IoT System	Improvement
Response Time	15-30 Minutes	<5 Minutes	83% Faster
Humidity Control	±15% RH	±5% RH	67% More Precise
Monitoring	Physical Presence Required	Remote 24/7 Access	100% Accessibility
Labor Requirement	100% Manual	30% Supervision	70% Reduction
Mode Adaptation	Manual Adjustment	Automatic Switching	100% Automation

VI. DISCUSSION

Based on the research results, the development of an Automated IoT-Based Mushroom Greenhouse Control System for Real-Time Environmental Management via the Blynk Application, designed to control water misting based on greenhouse environmental conditions across all weather conditions, proved effective and did not adversely affect the growth of Bhutan oyster mushrooms. From testing the temperature and humidity sensors along with testing the operation of the automated control cabinet, it was found that the optimal temperature for oyster mushroom mycelium growth is approximately 28°C, according to the research of Ha Thi Hoa and Wang [48], which is consistent with the general cultivation temperature range of 25-35°C reported by Rahman et al. [43], where humidity also plays an important role in mushroom growth. According to the research of Dawadi et al. [49], excessively high humidity or moisture content can inhibit mycelial respiration and suppress fruiting body development, while overly dry conditions may result in malformed or hardened mushrooms, leading to growth inhibition. The disadvantages identified are the investment costs of using Blynk technology, training personnel to properly understand the use of new systems, and the long-term maintenance of these technologies. Additionally, to improve environmental monitoring in mushroom cultivation greenhouses, according to research by Wajiran et al. [50], the number of monitoring sensors should be increased by 2 units in a 4x6 square-meter area. However, this research still lacks a comparison of mushroom yields from implementing technology and from cultivation without technology.

VII. CONCLUSION AND FUTURE WORK

This research successfully developed a cost-effective and practical IoT-based automated mushroom greenhouse control system. The system maintained optimal humidity and temperature for

mushroom growth throughout all seasons. A key highlight is the system's ability to switch between Auto Mode and Timer Mode with 100% reliability during weather transitions. The implementation resulted in a 70% reduction in manual labor and a 67% improvement in the precision of humidity control compared to traditional methods. Future work should focus on a direct quantitative comparison of mushroom yields between technology-assisted and traditional cultivation. Additionally, increasing the density of monitoring sensors is recommended to provide more comprehensive data coverage. Integrating predictive analytics could further enhance resource management and production planning, supporting long-term, sustainable, and efficient agricultural practices.

ACKNOWLEDGMENTS

The researchers sincerely thank Thonburi University for the research grant and financial support. We also appreciate the Faculty of Science and Technology for providing development resources, and Wat Don Sai School, Ratchaburi, for providing the site to install and test the system in real-world conditions.

REFERENCES

- [1] P. Katepan, R. Dokmaithes, T. Mekhora, and T. Srithawan, "Cost and return analysis on jew's ear mushroom growing of large farmers association in Bang Phae District, Ratchaburi Province," *J. Manag. Sci. Nakhon Pathom Rajabhat Univ.*, vol. 8, no. 2, pp. 111-124, Dec. 2021, <https://doi.org/10.14456/jmsnpru.2021.35>
- [2] W. Suksawang, S. Dumnail, A. Wannakayont, and N. Lonkunthosh, "Development of a mushroom house with a combination of technology," *J. MCU Nakhondhat*, vol. 9, no. 11, pp. 181-195, Nov. 2022.
- [3] W. Phongngen, S. Petchhan, and R. Yajo, "Application with the internet of things technology control in smart farms mushroom," *Appl. IoT Technol. Control Smart Farms Mushroom*, vol. 5, no. 1, pp. 172-182, May 2018.
- [4] Thai Meteorological Department, Climate Center, Meteorological Development Division, *Weather conditions in Thailand 2023*, Bangkok, Thailand: Thai Meteorological Department, Jan. 2024.
- [5] P. Chant-aon and Y. Borisutdhi, "Agricultural entrepreneurship: The challenge occupation for new generations; X, Y and Z Generations," *J. Greater Mekhong Subreg. Res. Dev.*, vol. 1, no. 1, pp. 82-98, Apr. 2022.
- [6] S. Sajit, C. Srison, S. Phoiem, W. Philuek, and R. Chalongsuppunyoo, "The development of humidity and temperature control system in fairy mushroom cultivation house banphotphisai phitthayakhom school with a smartphone system," *EAI Journal*, vol. 2, no. 2, pp. 39-49, Dec. 2022.
- [7] B. Khammon and W. Sanchana, "A prototype of home lighting control system with Blynk Application," *J. Acad. Inf. Technol.*, vol. 5, no. 1, pp. 71-80, Jun. 2024.
- [8] K. T. Tun, H. M. Oo, and C. T. Nwe, "Automatic plant watering system using Arduino," *Int. J. Creat. Innov. Res. All Stud.*, vol. 2, no. 3, pp. 178-186, Jul. 2019.
- [9] M. Mediawan, M. Yusro, and J. Bintoro, "Automatic watering system in plant house using Arduino," *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 434, no. 012220, Dec. 2018.

- [10] P. S. Bains, R. K. Jindal, and H. K. Channi, "Modeling and designing of automatic plant watering system using Arduino," *Int. J. Sci. Res. Sci. Technol.*, vol. 7, no. 3, pp. 676-680, Dec. 2017, <https://doi.org/10.13140/RG.2.2.12722.09922>
- [11] J. Kumar, N. Gupta, A. Kumari, and S. Kumari, "Automatic plant watering and monitoring system using NodeMCU," in *Proc. 9th Int. Conf. Cloud Comput., Data Sci. Eng. (Confluence)*, 2019, pp. 545-550.
- [12] K. Wongwut, C. Chaisermvong, and D. Angamnuaesiri, "The development of smart farming system for sea lettuce cultured process," *Int. Sci. J. Eng. Technol. (ISJET)*, vol. 8, no. 2, pp. 47-53, Dec. 2024.
- [13] K. Thapwiroch, A. Kumlue, N. Saoyong, P. Taprasan, and S. Puengsungewan, "Easy-Mushroom Mobile Application Using the Internet of Things (IoT)," *Indones. J. Educ. Res. Technol.*, vol. 1, no. 1, pp. 1-6, Mar. 2021.
- [14] S. K. Nagothu, O. P. Kumar, and G. Anitha, "Weather based Smart watering system using soil sensor and GSM," in *Proc. Int. Conf. Eco-Friendly Comput. Commun. Syst.*, 2014, pp. 261-265, <https://doi.org/10.1109/Eco-friendly.2014.60>
- [15] R. A. R. Siregar et al., "Automatic watering systems in vertical farming using the Adaline algorithm," in *Proc. Int. Seminar of Science and Applied Technology (ISSAT 2020)*, 2020, pp. 429-435, <https://doi.org/10.2991/aer.k.201221.070>
- [16] S. Bhardwaj, S. Dhir, and M. Hooda, "Automatic plant watering system using IoT," in *Proc. 2nd Int. Conf. Green Comput. Internet Things (ICGCIoT)*, 2018, pp. 659-663.
- [17] W. Wongthai, S. Chanmee, and S. Lohawet, "An enhancement of an automatic plant watering system," in *Proc. 22nd Int. Comput. Sci. Eng. Conf. (ICSEC)*, 2018, pp. 1-4.
- [18] G. Eragamreddy and K. R. Sree, "Solar powered auto watering system for irrigation using embedded controller," in *Proc. Int. Conf. Energy, Commun., Data Anal. Soft Comput. (ICECDS)*, 2017, pp. 2424-2428.
- [19] M. Ojha, S. Mohite, S. Kathole, and D. Tarware, "Micro-controller-based automatic plant watering system," *Int. J. Comput. Sci. Eng. (IJCSE)*, vol. 5, no. 3, pp. 25-36, Apr. 2016.
- [20] D. Divani, P. Patil, and S. K. Punjabi, "Automated plant watering system," in *Proc. Int. Conf. Comput. Power, Energy Inf. Commun. (ICCPEIC)*, Apr. 2016, pp. 180-182.
- [21] M. Suresh et al., "Monitoring and Automatic Control of various Parameters for Mushroom Farming," *IOP Conf Ser Mater Sci Eng.*, vol. 1055, no. 1, p. 012011, Feb. 2021.
- [22] J. Jongpluempiti, P. Vengsungnle, S. Prapakarn, N. Pannucha-roenwong, and P. Punnok, "Supervisory control for wireless automatic environment control in oyster mushroom house," *FEAT J. Farm Eng. Autom. Technol.*, vol. 6, no. 1, pp. 40-49, Jan. 2020.
- [23] P. Isaranontakul and C. Rukphong, "The Android Application of control fog watering Indian Oyster Mushroom," *J. Inf. Sci. Technol.*, vol. 9, no. 1, pp. 1-8, Jan. 2019, <https://doi.org/10.14456/jist.2019.1>
- [24] K. -W. Lee, J. -O. Jeon, K. -J. Lee, Y. -H. Kim, C. -J. Lee, and M. -J. Jang, "Analysis of growth environment of *Flammulina velutipes* using the smart farm cultivation technology," *J. Mushrooms*, vol. 17, no. 4, pp. 197-204, Dec. 2019, <https://doi.org/10.14480/JM.2019.17.4.197>
- [25] T. Sornnen, N. Sukla, P. Luekajorn, and K. Daoden, "Developing Android applications with MIT App Inventor for smart mushroom farm systems," *Technol. Innov. URUJ.*, vol. 4, no. 2, pp. 15-23, Dec. 2021.
- [26] J. L. Chong, K. W. Chew, A. P. Peter, H. Y. Ting, and P. L. Show, "Internet of Things (IoT)-based environmental monitoring and control system for home-based mushroom cultivation," *Biosensors*, vol. 13, no. 1, pp. 1-24, Jan. 2023, <https://doi.org/10.3390/bios13010098>
- [27] Y. Surige et al., "IoT-based monitoring system for oyster mushroom farms in Sri Lanka," *Int. J. Res. Comput. (IJRC)*, vol. 1, no. 1, Jan. 2022, <https://doi.org/10.1109/ICAC54203.2021.9671112>
- [28] D. A. Setiawati, S. G. Utomo, Murad, and G. M. D. Putra, "Design of temperature and humidity control system on oyster mushroom plant house based on IoT," *IOP Conf. Ser.: Earth Environ. Sci.*, vol. 712, no. 1, Apr. 2021, <https://doi.org/10.1088/1755-1315/712/1/012002>
- [29] A. A. Shakir et al., "Design and implementation of SENSEPACK: An IoT-based mushroom cultivation monitoring system," in *Proc. Int. Conf. Electr. Comput. Commun. Eng. (ECCE)*, Apr. 2019, <https://doi.org/10.1109/ECACE.2019.8679183>
- [30] J. Islam et al., "Design and development of microcontroller-based wireless humidity monitor," *IOSR J. Electr. Electron. Eng. (IOSR-JEEE)*, vol. 13, no. 2, pp. 41-46, Mar. 2018, <https://url.in.th/TmvKf>
- [31] S. Kamonkunanon et al., "The development of factor control of production process for cool group of mushrooms in smart farm using Internet of Things technology," *Life Sci. Environ. J.*, vol. 24, no. 1, pp. 1-13, Jan. 2023.
- [32] A. Marzuki and S. Y. Ying, "Environmental monitoring and controlling system for mushroom farm with online interface," *Int. J. Comput. Sci. Inf. Technol.*, vol. 9, no. 4, pp. 17-28, Aug. 2017, <https://shorturl.asia/spID3>
- [33] I. S. P. Pratama and M. Z. Agung, "Temperature and humidity control system with long range in mushroom barn using fuzzy logic," *J. Teknol. Inf. Pendidik.*, vol. 16, no. 2, pp. 93-102, Jun. 2023.
- [34] A. D. Fakhruddin, N. F. A. Hakim, A. H. S. Budi, "Implementasi Protokol TCP dan UDP pada Sistem Monitoring dan Otomasi Rumah Jamur Berorientasi WSN," *TELKA*, vol. 9, no. 2, pp. 130-144, Jun. 2023.
- [35] G. Panumonwatee, S. Suwittayaporn, P. Kammimnual, and N. Thainthong, "The Application of Internet of Things (IoT) technology using MQTT for hydroponic gardening management: An approach to autonomous farming," *Sci. Technol. Nakhon Sawan Rajabhat Univ. J.*, vol. 15, no. 1, pp. 27-41, Jan. 2023.
- [36] S. Velliangiri, R. Sekar, and P. Anbhazhagan, "Using MLPA for smart mushroom farm monitoring system based on IoT," *Int. J. Netw. Virtual Organ.*, vol. 22, no. 4, pp. 334-347, Jan. 2020.
- [37] K. Bunluewong and O. Surinta, "Semi-automated mushroom cultivation house using Internet of Things," *Eng. Access*, vol. 7, no. 2, pp. 181-188, Jul. 2021, <https://doi.org/10.14456/mijet.2021.24>
- [38] S. Singh, S. Anand, Simran, and S. S. J., "Smart mushroom cultivation using IoT," *Int. J. Eng. Res. Technol. (IJERT)*, vol. 8, no. 11, pp. 247-251, Nov. 2020.
- [39] A. Subedi, A. Luitel, M. Baskota, and T. D. Acharya, "IoT-based monitoring system for white button mushroom farming," in *Proc. MDPI*, Jan. 2020, pp. 1-6.
- [40] M. A. Mohd Ariffin et al., "Automatic climate control for mushroom cultivation using IoT approach," in *Proc. IEEE 10th Int. Conf. Syst. Eng. Technol. (ICSET)*, 2020, pp. 123-128.
- [41] S. Chantamunee et al., "IoT for smart farm: A case study of the Lingzhi mushroom farm at Maejo University," in *Proc. Int. Joint Conf. Comput. Sci. Softw. Eng. (JCSSE)*, Nakhon Si Thammarat, Thailand, Sep. 2017.
- [42] C. Thirawanutpong and P. Yuttana, "Smart control in mushroom farms using internet of things and solar technology," *J. Ind. Technol. Thepsatri Rajabhat Univ.*, vol. 15, no. 2, pp. 102-111, Jul. 2020.
- [43] H. Rahman et al., "IoT enabled mushroom farm automation with Machine Learning to classify toxic mushrooms in Bangladesh," *J. Agric Food Res*, vol. 7, Mar. 2022.
- [44] E. Rohadi et al., "The design and evaluation of an automatic watering system using fuzzy Mamdani," *J. Phys.: Conf. Ser.*, vol. 1402, no. 2, no. 022088, Dec. 2019. <https://iopscience.iop/article/10.1088/1742-6596/1402/2/022088>
- [45] E. Boonchieng, O. Chieochan, and A. Saokaew, "Smart farm: Applying NodeMCU, IoT, NETPIE and LINE API for a Lingzhi mushroom farm in Thailand," *IEICE Trans. Commun.*, vol. E101-B, no. 1, pp. 16-23, Jan. 2018.

- [46] F. Masykur, A. Prasetyo, I. Widaningrum, A. F. Cobantoro, and M. B. Setyawan, "Application of MQTT protocol in the Internet of Things to monitor mushroom cultivation," in *Proc. 7th Int. Conf. Inf. Technol., Comput., Electr. Eng. (ICITACEE)*, Sep. 2020, pp. 135-139.
- [47] P. Khlaichom et al., "The development of automatic environmental control systems for straw mushroom farms based on Internet of Things," *Technol. Innov. URU J.*, vol. 5, no. 1, pp. 25-39, Jan. 2022.
- [48] H. T. Hoa and C.-L. Wang, "The Effects of Temperature and Nutritional Conditions on Mycelium Growth of Two Oyster Mushrooms (*Pleurotus ostreatus* and *Pleurotus cystidiosus*)," *Mycobiology*, vol. 43, no. 1, pp. 14-23, Mar. 2015, <https://doi.org/10.5941/MYCO.2015.43.1.14>
- [49] E. Dawadi, P. B. Magar, S. Bhandari, S. Subedi, S. Shrestha, and J. Shrestha, "Nutritional and post-harvest quality preservation of mushrooms: A review," *Heliyon*, vol. 8, no. 12, Art. no. e12093, Dec. 2022, <https://doi.org/10.1016/j.heliyon.2022.e12093>
- [50] Wajiran et al., "Control and real-time monitoring system for mushroom cultivation fields based on WSN and IoT," *Journal of Physics: Conference Series*, vol. 1655, no. 1, Nov. 2020, <https://doi.org/10.1088/1742-6596/1655/012003>



Chainarong Janthoom received the B.Sc. degree in information technology from Thonburi University, Bangkok, Thailand, in 2011, and the M.Sc. degree in information technology from King Mongkut's University of Technology North Bangkok, Bangkok, Thailand, in 2019. He is a Lecturer in the Department of Data Science and Information Technology, Thonburi University, Bangkok, Thailand.



Suphalak Batpho received the B.B.A. degree in business computer from Thonburi College of Technology, Thailand, in 2002, and the M.S. degree in information technology from North Bangkok College, Thailand, in 2005. She is currently pursuing a Ph.D. in management information systems at King Mongkut's University of Technology North Bangkok in Bangkok, Thailand. She is a Lecturer in the Department of Data Science and Information Technology, Thonburi University, Thailand.



Sittisak Thongsuk received the B.S. degree in computer science from Ramkhamhaeng University, Thailand, in 1992 and the M.Sc. degree in information technology from King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand, in 1998. He is currently pursuing the Ph.D. degree in business administration at Thonburi University, Bangkok, Thailand. He is a Lecturer in the Department of Data Science and Information Technology, Thonburi University, Bangkok, Thailand.



Maytagorn Thongkaobua is a Lecturer in the Department of Data Science and Information Technology, Thonburi University, Thailand. He received the B.B.A. degree in business computer from Sripatum University, Bangkok, Thailand, in 2007, and the M.Sc. degree in information technology from King Mongkut's University of Technology North Bangkok, Bangkok, Thailand, in 2012.

Cost Minimization in 3D Bin Packing under Orientation and Operational Constraints: A Case Study in E-Commerce Fulfillment

Anupong Thuengnaitham¹ and Salilathip Thippayakraisorn^{2*}

^{1,2}Faculty of Logistics and Transportation Management, Panyapiwat Institute of Management, Nonthaburi, Thailand

E-mail: anupongthu@pim.ac.th, salilathipthi@pim.ac.th*

Received: February 20, 2026 / Revised: April 7, 2026 / Accepted: April 20, 2026

Abstract—Inefficient parcel packing in e-commerce fulfillment causes excessive void space, inflated dimensional weight charges, and unnecessary dunnage costs. This issue is rightly classified as a three-dimensional bin packing problem (3D-BPP), which is NP-hard and computationally intractable at realistic scales. Current industry practice relies on manual, horizontal-only packing with predefined box sizes, thereby systematically overlooking opportunities to reduce costs through flexible orientation. While extensive literature addresses 3D-BPP, existing research typically applies operational constraints as post hoc checks after geometric optimization, and rarely combines discrete box selection with explicit logistics cost minimization. This study develops and evaluates an orientation-flexible packing framework that integrates fragility protection, weight precedence, and base-support stability into the heuristic placement process, rather than filtering out infeasible solutions after packing. The framework selects from a set of discrete, predefined box sizes and minimizes the total logistics cost, including packaging and dimensional weight charges. Three strategies—a horizontal-only baseline, First Fit Decreasing (FFD), and Best Fit (BF) with orientation flexibility, are systematically compared using simulation experiments on 600 real-world e-commerce orders from a personal care products fulfillment operation. Results demonstrate that BF achieves a 4.47% reduction in total logistics cost, 9.31% reduction in void space, and 9.31% reduction in dunnage cost relative to the baseline. At the same time, FFD is approximately eight times faster than BF, with moderate efficiency gains. Cross-validation confirms solution stability across heterogeneous subsets of orders, and sensitivity analyses verify robustness to variations in support thresholds, dimensional-weight pricing, and box-material costs. These findings provide actionable guidance for context-dependent

strategy selection, demonstrating that integrating operational constraints into packing heuristics do not degrade cost performance and remain computationally feasible for real-world e-commerce fulfillment.

Index Terms—Three-Dimensional Bin Packing, Parcel Packing Optimization, E-Commerce Logistics, Heuristic Algorithms, Transport Efficiency, Supply Chain Cost Management

I. INTRODUCTION

Global e-commerce expansion has fundamentally transformed supply chain dynamics, with operational efficiency at each stage of fulfillment becoming critical to profitability and customer satisfaction [1], [2]. Parcel packing is the process of placing items into shipping containers and is one of the most cost-significant activities in this ecosystem [3]. When packing is done inefficiently, it generates excessive void space, inflating packaging material consumption and disposal costs [4], while dimensional weight pricing penalizes oversized parcels by charging transportation costs proportional to package dimensions rather than actual weight [5]. For high-volume operations, cumulative packing inefficiencies translate into millions of dollars in unnecessary annual logistics expenditure.

Parcel packing is formally recognized as a three-dimensional Bin Packing Problem (3D-BPP), an NP-hard combinatorial optimization problem for which no polynomial-time algorithm guarantees an optimal solution at realistic scales [6]–[8]. Consequently, practitioners rely on heuristic approaches that sacrifice theoretical optimality for computational tractability [9]. Current industry practice involves manual labor guided by simple rules: selecting box sizes from predefined standard cartons and arranging items in fixed horizontal orientation [10]. This method creates systematic inefficiencies, restricting items to horizontal placement forces workers to select oversized containers when tall products consume excessive floor area, even

though smaller boxes would suffice if items could rotate. Moreover, manual packing typically disregards economic factors, basing decisions on geometric feasibility rather than explicitly considering box costs, dimensional weight charges, or dunnage expenses [11].

Despite the substantial body of literature on 3D-BPP that addresses exact algorithms, classical heuristics, and metaheuristics, several important gaps remain [7]. To begin with, most existing studies prioritize geometric packing efficiency, treating operational constraints such as product fragility, load-bearing capacity, and stacking stability as secondary considerations rather than embedding them as core components of the optimization model [10], [12]. Related to this, the discrete nature of box selection in real-world fulfillment operations is frequently overlooked. Many models either assume continuous container dimensions or frame the objective purely as minimizing bin count, neither of which reflects how carton selection actually works in practice [1].

A further gap is the lack of a method for comparing heuristic algorithms in real-world settings. Although First Fit Decreasing (FFD) and Best Fit (BF) are widely recognized as practical packing strategies, their relative performance in settings that simultaneously involve orientation flexibility, operational constraints, and discrete box selection has received limited empirical attention. Lastly, integration of packing decisions with logistics economics remains underexplored, with most models optimizing space utilization without incorporating packaging material costs, dimensional weight charges, or vehicle utilization impacts.

This study addresses these gaps through a simulation-based comparative analysis of orientation-flexible packing strategies under realistic operational constraints. Three objectives guide the investigation. The first is to develop a three-dimensional packing framework that allows full orientation flexibility while embedding operational constraints, including fragility protection, weight precedence, and stacking stability, directly into the optimization logic, alongside discrete box selection and cost-based objective functions. The second objective is to systematically compare the FFD and BF heuristics across three performance dimensions: Cost efficiency, space utilization, and computational tractability. The third is to quantify the economic consequences of permitting orientation flexibility relative to conventional horizontal-only packing, with particular attention to packaging material expenditures, carrier-imposed dimensional-weight charges, and the cost of protective void fill.

The simulation employs operational data from a personal care products e-commerce fulfillment operation. It evaluates three packing strategies:

A horizontal-only baseline representing current practice, and orientation-flexible FFD and BF implementations across 600 customer orders with heterogeneous items. All strategies operate under identical constraints, including discrete box selection (four predefined standard sizes), fragility-based stacking rules, weight precedence requirements, and minimum base-support-area constraints. Performance is measured across economic (total logistics cost, packaging cost, dunnage cost), spatial efficiency (void volume, packing density), and computational (runtime per order) metrics.

II. LITERATURE REVIEW

Three-dimensional bin packing has evolved from pure geometric optimization into a contested space where theoretical elegance confronts operational reality. The problem's NP-hard complexity initially drove research toward computational tractability, and accepting heuristic approximations over exact solutions became the pragmatic consensus [13], [14]. This shift enabled practical implementation at scale, yet introduced a subtler challenge: the optimization objectives themselves diverged from industrial requirements. Academic models optimized what could be easily formulated, volumetric efficiency, bin count minimization, while practitioners needed what remained difficult to encode, cost minimization under safety constraints, discrete box selection, and dimensional weight pricing. This misalignment persists as a defining tension in contemporary research.

Orientation flexibility emerged as a compelling solution pathway, with empirical evidence showing 5-15% volumetric improvements when items could rotate freely rather than remain fixed horizontally [15], [16]. The logic appeared straightforward, i.e., more geometric degrees of freedom enable denser configurations. Yet this geometric promise created an operational paradox. Allowing full rotational freedom maximized space utilization but violated fundamental handling principles, such as placing fragile items sideways, stacking heavy objects atop breakables, and products lacking adequate base support. The literature's response revealed an uncomfortable reality: Operational constraints were treated as post-optimization filters, secondary concerns to be checked after geometric solutions were generated [10]. This sequential approach optimizes geometry first, validates safety second, and proves computationally wasteful when constraint violations necessitate solution rejection and re-optimization, and is philosophically troubling for prioritizing abstract efficiency over tangible product protection.

The algorithm selection debate centers on a deceptively simple trade-off between First Fit Decreasing and Best Fit heuristics. FFD's greedy

strategy places each item in the first feasible location, has linear computational complexity, and is straightforward to implement [17], [18]. BF's exhaustive evaluation of all placement options yields only marginal improvement in superior packing density at substantially higher computational cost [19]. Conventional wisdom, largely derived from unconstrained continuous optimization contexts, favors BF when solution quality justifies an extended runtime. However, this recommendation rests on assumptions that have rarely been tested under realistic conditions. When discrete box catalogs limit container choices, when operational constraints restrict feasible placements, when real-time fulfillment demands rapid decisions, does BF's theoretical advantage persist or collapse under constraint pressure? The literature provides surprisingly limited empirical guidance on this fundamental implementation question [20], [21].

Recent advances in deep reinforcement learning have opened new pathways for online bin packing optimization. Multiple studies demonstrate that DRL-based approaches can outperform traditional heuristics while maintaining computational tractability for real-time applications [22], [23]. However, integrating physical-constraints stability validation, load-bearing requirements, and orientation restrictions within DRL frameworks remains an active research challenge [10]. The integration of heuristic methods with learning-based approaches shows particular promise, with hybrid frameworks achieving competitive results across diverse problem instances [15], [24].

Perhaps the most consequential disconnect involves economic optimization. Academic formulations predominantly minimize bin counts or maximize volumetric utilization metrics, offering intuitive geometric appeal but limited economic relevance [7], [10]. Industrial packing decisions, conversely, occur within explicit financial frameworks in which dimensional weight pricing formulas convert package dimensions into transportation charges, where discrete box catalogs assign differential material costs to size selections, and where void filling requirements create dunnage expenses proportional to unused space [25]–[27]. A configuration achieving 95% space utilization in a large box may incur a higher total cost than a configuration achieving 85% space utilization in a smaller box when dimensional weight penalties dominate material savings. This economic reality requires objective functions directly targeting cost minimization rather than treating volumetric efficiency as a proxy. Yet empirical validation of cost-aware algorithms under realistic pricing structures, actual box catalogs, and genuine operational data remains conspicuously absent from published research [28].

Constraint programming and integer linear programming approaches have demonstrated effectiveness in handling complex multidimensional packing scenarios with safety and stability requirements [29]–[31]. These methods provide provably optimal solutions for small instances but struggle to scale for large-scale real-world problems. Consequently, researchers have increasingly focused on hybrid approaches that combine exact optimization techniques with metaheuristic search strategies [24]–[33].

These tensions converge in a fundamental integration challenge: Can orientation flexibility, operational safety constraints, discrete box selection, and economic optimization coexist within computationally tractable frameworks? Recent attempts to embed constraints directly into optimization logic, rather than apply them post hoc, represent methodological progress, yet joint consideration remains incomplete. Models integrate fragility, orientation, or discrete economics, but rarely all simultaneously. This fragmentation reflects not merely research specialization but deeper uncertainty about whether these objectives fundamentally conflict or can be harmonized. The present study addresses this integration challenge directly by embedding operational constraints into orientation-flexible heuristics operating on discrete box catalogs with explicit cost functions. Rather than assuming constraint-efficiency trade-offs, we empirically assess whether such trade-offs exist under realistic conditions, using operational data to test whether the perceived tension between safety and optimization reflects genuine incompatibility or an implementation artifact. This positioning tests theoretical assumptions against operational reality, distinguishes the contribution from prior work that either sacrificed realism for theoretical tractability or implemented solutions without systematic performance validation.

III. METHODOLOGY

A. Research Design

This study employs computational simulation-based experimental design to evaluate heuristic packing strategies under realistic operational constraints. The methodology addresses three research objectives through controlled comparison: (1) develop an orientation-flexible framework integrating operational constraints with discrete box selection and cost optimization; (2) compare FFD and BF heuristics across cost efficiency, space utilization, and computational performance; and (3) quantify economic impacts of orientation flexibility relative to horizontal-only practices. Three packing strategies are evaluated: Orientation-flexible FFD, orientation-flexible BF, and a horizontal-only baseline representing current manual practice. All strategies process identical order

instances under identical constraints, enabling direct attribution of performance differences to algorithmic differences.

B. Case Context and Data Source

Empirical data derive from an operating e-commerce fulfillment company distributing personal care products (hair and scalp care). The operation processes high-volume orders containing multiple heterogeneous items with varying dimensions, weights, and levels of fragility. Packing decisions are constrained by dimensional weight-based pricing, in which transportation charges equal the greater of the actual weight or the volumetric weight (dimensional weight = volume/ δ , where $\delta = 5000 \text{ cm}^3/\text{kg}$). The operation utilizes four standardized box sizes: A ($14 \times 20 \times 6 \text{ cm}$, \$9), B ($17 \times 25 \times 9 \text{ cm}$, \$12), C ($20 \times 30 \times 11 \text{ cm}$, \$16), and D ($22 \times 35 \times 14 \text{ cm}$, \$20). Product catalog includes 8 SKUs with dimensions ranging 3.2-9.5 cm (length), 4.2-8.0 cm (width), 7.5-18.5 cm (height); weights 100-250g (average 213g, SD 58g); and fragility classifications distributed as 25% low (index 1-2), 50% medium (index 3), 25% high (index 4-5). This configuration reflects typical e-commerce fulfillment conditions, including order heterogeneity, discrete box catalogs, and dimensional weight pricing structures.

C. Mathematical Model and Problem Formulation

The three-dimensional bin packing problem with operational constraints is formulated as follows. Let $i \in I$ denote items in customer orders, where I is the total item count. Let $b \in B$ denote available box types. Let $k \in k_i$ denote feasible orientations for item i where $|k_i| = 6$ representing six possible three-dimensional rotations. Parameters include:

- (l_i, w_i, h_i) As original dimensions of item i ;
- (L_b, W_b, H_b) As internal dimensions of box b ;
- C_b As the packaging cost of box b ;
- f_i As fragility index of item i (dimensionless, higher values indicate greater fragility);
- m_i As mass of item;
- $\alpha = 0.80$ as the minimum base support ratio;
- δ As a dimensional weight divisor.

Objective Function: Minimize total logistics cost;

$$Z = \sum_{b \in B} (C_b + S_b)$$

where S_b represents shipping cost calculated via dimensional weight pricing.

$$S_b = \max \left(\sum_{i \in I_b} m_i, \frac{L_b \times W_b \times H_b}{\delta} \right)$$

This formulation captures the practical mechanism by which carriers charge based on either actual or volumetric weight, whichever is greater.

Constraints:

(1) Container Boundaries:

$$x_i + l'_{i,k} \leq L_b; y_i + w'_{i,k} \leq W_b; z_i + h'_{i,k} \leq H_b$$

where (x_i, y_i, z_i) denote placement coordinates and $(l'_{i,k}, w'_{i,k}, h'_{i,k})$ represent effective dimensions under orientation k .

(2) Non-Overlapping: Items $i \neq j$ in the same box must not occupy identical three-dimensional space, enforced via coordinate-based overlap checking.

(3) Fragility and Weight Precedence: To prevent product damage during handling and transportation, stacking decisions must respect both weight and fragility ordering. For item i placed above item j :

$$m_i \leq m_j \text{ and } f_i \leq f_j$$

This constraint ensures that heavier or more fragile items are not stacked on top of lighter or less fragile items.

(4) Base Support Stability: Stacking stability is further ensured through a minimum contact area constraint. Let A_{ij}^{contact} denote the contact area between item i (upper) and item j (lower), and let A_i^{base} denote the base area of item i . The following condition must hold:

$$\frac{A_{ij}^{\text{contact}}}{A_i^{\text{base}}} \geq \alpha \quad \text{where } \alpha = 0.80$$

This constraint ensures that at least 80% of an upper item's base area is supported by the item below.

D. Algorithm Selection Rationale

While recent literature explores deep reinforcement learning [22], [23] and exact methods including constraint programming and integer linear programming [29], [30], [31], this study focuses on classical constructive heuristics—First Fit Decreasing (FFD) and Best Fit (BF)—for three methodologically justified reasons.

Practical Deployability: Classical heuristics require no training data, hyperparameter tuning, or GPU acceleration. Deep reinforcement learning demands substantial training datasets and computational resources that may be unavailable in industrial e-commerce environments. Our focus on immediately deployable algorithms aligns with efforts to bridge theoretical optimization and practical fulfillment operations.

Computational Tractability: Exact methods (CP/IP) guarantee optimality but scale exponentially. Prior work shows CP/IP becomes prohibitive beyond 100-200 items [29], [30]. Our 600-order dataset (2-7 items per order) requires real-time solutions (<1 second for FFD, <5 seconds for BF). Exact methods would exceed these operational constraints by orders of magnitude.

Algorithmic Transparency: Deterministic heuristics provide clear decision paths facilitating operational debugging, quality assurance, and user trust. Warehouse

staff can trace decision logic and override when appropriate. Black-box approaches (neural networks, complex metaheuristics) obscure rationale, reducing operational acceptance where human workers retain final packing authority.

The traditional heuristic focus reflects a deliberate methodological decision prioritizing immediate industrial applicability, computational feasibility at scale, and operational transparency over algorithmic innovation.

E. Heuristic Packing Algorithms

First Fit Decreasing (FFD): Items are sorted in descending order by footprint area (length $\times$ width), with a secondary sort by weight and fragility. For each item, the algorithm sequentially attempts to place it in the smallest feasible box that satisfies all constraints. Upon finding the first feasible box-orientation combination, the algorithm accepts this placement without further evaluation. If no feasible placement exists across all boxes and orientations, the algorithm escalates to the next larger box size. Computational complexity is $O(n \log n)$ for sorting plus $O(nb)$ for placement, where n = items, b = box types. This efficiency enables real-time deployment in high-volume operations.

Best Fit (BF): The BF algorithm differs in its placement decision logic. For each item, all feasible box types and all feasible orientations are evaluated. The algorithm then selects the box-orientation combination that minimizes residual void volume in the packed box:

$$V_{void} = (L_{box} \times W_{box} \times H_{box}) - \sum (l'_{i,k} \times w'_{i,k} \times h'_{i,k})$$

By minimizing void space, the BF algorithm typically achieves higher space utilization than the FFD algorithm. This exhaustive evaluation typically achieves superior space utilization at substantially higher computational cost $O(nb \times 6)$ per item, where 6 represents orientation permutations.

Baseline (Horizontal-Only): The baseline strategy constrains all items to a single horizontal orientation (zero rotation) and selects the smallest box using the same greedy logic as FFD. All four

constraint categories are validated in the same way as the orientation-flexible strategies. Computational complexity is $O(nb)$. The baseline was simulated using the same framework as FFD and BF, ensuring that performance differences are attributable solely to orientation flexibility.

Fig. 1 and 2 illustrate the FFD and BF algorithmic workflows. FFD accepts the first feasible box-orientation combination, while BF evaluates all combinations and selects the one minimizing residual void volume. Both algorithms apply the same four constraint checks and escalate to the next larger box when placement fails.

F. Simulation Implementation

Fig. 3 illustrates the overall simulation workflow, consisting of six phases: input data loading, order-level processing across three strategies, packing with constraint validation, performance measurement, model validation, and output generation.

The computational model was implemented in Python 3.10 using Google Colaboratory (Colab), a cloud-based Jupyter notebook environment. Colab provides pre-installed scientific computing libraries, including Pandas for data manipulation, NumPy for numerical computation [34], Matplotlib for visualization [35], and Seaborn for statistical graphics [36]. The architecture employs a modular design with separate modules for constraint validation, box selection, item placement, and metric calculation. At each placement attempt, algorithms validate four constraint categories: (1) container boundaries ensuring items fit within box dimensions; (2) non-overlapping, verifying distinct items occupy separate spatial regions; (3) fragility and weight precedence maintaining safe stacking order; and (4) base support ensuring adequate stability. If constraint violations occur, algorithms increase the box size and retry placement. The simulation processes customer orders sequentially, applying designated algorithms to each order independently. Post-packing, performance metrics including total logistics cost, void volume, dunnage requirements, and computational runtime are calculated and recorded.

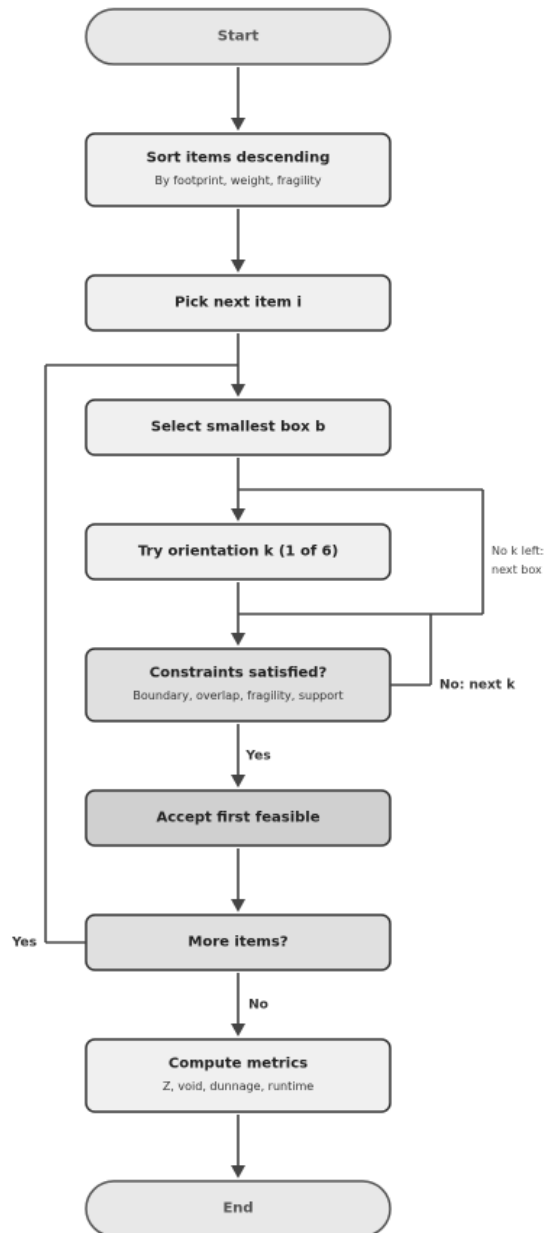


Fig. 1. Flowchart of the FFD algorithm. Items are sorted by footprint area (descending), then the first feasible box-orientation combination is accepted. Complexity: $O(n \log n)$ plus $O(nb)$.

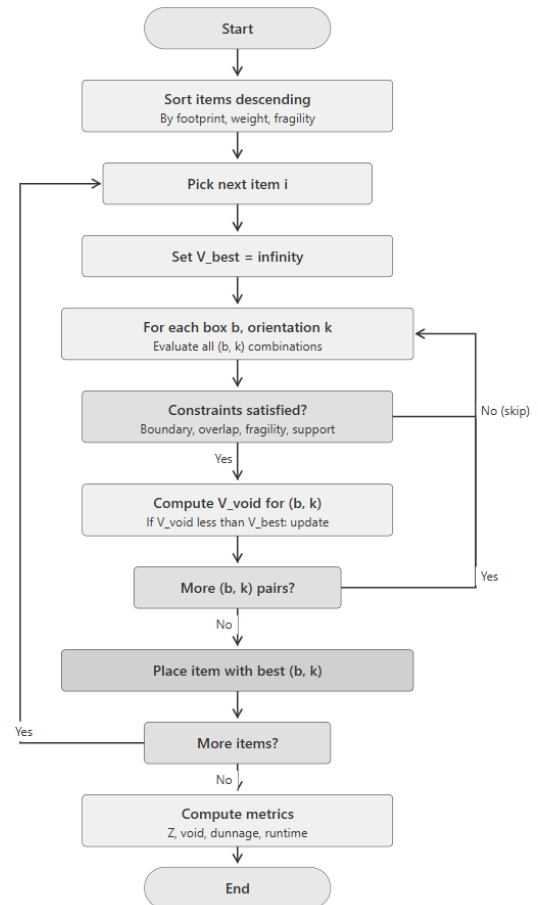


Fig. 2. Flowchart of the BF algorithm. All feasible box-orientation combinations are evaluated for each item, selecting the one that minimizes residual void volume. Complexity: $O(nb \times 6)$ per item.

G. Experimental Dataset

The framework was evaluated using 600 customer orders, each containing 2-7 items selected from the 8-SKU catalog. Orders reflect realistic e-commerce characteristics, including dimensional variation, weight distribution, and heterogeneity in fragility. All experiments used identical predefined box sizes to ensure comparability. Although the framework is generic, input parameters, item dimensions, weights, fragility indices, and order compositions are derived from operational data, thereby strengthening external validity and ensuring findings reflect realistic fulfillment behavior rather than idealized scenarios.

H. Performance Evaluation Metrics

Algorithm performance is evaluated through four metrics capturing economic efficiency, space utilization, and computational feasibility: (1) Total Logistics Cost (Z): Sum of packaging material costs and transportation charges representing primary economic metric enabling bottom-line financial impact comparison; (2) Void Space Volume (V_{void}): aggregate unused volume across boxes quantifying geometric efficiency; (3) Dunnage Material Cost: estimated cost of protective padding required for product safety reflecting trade-off between tight packing and adequate protection; and (4) Average Computational Runtime: CPU execution time per order measured in milliseconds capturing implementation feasibility for real-time deployment. These metrics provide a multidimensional evaluation considering solution quality, cost efficiency, and practical deployment constraints.

First, solution stability was evaluated using five-fold stratified cross-validation. The 600-order dataset was partitioned into five approximately equal subsets, stratified by order size to preserve heterogeneous order characteristics across folds. Each packing strategy is applied independently to all folds. One-way Analysis of Variance (ANOVA) was applied within each strategy to test whether mean performance metrics differed across folds.

Second, robustness to parameter uncertainty was assessed through One-At-A-Time (OAT) sensitivity analysis. Three key operational parameters were systematically varied while holding others constant: (i) the base support ratio (α), (ii) the dimensional weight divisor (δ), and (iii) box cost levels. Performance sensitivity was assessed based on total logistics cost, void space, dunnage requirements, and computational runtime.

IV. RESULTS

A. Experimental Dataset and Simulation Setup

The experimental dataset comprises 600 customer orders drawn from real operational data in the personal care products sector. Each order contains 2 to 7 items, with an average of 4.17 items per order, selected from a catalog of 8 SKUs. The products exhibit heterogeneous physical characteristics, including variations in dimensions, weight, and fragility levels, thereby reflecting realistic e-commerce fulfillment conditions.

For all experiments, a standardized set of four discrete box sizes commonly employed in e-commerce operations was used: A ($14 \times 20 \times 6$ cm), B ($17 \times 25 \times 9$ cm), C ($20 \times 30 \times 11$ cm), and D ($22 \times 35 \times 14$ cm). A dimensional weight divisor of $= 5000 \text{ cm}^3/\text{kg}$ and a minimum base support ratio of $= 0.80$ were applied consistently across all strategies. To ensure strict comparability, each packing algorithm processed the same sequence of order instances under identical operational constraints.

B. Comparative Performance Analysis

Table I presents comprehensive performance comparisons across three packing strategies: Baseline (horizontal-only manual rule), FFD (First Fit Decreasing with orientation flexibility), and BF (Best Fit with orientation flexibility). Results represent deterministic outcomes from simulation across all 600 orders. All baseline results were generated through the same simulation framework as FFD and BF, applying the manual rule logic (horizontal-only, smallest-box-first) programmatically to ensure controlled comparison.

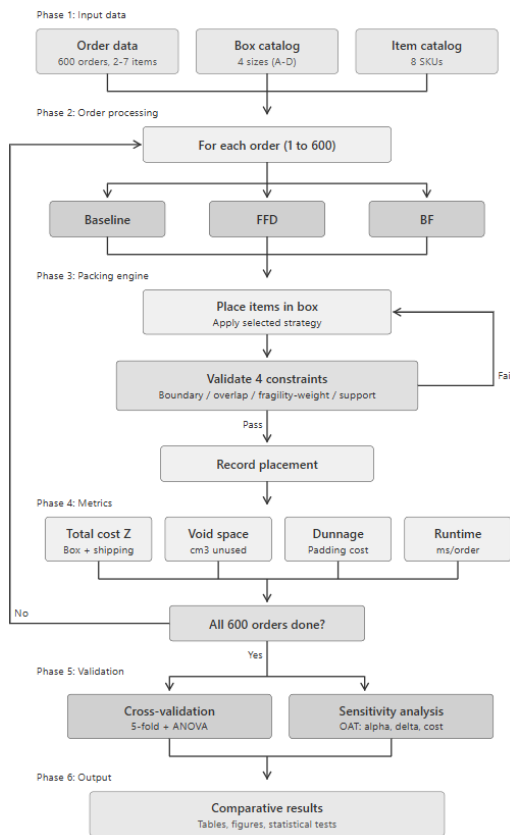


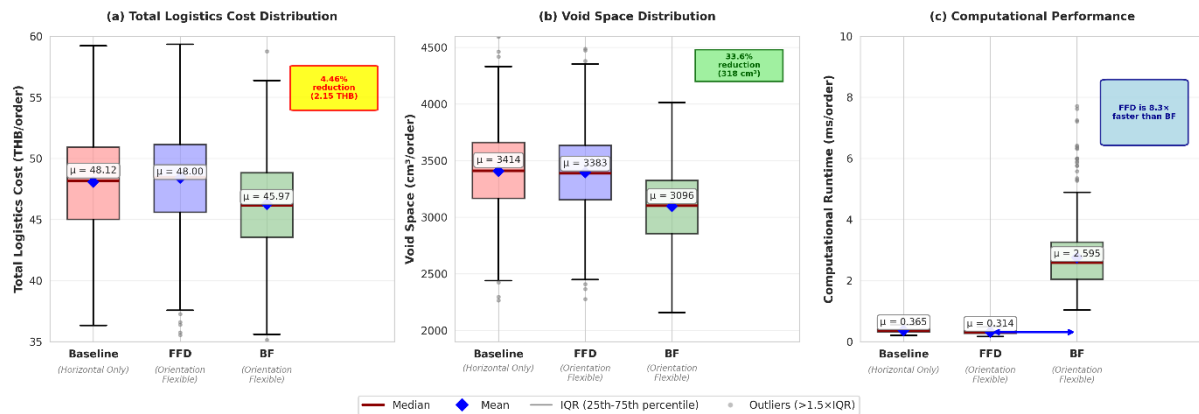
Fig. 3. Flowchart of the simulation process. The pipeline processes 600 orders through three packing strategies (Baseline, FFD, BF) under identical constraints, computes performance metrics, and validates results through cross-validation and sensitivity analysis.

I. Model Validation Procedures

To assess the reliability and robustness of the proposed packing optimization framework, two complementary validation procedures were employed: cross-validation and sensitivity analysis.

TABLE I
AVERAGE PERFORMANCE PER ORDER (N = 600)

Metric	Baseline	FFD	BF
Total Logistics Cost, Z (THB/Order)	48.12	48.00	45.97
Void Space (cm ³ /Order)	3,413.56	3,383.24	3,095.91
Dunnage Cost (THB/Order)	1.707	1.692	1.548
Runtime (ms/Order)	0.365	0.314	2.595



Note: Distributions based on simulation results. Whiskers extend to 1.5×IQR. Outliers shown as individual points. BF runtime variability reflects exhaustive evaluation in complex orders. ANOVA confirms no significant differences across 5-fold cross-validation (all $p > 0.05$).

Fig. 4. Performance distributions across 600 orders for Baseline (horizontal-only), FFD, and BF strategies: (a) total logistics cost, (b) void space, and (c) computational runtime. BF achieves a 4.4% cost reduction and a 33.6% reduction in void space relative to the Baseline. FFD is 8× faster than BF but yields only a minimal cost improvement

As shown in Table I, the results clearly demonstrate that orientation flexibility significantly improves in both economic and spatial performance relative to the horizontal-only baseline. Among the evaluated strategies, BF consistently achieves the lowest logistics cost, void space, and dunnage consumption per order. On average, BF reduces logistics costs by approximately 2.14 THB per order (4.4%) relative to the baseline. FFD also improves upon the baseline, though to a lesser extent. While FFD benefits from flexible orientation, it's greedy.

The acceptance rule limits its ability to fully exploit the available box capacity, resulting in performance outcomes that remain closer to Baseline than to BF in terms of cost and space utilization.

C. System-Level Economic Impact

To evaluate the operational significance of these improvements, Table II reports cumulative. Performance metrics aggregated across all 60 orders.

TABLE II
AGGREGATE PERFORMANCE ACROSS 600

Metric	Baseline	FFD	BF
Total Z (THB)	28,869.48	28,798.30	27,583.20
Total Void (cm ³)	2,048,133.14	2,029,943.14	1,857,543.14

At the system level, the economic benefits of BF become more pronounced. Relative to Baseline, BF achieves a total logistics cost reduction of 1,286.28 THB across 600 orders, accompanied by a reduction of approximately 190,590 cm³ in unused box volume and a decrease of more than 95 units in dunnage material consumption. Although BF incurs a higher computational cost, its total runtime remains within a feasible range for batch-based fulfillment planning. FFD yields only modest aggregate savings relative to Baseline, indicating that orientation flexibility alone is insufficient to generate substantial system-level benefits without an informed box-selection mechanism.

D. Decomposition of Performance Improvements

Table III summarizes the incremental gains of FFD_OF and BF_OF relative to the baseline. These results indicate that the dominant driver of cost reduction is improved space utilization rather than algorithmic overhead. The substantial reduction in Lower-dimensional-weight charges and the reduced need for protective dunnage. In contrast, FFD_OF yields only a marginal reduction in void, which explains its limited cost advantage. The void space under BF_OF directly translates into

TABLE III
PERFORMANCE DIFFERENCES RELATIVE TO BASELINE

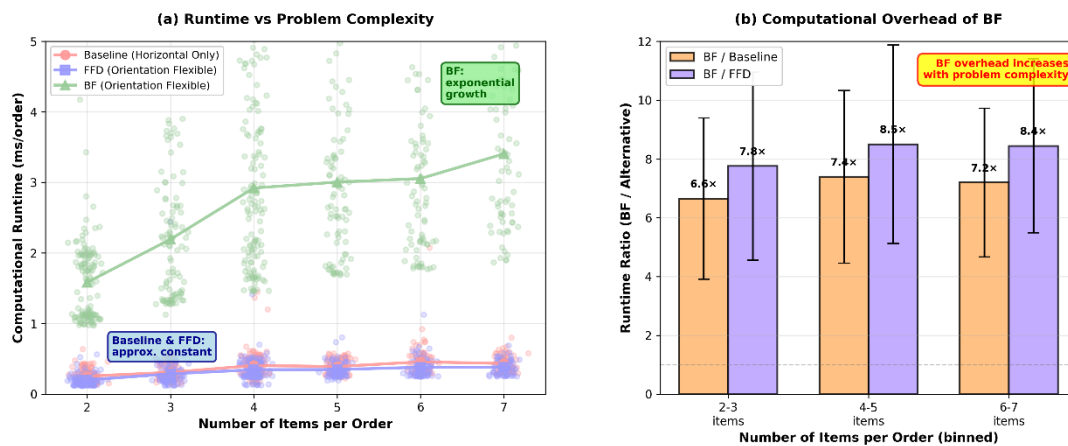
Metric	Baseline	FFD	BF
ΔZ (THB)	0	-71.18	-1,286.28
ΔVoid (cm ³)	0	-18,190	-190,590
$\Delta \text{Dunnage}$	0	-9.10	-95.30
Runtime (ms/Order)	0.365	0.314	2.595

E. Packing Efficiency, Computational Trade-offs, and Operational Implications

The experimental results show that improvements in packing efficiency arise from the interaction between geometric flexibility and box-selection logic, rather than from orientation flexibility alone. Allowing item rotation increases available packing space and enables configurations unattainable under horizontal-only constraints, resulting in substantial reductions in void space. However, the degree to which this expanded feasibility space translates into economic and spatial benefits depends critically on the algorithm's selection criterion.

While both FFD and BF utilize orientation flexibility, BF further exploits this flexibility by systematically evaluating alternative box-orientation combinations and selecting configurations that minimize residual void volume. As a result, BF achieves the largest reductions in void space and associated logistics costs. In contrast, FFD, which accepts the first feasible placement, captures only a portion of the potential gains, indicating that orientation flexibility is necessary but not sufficient to maximize packing efficiency.

These efficiency gains must be interpreted in the context of computational performance. A clear trade-off emerges between solution quality and runtime. FFD offers the fastest execution, processing orders approximately eight times faster than BF, making it well-suited for high-throughput environments with strict latency requirements. BF, while computationally more demanding, maintains average runtimes at the millisecond level, which remain acceptable for batch-oriented or cost-sensitive fulfillment operations.



Note: Individual points show order-level measurements; lines connect mean values. Whiskers extend to $1.5 \times \text{IQR}$. BF exhibits exponential runtime growth with order complexity, while Baseline and FFD remain approximately constant across all problem sizes.

Fig. 5. Computational Scaling Analysis Across Problem Sizes. (a) Runtime vs problem complexity (b) Computational overhead of BF

Fig. 5 illustrates computational scaling behavior across problem sizes ranging from 2 to 7 items per order. As shown in Fig. 5a, BF runtime increases exponentially with increasing order complexity, rising from 1.58 ms for 2-item orders to 3.41 ms for 7-item orders. In contrast, both Baseline and FFD strategies maintain approximately constant runtime (0.26-0.46 ms and 0.20-0.38 ms, respectively) across all order sizes examined. Despite their theoretical complexities being $O(n \log n)$ for sorting, $O(nb)$ for FFD, and $O(nb)$ for Baseline, the limited range of order sizes in this dataset (2-7 items) means the growth factor is negligible in practice. The difference between sorting 2 items and 7 items is negligible at the millisecond scale. This consistent observation suggests that the computational cost of Baseline and FFD is practically

unaffected by order complexity, given typical e-commerce order sizes. This supports their use in real-time systems.

Fig. 5b quantifies this computational disadvantage through runtime ratios. BF is consistently 6.6-7.2 $\times$ slower than Baseline and 7.6-8.5 $\times$ slower than FFD across all order-size bins. Notably, this overhead increases slightly with problem complexity: For 2-3-item orders, BF is 6.6 $\times$ slower than Baseline, increasing to 7.2 $\times$ for 6-7-item orders. This pattern confirms that BF's exhaustive orientation-evaluation approach becomes increasingly costly as the solution space expands.

These results have significant operational implications. For warehouses mainly handling small orders (≤ 4 items), BF's 2-3 ms overhead may be

acceptable if cost optimization is paramount. However, for facilities handling larger orders or requiring high throughput (>100 orders/hour), FFD's superior computational efficiency becomes critical, enabling real-time packing decisions without increasing costs.

These findings indicate that both orientation-flexible strategies are operationally viable; however, the choice of strategy depends on organizational priorities. Systems emphasizing responsiveness and throughput may prefer FFD, while operations focusing on cost reduction and material efficiency may benefit more from BF. This integrated perspective emphasizes that packing performance, computational feasibility, and operational objectives must be jointly considered when deploying algorithmic packing solutions in real-world e-commerce fulfillment systems.

F. Model Validation Results

This section reports the results of the model validation procedures, including cross-validation and sensitivity analysis, to evaluate the stability and robustness of the proposed packing optimization framework.

Cross-Validation Results

To examine solution stability, the 600-order dataset was divided into five stratified folds of approximately equal size. Each packing strategy was evaluated independently on all folds, and order-level performance metrics, including total logistics cost (Z), void space, and computational runtime, were recorded.

One-way Analysis of Variance (ANOVA) was performed for each strategy to test whether mean performance differed significantly across folds. The results show no statistically significant differences across subsets for any strategy or performance metric (all p-values > 0.05). Specifically, p-values ranged from approximately 0.62 to 0.99 across total logistics cost, void space, and runtime. The consistently low F-statistics suggest that between-fold variability is negligible relative to within-fold variability. These findings confirm that the observed performance outcomes are not attributable to specific subsets of customer orders, but rather reflect stable algorithmic behavior under heterogeneous order compositions.

Constraint Feasibility Validation: All 600 orders achieved feasible packing solutions under the three strategies (Baseline, FFD, BF) at baseline constraint settings ($\alpha=0.80$, fragility-based orientation restrictions, weight precedence enforcement). Post-solution verification confirmed 100% compliance with operational constraints: Fragile items were placed upright, weight precedence was maintained (no light items supporting heavier items), and base support ratios exceeded the 80% threshold across all solutions. This high feasibility rate indicates that the constraint severity strikes an appropriate balance: Strict enough to ensure operational safety yet permissive enough to maintain algorithmic tractability, without requiring manual intervention or constraint relaxation.

Sensitivity Analysis Results

Sensitivity analysis was conducted using a One-At-A-Time (OAT) approach to assess robustness to key operational parameters.

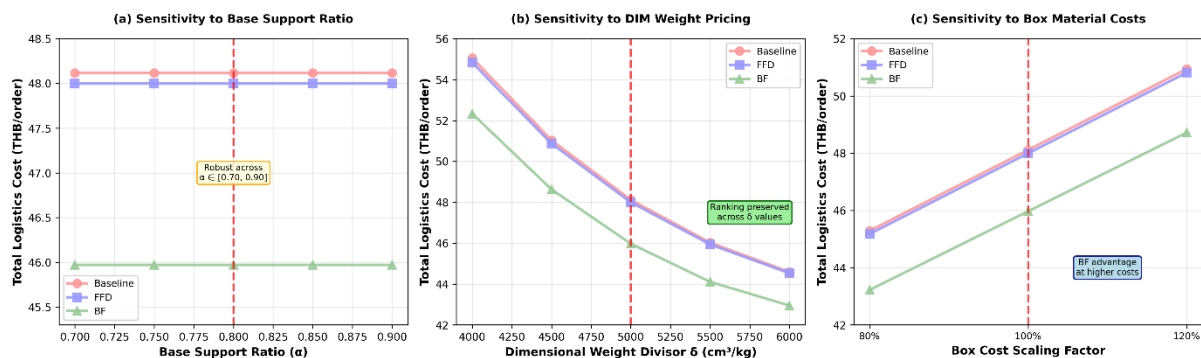


Fig. 6. Sensitivity analysis showing robustness to operational parameter variations: (a) base support ratio α , (b) dimensional weight divisor δ , and (c) box material costs. Relative performance ranking (BF > FFD > Baseline) is preserved across all parameter ranges, confirming framework robustness.

Sensitivity to Base Support Ratio (α)

Fig. 6a examines sensitivity to the base support ratio (α), which governs minimum contact area requirements for fragile item placement. Across the range $\alpha \in [0.70, 0.90]$ ($\pm 12.5\%$ of baseline $\alpha = 0.80$), all three strategies exhibit perfect stability with zero cost variation. Specifically, Baseline maintains 48.12 THB/order, FFD maintains 48.00 THB/order, and BF maintains 45.97 THB/order across all five α values tested. This remarkable invariance indicates that the framework's performance is independent of the strictness of constraints within plausible operational ranges. The consistent 2.15 THB/order gap between BF and Baseline confirms that strategic superiority persists regardless of whether facilities adopt conservative ($\alpha = 0.85-0.90$) or moderate ($\alpha = 0.70-0.75$) fragility protection policies. This outcome challenges the common assumption that tighter operational constraints necessarily reduce solution quality and demonstrates that our constraint-aware packing logic maintains efficiency without incurring performance penalties.

Sensitivity to Dimensional Weight Divisor (δ)

Sensitivity to dimensional weight pricing, as shown in Fig. 6b, assesses the robustness variations in carrier policy. Across δ values ranging from 4000 to 6000 cm^3/kg ($\pm 20\%$ of baseline $\delta = 5000$), all strategies exhibit expected linear response. At $\delta = 4000$ (high-dimensional weight penalty), costs reach 55.05 THB/order for Baseline, 54.83 THB/order for FFD, and 52.33 THB/order for BF, representing a 14.4% increase relative to baseline costs due to heightened volumetric charges. At $\delta = 6000$ (low penalty), costs decrease to 44.59 THB/order (Baseline), 44.53 THB/order (FFD), and 42.95 THB/order (BF), a 7.4% reduction. Critically, the relative ranking (BF > FFD > Baseline) remains unchanged throughout this range, with BF maintaining a consistent 2.1-2.7 THB/order advantage over Baseline. The linear relationship between δ and total cost ($R^2 > 0.99$ across all strategies) enables straightforward cost projections under varying carrier agreements, facilitating informed carrier selection and contract negotiation.

Sensitivity to Box Cost Scaling

Fig. 6c examines sensitivity to changes in box material cost by evaluating scenarios at 80%, 100%, and 120% of the baseline packaging cost. At 80% baseline, total logistics costs are 45.29 THB/order (baseline), 45.18 THB/order (FFD), and 43.22 THB/order (BF), yielding a BF cost advantage of 2.07 THB/order (4.57%) over baseline. At 100% baseline costs, this advantage is 2.14 THB/order (4.45%). At 120% costs, the advantage increases to 2.22 THB/order (4.36%), a 7.2% increase in absolute savings relative to the 80% scenario. This amplification pattern provides an important operational insight: BF's void reduction capabilities become increasingly

valuable in absolute terms when packaging materials are expensive, even though the percentage advantage remains relatively stable (4.36-4.57%). This finding provides a practical decision criterion: Facilities facing premium packaging costs ($\geq 110\%$ of the baseline) realize greater absolute per-order savings from BF implementation and should prioritize this strategy, while facilities with discounted materials ($< 90\%$ of the baseline) may find FFD's superior computational efficiency sufficient for their cost structure.

Collectively, these sensitivity analyses demonstrate three key properties of the framework. First, constraint-invariant performance (Fig. 6a) validates the consistency of recommendations across facilities with different safety requirements. Second, predictable linear responses to pricing variations (Fig. 6b) enable reliable cost forecasting under diverse carrier policies. Third, value amplification at premium costs (Fig. 6c) provides actionable guidance for context-dependent strategy selection. The preservation of relative performance rankings (BF > FFD > Baseline) across all parameter combinations confirms that the framework maintains robust performance across diverse e-commerce operating conditions without requiring extensive recalibration.

V. DISCUSSION

The findings reveal that orientation flexibility is a primary driver of packing efficiency improvements, in packing efficiency enabling algorithms to exploit configurations inaccessible under horizontal-only constraints, as noted in [9]. This flexibility yields a substantial reduction in void space, directly lowers dimensional-weight charges, and dunnage requirements. Importantly, economic benefits derive primarily from geometric efficiency rather than algorithmic sophistication, as cost reductions are closely tied to spatial compaction, indicating that geometric exploitation is the dominant mechanism driving savings in dimensional-weight-based pricing environments [37] where inappropriate box selection and item placement lead to higher logistics costs, wasted space, and reduced sustainability. This study aims to minimize the number of boxes used per order while improving space utilization by formulating the problem as a 3D Bin Packing Problem (3D-BPP).

A comparative analysis reveals a divergence between first-fit and best-fit heuristics when orientation flexibility is present. While both outperform baselines, best-fit consistently achieves superior outcomes by explicitly minimizing residual void via exhaustive evaluation of box-orientation combinations. However, this advantage diminishes in highly constrained solution spaces with limited box catalogs, where greedy heuristics may approximate optimal solutions reasonably well. This convergence

highlights structural properties of discrete packaging environments where catalog limitations naturally bound combinatorial complexity.

Results demonstrate clear trade-offs between solution quality and computational efficiency. Best-fit with orientation flexibility achieves the greatest economic and spatial efficiency but requires longer computation time, while first-fit offers near-baseline speed with moderate gains. Crucially, even intensive strategies operate within millisecond-level runtimes, reframing the trade-off from feasibility versus infeasibility to degrees of optimization. This enables heuristic selection tailored to operational priorities without compromising real-world deployability.

Integration of operational safety constraints, fragility protection, weight precedence, and base support imposes only marginal performance penalties. The constrained framework eliminates all safety violations at minimal cost and minimal increase in void space, demonstrating that operational realism and optimization performance are not mutually exclusive. This challenges assumptions that practical constraints necessarily degrade solution quality and underscores the value of embedding domain knowledge directly into decision logic.

From a managerial perspective, substantial cost savings can be achieved through modest changes in packing logic. For high-volume fulfillment centers, small per-order savings add up to significant annual reductions, while reduced dunnage usage yields secondary benefits, including lower material costs, reduced packing labor, and improved sustainability. Different strategies suit different contexts: first-fit with orientation flexibility is optimal for high-throughput environments with stringent latency requirements, while best-fit is preferable in cost-sensitive operations where computation time is justified by greater savings. Managers can therefore adopt context-dependent strategy selection rather than universal solutions.

The observed migration toward smaller boxes under orientation-flexible strategies highlights interactions between algorithm design and packaging policy. Expanding box catalogs may amplify best-fit benefits by increasing the granularity of choice, suggesting that investments in algorithmic optimization and packaging design should be considered jointly. More diverse catalogs unlock additional value from advanced heuristics, while improved heuristics increase the utilization efficiency of existing assets.

Millisecond-level runtimes indicate compatibility with real-world fulfillment systems, and modular design facilitates integration into existing warehouse or order management systems without extensive infrastructure changes. Framework robustness to data partitioning and parameter variations enables reliable deployment across diverse demand patterns and operating conditions. These findings collectively

demonstrate that orientation flexibility, when combined with informed box selection and realistic operational constraints, yields meaningful improvements in e-commerce packing efficiency, providing both theoretical insight into discrete packing problems and actionable guidance for balancing cost, efficiency, and operational safety.

VI. CONCLUSION

This study proposed and evaluated an orientation-flexible packing optimization framework integrating realistic operational constraints with discrete box selection and cost minimization. Using simulation-based experiments on 600 real-world e-commerce orders, the framework systematically compared a horizontal-only baseline with two orientation-flexible heuristics: First Fit Decreasing and Best Fit. Results demonstrate that orientation flexibility significantly improves packing efficiency across economic, spatial, and material dimensions, consistently reducing total logistics cost, void space, and dunnage consumption. Best Fit achieves the greatest overall efficiency, while First Fit offers a favorable balance between gains and computational speed. Integration of operational safety constraints, fragility protection, weight precedence, and base support eliminates unsafe configurations with only marginal performance penalties, confirming that practical requirements and optimization objectives can be aligned within unified algorithmic frameworks. Cross-validation and sensitivity analyses confirm robustness across heterogeneous order subsets and logical responses to parameter variations, including support thresholds, carrier pricing, and box costs, establishing practical viability and generalizability for real-world fulfillment operations.

Despite contributions, several limitations warrant acknowledgment. The experimental setting relies on a discrete catalog of four predefined box sizes; while reflecting common practice, more granular assortments may alter the relative performance of heuristics, particularly the best-fit advantage. The packing model employs approximate height-map-based support evaluation rather than full three-dimensional contact mechanics, which may not capture complex interactions from irregular geometries or deformable materials. Simulations assume deterministic item dimensions, weights, and fragility indices derived from historical data, yet measurement uncertainty, packaging tolerances, and handling variability introduce deviations not explicitly modeled. Computational experiments were conducted in single-threaded Python environments; while runtimes proved sufficient for deployment, alternative implementations or hardware configurations could affect scalability and real-time applicability.

Several research avenues emerge from this work. Extending the framework to larger, more diverse box catalogs could illuminate how packaging assortment design interacts with heuristic selection, with joint optimization of box assortment and packing strategy representing promising directions. Future studies may incorporate detailed physical modeling, including partial material deformability, irregular item shapes, or stochastic stability constraints to enhance realism when product geometry deviates from cuboid assumptions. Integrating dynamic order arrival with real-time decision-making would enable evaluation in streaming or online fulfillment environments, using hybrid approaches that combine fast heuristics with periodic optimization-based refinement to balance solution quality and responsiveness. Finally, future research may explore broader sustainability objectives, including carbon emissions, material recyclability, and reverse logistics, and embed environmental performance metrics alongside economic objectives to align packing optimization with emerging sustainability goals in e-commerce logistics.

REFERENCES

- [1] N. Alherimi, A. Saihi, and M. Ben-Daya, "A systematic review of optimization approaches employed in digital warehousing transformation," *IEEE Access*, vol. 12, pp. 145809-145831, Aug. 2024, <https://doi.org/10.1109/ACCESS.2024.3463531>
- [2] I. Zennaro, S. Finco, M. Calzavara, and A. Persona, "Implementing e-commerce from logistic perspective: Literature review and methodological framework," *Sustainability*, vol. 14, no. 2, pp. 1-37, Jan. 2022.
- [3] L. Duan et al., "A multi-task selected learning approach for solving 3D flexible bin packing problem," in *Proc. 18th Int. Conf. Auton. Agents Multi-Agent Syst (AAMAS)*, 2019, pp. 1386-1394, <https://dl.acm.org/doi/10.5555/3306127.3331847>
- [4] H. Zghair and R. G. Konathala, "Using AI for optimizing packing design and reducing cost in e-commerce," *AI*, vol. 6, no. 7, p. 146, Jul. 2025, <https://www.mdpi.com/2673-2688/6/7/146>
- [5] J. Ampuja, "Dimensional weight: Optimize packaging and savemoney," *Logist. Manag.*, vol. 54, no. 1, pp. 32-35, Jan. 2015.
- [6] J. Yang et al., "A genetic algorithm with lower neighborhood search for the three-dimensional multiorder open-size rectangular packing problem," *Int. J. Intell. Syst.*, vol. 2024, pp. 1-18, May 2024.
- [7] Y. Yang, Z. Wu, X. Hao, H. Liu, and M. Qi, "Two-layer heuristic for the three-dimensional bin design and packing problem," *Eng. Optim.*, vol. 56, no. 10, pp. 1601-1638, Oct. 2024, <https://doi.org/10.1080/0305215X.2023.2269868>
- [8] X. Wang, L. Gong, H. Zhao, B. Li, and M. Tian, "A 3D offline packing algorithm considering cargo orientation and stability," *Int. J. Distrib. Sens. Netw.*, vol. 2023, pp. 1-14, Mar. 2023, <https://doi.org/10.1155/2023/5299891>
- [9] M. Kasem Alqudah, D. Pambudi, and M. Z. Zakaria, "3D box packing with heuristics and metric analytics," *Int. J. Distrib. Sens. Netw.*, vol. 2, no. 2, pp. 56-66, Jun. 2025, <https://ejournal.gomit.id/index.php/ijaa/ml/article/view/409>
- [10] J. Yang et al., "Optimizing e-commerce warehousing through open dimension management in a three-dimensional bin packing system," *PeerJ Comput. Sci.*, vol. 9, p. e1613, Oct. 2023, <https://doi.org/10.7717/peerj-cs.1613>
- [11] T. Wang, Q. Hu, and A. Lim, "An exact algorithm for two-dimensional vector packing problem with volumetric weight and general costs," *Eur. J. Oper. Res.*, vol. 300, no. 1, pp. 20-34, Jul. 2022, <https://doi.org/10.1016/j.ejor.2021.10.011>
- [12] E. E. Bischoff, "Three-dimensional packing of items with limited load bearing strength," *Eur. J. Oper. Res.*, vol. 168, no. 3, pp. 952-966, Feb. 2006, <https://doi.org/10.1016/j.ejor.2004.04.037>
- [13] M. Haouari and M. Serairi, "Heuristics for the variable sized bin-packing problem," *Comput. Oper. Res.*, vol. 36, no. 10, pp. 2877-2884, Oct. 2009, <https://www.sciencedirect.com/science/article/abs/pii/S0305054808002748>
- [14] S. Yang et al., "Heuristics integrated deep reinforcement learning for online 3D bin packing," *IEEE Trans. Autom. Sci. Eng.*, vol. 21, no. 1, pp. 939-950, Jan. 2024, <https://doi.org/10.1109/TASE.2023.3235742>
- [15] C.-C. Wong, T.-T. Tsai, and C.-K. Ou, "Integrating heuristic methods with deep reinforcement learning for online 3D bin-packing optimization," *Sensors*, vol. 24, no. 16, p. 5370, Aug. 2024, <https://doi.org/10.3390/s24165370>
- [16] Z. Kang, Y. Guan, J. Wang, and P. Chen, "Research on genetic algorithm optimization with fusion tabu search strategy and its application in solving three-dimensional packing problems," *Symmetry*, vol. 16, no. 4, p. 449, Apr. 2024, <https://doi.org/10.3390/sym16040449>
- [17] B. Lin et al., "A pattern-based algorithm with fuzzy logic bin selector for online bin packing problem," *Expert Syst. Appl.*, vol. 249, p. 123515, Sep. 2024, <https://doi.org/10.1016/j.eswa.2024.123515>
- [18] C. Munien and A. E. Ezugwu, "Metaheuristic algorithms for one-dimensional bin-packing problems: A survey of recent advances and applications," *J. Intell. Syst.*, vol. 30, no. 1, pp. 636-663, Apr. 2021, <https://doi.org/10.1515/jisys-2020-0117>
- [19] T.-H. Nguyen, V.-T. Tran, Q. Doan, and T.-T. Mac, "A novel heuristic algorithm for online 3D bin packing," in *Proc. Int. Conf. Control, Autom. Syst. (ICCAS)*, 2021, pp. 1993-1997, <https://doi.org/10.23919/ICCAS52745.2021.9649790>
- [20] A. V. Puche and S. Lee, "Online 3D bin packing reinforcement learning solution with buffer," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 8902-8909, <https://doi.org/10.1109/IROS47612.2022.9982095>
- [21] J. Zhang and T. Shuai, "Online three-dimensional bin packing: A DRL algorithm with the buffer zone," *Found. Comput. Decis. Sci.*, vol. 49, no. 1, pp. 63-74, Feb. 2024, <https://doi.org/10.2478/fcds-2024-0005>
- [22] H. Zhao, Q. She, C. Zhu, Y. Yang, and K. Xu, "Online 3D bin packing with constrained deep reinforcement learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 1, pp. 741-749, May 2021, <https://doi.org/10.1609/aaai.v35i1.16155>
- [23] J. Jia, H. Shang, and X. Chen, "Robot online 3D bin packing strategy based on deep reinforcement learning and 3D vision," in *Proc. IEEE Int. Conf. Netw., Sens. Control (ICNSC)*, 2022, pp. 1-6, <https://doi.org/10.1109/ICNSC55942.2022.10004170>
- [24] G. Tresca, G. Cavone, R. Carli, A. Cerviotti, and M. Dotoli, "Automating bin packing: A layer building matheuristics for cost effective logistics," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 3, pp. 1599-1613, Jul. 2022, <https://doi.org/10.1109/TASE.2022.3177422>
- [25] A. A. Ananno and L. Ribeiro, "A multi-heuristic algorithm for multi-container 3-D bin packing problem optimization using real world constraints," *IEEE Access*, vol. 12, pp. 42105-42130, Mar. 2024, <https://doi.org/10.1109/ACCESS.2024.3378063>
- [26] P. Thapatsuwan, W. Thapatsuwan, and C. Kulworatit, "A heuristic enhancing artificial immune system for three-dimensional loading capacitated vehicle routing problem," *J. Appl. Eng. Technol. Sci.*, vol. 6, no. 2, pp. 1019-1039, Apr. 2025, <https://doi.org/10.37385/jaets.v6i2.6295>
- [27] N. T. X. Hoa, "Optimizing container fill rates for the textile and garment industry using a 3D bin packing approach," *HighTech Innov. J.*, vol. 5, no. 2, pp. 462-478, Jun. 2024, <https://doi.org/10.28991/HIJ-2024-05-02-017>

- [28] J. Zhang and T. Shuai, "Online three-dimensional bin packing: A DRL algorithm with the buffer zone," *Found. Comput. Decis. Sci.*, vol. 49, pp. 63-74, Feb. 2024, <https://doi.org/10.2478/fcds-2024-0005>
- [29] D. Pisinger and M. Sigurd, "Using decomposition techniques and constraint programming for solving the two-dimensional bin-packing problem," *INFORMS J. Comput.*, vol. 19, no. 1, pp. 36-51, Feb. 2007, <https://doi.org/10.1287/ijoc.1060.0181>.
- [30] G. Jaykrishnan and A. Levin, "APTAS for bin packing with general cost structures," *Theor. Comput. Sci.*, vol. 1055, p. 115510, Nov. 2025, <https://doi.org/10.1016/j.tcs.2025.115510>
- [31] Y. Jiang, Z. Cao, and J. Zhang, "Learning to solve 3-D bin packing problem via deep reinforcement learning and constraint programming," *IEEE Trans. Cybern.*, vol. 53, no. 5, pp. 2864-2875, May 2023, <https://doi.org/10.1109/TCYB.2021.3121542>
- [32] S. Krisadee and W. Jindaluang, "An improvement of a heuristic algorithm for 3D bin-packing problem," in *Proc. Int. Comput. Sci. Eng. Conf. (ICSEC)*, 2024, pp. 1-6, <https://doi.org/10.1109/ICSEC62781.2024.10770638>
- [33] W. Zahrouni and H. Kamoun, "Exact and approximate size reduction for the bin packing problem with conflicts," *Intell. Decis. Technol.*, vol. 19, no. 3, pp. 1822-1836, Jun. 2025.
- [34] C. R. Harris et al., "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357-362, Sep. 2020, <https://doi.org/10.1038/s41586-020-2649-2>
- [35] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Comput. Sci. Eng.*, vol. 9, no. 3, pp. 90-95, 2007, <https://ieeexplore.ieee.org/document/4160265>
- [36] M. L. Waskom, "Seaborn: Statistical data visualization," *J. Open Source Softw.*, vol. 6, no. 60, p. 3021, Apr. 2021, <https://doi.org/10.21105/joss.03021>
- [37] W. W. Adimukti and S. A. Sanjaya, "Minimizing packaging boxes in online retail fulfillment using multi-strategy heuristics and extreme-point based 3D bin packing," in *Proc. Int. Conf. Smart Cities, Autom. Intell. Comput. Syst. (ICON-SONICS)*, 2025, pp. 158-163, <https://doi.org/10.1109/ICON-SONICS67145.2025.11309244>



from Huazhong University of Science and Technology, Wuhan, China. His research interests include production and operations management, supply chain management, and humanitarian logistics.



Salilathip Thippayakraisorn is an Assistant Professor and Dean at the Faculty of Logistics and Transportation Management, Panyapiwat Institute of Management. She earned her Ph.D. in Business Administration from Kasetsart University, Thailand. Her research interests are risk management, supply chain management, social commerce, and logistics in e-commerce.

Effect of Intermittent Photoperiod on Yield and Light-Use Efficiency of Common Ice Plant (*Mesembryanthemum crystallinum* L.) under Controlled-Environment Conditions

Kornlawat Tantivit^{1*}, Voravit Siripholvat², and Nopparat Tatmala³

^{1,2}Faculty of Innovative Agriculture and Management, Panyapiwat Institute of Management, Nonthaburi, Thailand

³Department of Crop Production, Faculty of Agricultural Technology, Rajamangala University of Technology Thanyaburi, Pathumthani, Thailand

E-mail: kornlawattan@pim.ac.th*, voravitsir@pim.ac.th, nopparat_t@rmutt.ac.th

Received: January 28, 2026 / Revised: March 20, 2026 / Accepted: March 31, 2026

Abstract— This study evaluated the effects of different photoperiod regimes: Long-day (16 h light/8 h dark), short-day (8 h light/16 h dark), and intermittent photoperiods, including three intermittent regimes (8/8 h, 4/4 h, and 10 min/10 min) on growth performance and biomass productivity of *Mesembryanthemum crystallinum* L. under controlled greenhouse conditions. Plants grown under intermittent photoperiod regimes exhibited significantly greater fresh weight, dry weight, and plant height than those grown under continuous short-day or long-day conditions ($p < 0.05$). These results indicate that temporal redistribution of light exposure can enhance growth efficiency, potentially by improving the coordination between photosynthetic activity and endogenous circadian regulation. Moreover, intermittent photoperiod suggests potential to improve energy-use efficiency without compromising biomass production. The findings demonstrate that intermittent photoperiod management is a promising strategy for the sustainable cultivation of *M. crystallinum* in smart greenhouses and controlled-environment agriculture systems.

Index Terms—Common Ice Plant, *M. Crystallinum*, PFAL, Intermittent Photoperiod, Circadian Clock, Photoreceptors

I. INTRODUCTION

The common ice plant (*M. crystallinum* L.), a member of the family *Aizoaceae* is native to eastern and southern Africa and has subsequently been introduced to coastal and arid regions worldwide. This species exhibits distinctive physiological traits, including high salinity tolerance and drought resistance, which are primarily attributed to specialized epidermal bladder cells that function in water and ion

storage. These characteristics enable *M. crystallinum* to maintain growth under environmental conditions that are generally unsuitable for conventional leafy vegetables [1].

Owing to these distinctive physiological and morphological traits, *M. crystallinum* has been increasingly recognized as a commercially valuable leafy vegetable and is currently cultivated and consumed in several countries, including the United States, Germany, the Netherlands, Australia, and New Zealand [2]. Notably, this species exhibits a high degree of salt tolerance, enabling successful growth and productivity in saline soils [2], [3]. It is classified as a functional vegetable because of its high nutritional value and abundance of health-promoting bioactive compounds, such as proline, inositol, vitamin K, β -carotene, and various antioxidants. In addition, pharmacological studies have reported diuretic activity [4], ultraviolet-protective properties [5], and antimicrobial effects [6], further highlighting its potential as a high-value crop [7].

Previous research has demonstrated that *M. crystallinum* can be successfully cultivated in Thailand, with Phitsanulok Province identified as the most suitable growing area when appropriate protection from rainfall and fog is provided. Plants cultivated under Thai conditions exhibited a calcium concentration approximately threefold higher and a sodium concentration forty-three fold lower than that reported for plants grown in China, while no significant differences were observed in vitamin A and sugar contents. These findings highlight the potential of *M. crystallinum* as a high-value edible crop for agricultural development in Thailand [8]. Although *M. crystallinum* can be successfully cultivated under Thai conditions, several limitations remain, particularly in production areas that are often far from major distribution and consumption centers,

thereby affecting postharvest quality and market accessibility. As a result, harvested products undergo prolonged transportation and handling, which can accelerate postharvest quality deterioration. As *M. crystallinum* is classified as a fresh-cut vegetable, it is highly perishable and sensitive to postharvest stresses. Therefore, the use of appropriate packaging systems, combined with low-temperature management throughout postharvest processing and distribution, is essential [9]. Maintaining an uninterrupted cold chain is critical for preserving product quality, extending shelf life, and minimizing physiological degradation, including water loss, tissue softening, and nutrient deterioration. Failure to adequately control temperature during postharvest handling may result in rapid quality loss, thereby reducing marketability and economic value. Consequently, the development and optimization of postharvest handling strategies, particularly those focusing on packaging and cold chain management, are crucial for the successful commercialization of ice plant as a high-value fresh vegetable in Thailand.

Furthermore, there is a lack of systematic studies on environmental control strategies that influence growth performance and yield. Among these factors, photoperiod is a key environmental signal governing a wide range of physiological processes and developmental events in plants. Photoperiodic regulation has been reported to extend beyond floral induction, encompassing processes such as hypocotyl elongation, tuber formation in potato, seasonal growth cessation in perennial species, and stomatal regulation [10]. These responses are mediated through the coordinated action of central regulatory genes, including FT-like, CO-like, and PRR gene families, which integrate light signals with endogenous circadian rhythms. Modulation of these genetic networks enables plants to adapt to diverse climatic conditions and seasonal cues, underscoring their importance in crop improvement and future agricultural production systems.

Beyond its role in developmental regulation, photoperiod also directly influences plant stress responses to both abiotic and biotic stressors through interactions between the circadian clock and photoreceptors [11]. Key components such as GIGANTEA (GI) and redox signaling pathways regulate Reactive Oxygen Species (ROS) homeostasis [12], [13]. Sudden alterations in photoperiod can induce photoperiod stress, a response that shares similarities with pathogen-induced defense mechanisms and may ultimately lead to programmed cell death. A deeper understanding of these mechanisms is therefore essential for developing stress-tolerant cultivars and optimizing light management strategies in CEA. Previous studies have shown that extended

photoperiods under Long-Day (LD) conditions generally enhance vegetative growth, particularly by increasing dry matter accumulation through leaf expansion and a greater photosynthetic surface area, although responses are species-specific [14]. However, extending the Photoperiod in Plant Factory with Artificial Lighting (PFAL) systems inevitably increases energy consumption. An alternative approach to achieving comparable photoperiodic responses with lower energy input is to use intermittent photoperiods, in which short light periods are alternated with dark intervals. Light–dark cycles, including transitional phases such as dusk and dawn, serve as critical cues for plant photoreceptors and the circadian clock, enabling the measurement of photoperiodic signals. Consequently, photoperiod perception depends not only on prolonged illumination but also on the duration and continuity of darkness as well as specific light signals occurring during the night [15]. In carnation, the insertion of brief light periods during the night has been shown to accelerate flowering to an extent comparable to that observed under true long-day conditions, indicating that photoperiodic perception is strongly influenced by disruption of the dark period [16]. More recently, studies on basil species have shown that intermittent nighttime lighting can increase plant dry biomass even when the total daily photoperiod is shorter, suggesting that intermittent light regimes can significantly enhance growth and photosynthetic efficiency in greenhouse and controlled-environment systems [17].

Despite its agronomic and nutritional potential, production-oriented studies on *M. crystallinum* remain limited, particularly under tropical conditions. In Thailand, information on environmental control strategies affecting growth and yield is scarce. In particular, the influence of the temporal structure of photoperiod, independent of total daily light duration, has not been systematically evaluated. Therefore, this study aimed to compare the effects of intermittent photoperiod regimes on growth performance and yield of *M. crystallinum* under controlled-environment conditions.

In this context, the present study aims to investigate the effects of intermittent photoperiod regimes on the growth and yield of *M. crystallinum*. The findings are expected to provide fundamental knowledge for production planning, the selection of appropriate light-management strategies, and the development of commercially viable cultivation practices tailored to Thailand's climatic conditions and resource constraints. Ultimately, this research seeks to support the introduction of ice plant as a novel alternative economic crop and to contribute to food security and the sustainable development of agricultural systems.

II. LITERATURE REVIEW

Photoperiodic Control of Plant Growth and Biomass Accumulation

The life cycle and growth of *M. crystallinum* can be divided into several key stages: seed germination, juvenile phase, adult phase, flowering, and seed development. Seeds typically germinate under moderate temperatures and slightly saline conditions. After transplanting, the plant enters the juvenile stage, during which it produces approximately one pair of leaves per week until about seven leaf pairs are formed. It then transitions to the adult stage, characterized by the development of secondary shoots and a significant physiological shift from C3 to CAM photosynthesis, especially under environmental stresses such as salinity or drought. During this stage, specialized bladder cells for salt storage become prominent. Flowering generally occurs from late spring to early summer and can be accelerated by salt stress, followed by capsule formation and seed maturation, completing its annual life cycle. Photoperiod is a fundamental environmental cue that regulates plant growth, development, and biomass accumulation by affecting photosynthesis, carbon allocation, and morphogenesis. In Controlled-Environment Agriculture (CEA), photoperiod manipulation has been widely employed to enhance vegetative growth and stabilize yield by extending the duration of daily photosynthetic activity. Long-day photoperiods generally promote vegetative biomass accumulation by stimulating leaf expansion and carbon assimilation; however, growth responses vary considerably among plant species and physiological types [14]. Notably, prolonged illumination does not necessarily increase productivity. Excessive or inadequately structured light exposure may disrupt metabolic coordination, induce photoperiod-related stress, and reduce growth efficiency, particularly under artificial lighting conditions [18], [19]. Therefore, optimizing photoperiod architecture, rather than merely extending day length, has become a key objective in PFAL systems [20]. Accumulating evidence indicates that photoperiodic regulation of biomass accumulation is closely linked to the plant circadian clock, which governs the temporal coordination of photosynthesis, carbohydrate metabolism, and growth-related gene expression [21], [22]. When photoperiodic signals are properly synchronized with endogenous circadian rhythms, plants exhibit enhanced carbon-use efficiency and improved biomass partitioning [23]. In contrast, circadian misalignment resulting from inappropriate or continuous light regimes can impair photosynthetic performance and suppress growth, even under conditions of sufficient light availability [18]. Consequently, alternative photoperiod strategies, including intermittent or non-continuous light

regimes, have received increasing attention as approaches to maintain circadian entrainment while reducing energy input in CEA systems. These lighting strategies have been shown to modulate daily patterns of carbon assimilation and biomass accumulation without proportional increases in total light duration, underscoring the importance of temporal light distribution in addition to overall light quantity [24], [25]. In physiologically plastic species such as *M. crystallinum*, photoperiodic control of biomass accumulation is particularly relevant. As a facultative Crassulacean Acid Metabolism (CAM) plant, ice plant exhibits strong circadian regulation of carbon metabolism and growth, enabling flexible physiological adjustments in response to environmental cues [26]. Modified photoperiod regimes may therefore influence biomass production by reprogramming circadian-controlled pathways associated with photosynthetic activity, carbon allocation, and water-use efficiency. Elucidating these interactions is essential for optimizing yield formation and improving energy-use efficiency in the controlled-environment cultivation of ice plant.

A. Circadian Clock, Photoreceptors, and Temporal Light–Dark Signalling

Plant responses to photoperiod are mediated by the integration of external light signals with endogenous circadian rhythms. Photoreceptors, including phytochromes and cryptochromes, perceive changes in light conditions and relay this information to the circadian clock, which coordinates rhythmic gene expression and metabolic activity across the diel cycle. Core clock components, such as TIMING OF CAB EXPRESSION 1 (TOC1), CIRCADIAN CLOCK ASSOCIATED 1 (CCA1), LATE ELONGATED HYPOCOTYL (LHY), PSEUDO-RESPONSE REGULATORS (PRRs), and GIGANTEA (GI), function collectively to regulate photoperiodic outputs related to growth, carbon metabolism, and stress responses [10].

Photoperiod perception is determined not only by the total duration of light exposure but also by the timing and continuity of darkness. Light–dark transitions provide critical entrainment signals that enable plants to accurately measure day length and synchronize physiological processes with environmental cycles [15]. This temporal dimension of photoperiod signaling is particularly relevant under artificial lighting, where light regimes can be precisely structured and repeatedly interrupted.

B. Intermittent Photoperiods and Circadian Entrainment in CEA

Intermittent photoperiods, characterized by repeated alternation of light and dark periods within a 24-h cycle, have been proposed as an alternative photoperiod management strategy in controlled-

environment systems. Classical studies have demonstrated that brief light interruptions at night can effectively alter photoperiodic perception by disrupting the scotoperiod, thereby inducing responses comparable to those observed under continuous long-day conditions [16].

More recent studies in greenhouse and indoor production systems have shown that intermittent photoperiod can enhance vegetative growth and biomass accumulation while reducing total daily light input. For example, increased dry biomass production has been reported in basil exposed to intermittent nighttime lighting, suggesting improved synchronization between light-driven photosynthesis and circadian-regulated metabolic processes [17]. These findings indicate that intermittent photoperiods may enhance growth efficiency by maintaining circadian entrainment while alleviating metabolic and photochemical stress associated with prolonged illumination.

C. Photoperiod Stress, Redox Regulation, and Energy Use Efficiency

Photoperiod structure also influences plant stress responses through interactions between the circadian clock and redox signaling pathways. Abrupt changes or prolonged extension of photoperiod can induce photoperiod stress, which is associated with oxidative imbalance and activation of defense-related pathways [11]. Circadian regulators such as GI and PRR proteins have been implicated in coordinating Reactive Oxygen Species (ROS) homeostasis with daily metabolic rhythms [12].

From an applied perspective, photoperiod management in PFAL systems must balance growth promotion with energy efficiency. Continuous long-day lighting substantially increases electricity consumption, limiting economic and environmental sustainability. By redistributing light exposure without increasing the total Daily Light Integral (DLI), intermittent photoperiods can improve Light Use Efficiency (LUE) and reduce energy costs per unit biomass produced. However, empirical evaluations of these benefits remain limited, particularly for non-model crops.

D. Photoperiodic Regulation of CAM Metabolism

M. crystallinum is a facultative CAM halophyte exhibiting high tolerance to salinity and drought. Its photosynthetic metabolism can shift between C_3 and CAM modes depending on environmental conditions, with CAM induction involving a coordinated sequence of four diel phases. During Phase I, stomata open at night, allowing CO_2 uptake and fixation via Phosphoenolpyruvate Carboxylase (PEPC) into malate, which is stored in vacuoles. Phase II represents a transitional period at dawn, when both PEPC and ribulose-1, 5-bisphosphate

carboxylase/oxygenase (RubisCO) may contribute to carbon fixation. In Phase III, stomata close during the day and malate is decarboxylated, providing CO_2 for refixation by Rubisco. Phase IV occurs toward the end of the light period, when stomata may reopen and direct CO_2 assimilation resumes.

The timing and duration of these CAM phases are tightly linked to light–dark cycles and circadian regulation. Alterations in photoperiod structure, particularly through intermittent light–dark transitions, are therefore expected to influence CAM phase dynamics, carbohydrate partitioning, and ion accumulation in *M. crystallinum*. Despite extensive research on salinity-induced CAM expression, little is known about how the architecture of photoperiods modulates CAM-related growth responses under PFAL conditions.

E. Knowledge Gaps and Research Rationale

Although previous studies have established the importance of photoperiod and circadian regulation in plant growth and stress adaptation, the effects of intermittent photoperiods on facultative CAM crops remain poorly understood. In particular, it is unclear how the temporal distribution of light and dark periods, independent of total daily light duration, influences biomass production, morphological development, and physiological balance in *M. crystallinum*.

Therefore, a systematic comparison of short-day, long-day, and intermittent photoperiod regimes under PFAL conditions is required to elucidate growth responses and to identify lighting strategies that optimise both productivity and energy-use efficiency. Addressing this knowledge gap will advance fundamental understanding of photoperiod–circadian–CAM interactions and support the development of sustainable production systems for high-value halophytic vegetables.

III. MATERIALS AND METHODS

A. Plant Material and Growing Conditions

Seeds of *M. crystallinum* used in this experiment were obtained from a commercial grower cultivating ice plant in the production area of Phitsanulok Province, Thailand. Growth conditions were maintained at an air temperature of $25 \pm 2^\circ C$, relative humidity of 65–75%. Juvenile phase seedlings (28 days after germination) with comparable size and developmental stage were transplanted into individual containers and transferred to a controlled-environment cultivation facility representative of a plant factory with PFAL conditions. Plants were maintained under controlled air temperature ($25 \pm 2^\circ C$) and relative humidity (65–75%) throughout the experimental period. Environmental parameters were continuously monitored to ensure consistency

among treatments. Irrigation and nutrient supply were applied uniformly across treatments to minimise confounding effects, thereby isolating the photoperiod regime as the primary experimental variable affecting plant growth and physiological responses.

B. Experimental Design

The experiment followed a Completely Randomized Design (CRD) with 5 photoperiod treatments: long-day (Trt.1; 16 h light / 8 h dark), short-day (Trt.2; 8 h light / 16 h dark), and an intermittent photoperiod (Trt.3-5) consisting of alternating light and dark cycles within a 24-h period. The intermittent photoperiod treatment was divided into three regimes: Trt.3, with alternating light and dark periods every 8 h; Trt.4, with alternating light and dark periods every 4 h; and Trt.5, with alternating light and dark periods at 10-min intervals. Light was supplied exclusively by Light-Emitting Diode (LED) fixtures. Photosynthetic Photon Flux Density (PPFD) at the canopy level was maintained uniformly across treatments, and spectral composition was kept constant to minimize variation in light quality effects. Plant positions were periodically randomized to reduce potential positional bias within the growth environment.

C. Growth, Morphological, and Biomass Measurements

After 6 weeks of photoperiod treatment, plants were harvested for growth and biomass assessments. Morphological parameters, including plant height, number of leaves, and canopy width, were measured as indicators of vegetative growth and canopy development. Fresh weight of aboveground biomass was recorded immediately after harvest. Samples were subsequently oven-dried at 70 °C until constant weight was achieved to determine dry matter accumulation. These parameters were selected to capture photoperiod-induced differences in growth performance and yield under PFAL conditions.

D. Light Use Efficiency (LUE) Calculation

Light Use Efficiency (LUE) was calculated to assess how efficiently plants converted incident photosynthetically active radiation into biomass under different photoperiod regimes. LUE was expressed as dry biomass produced per unit of cumulative photosynthetic photon flux and was calculated using the following equation:

$$\text{LUE} = \frac{\text{Total dry biomass (g plant}^{-1}\text{)}}{\text{cumulative light integral (mol photons m}^{-2}\text{)}}$$

The cumulative light integral was calculated from the measured canopy-level PPFD and the total duration of light exposure for each treatment over the experimental period. By maintaining equivalent total

daily light duration between the IP and LD treatments, differences in LUE could be attributed primarily to the temporal structure of photoperiod rather than to total light input. This approach enables direct evaluation of the potential of intermittent photoperiod regimes to improve energy-use efficiency in PFAL systems.

E. Nutritional Composition Analysis

The nutritional composition of the samples was analyzed in accordance with the Thai Recommended Daily Intake (Thai-RDI No. 445) guidelines. Laboratory analyses were conducted at the Foundation for Industrial Development, National Food Institute (NFI), Bang Phlat, and Bangkok, Thailand. Standard analytical methods were employed to determine proximate composition, including moisture, ash, protein, fat, carbohydrate, and energy content, as well as selected minerals relevant to Thai-RDI criteria. The obtained values were subsequently calculated and expressed per serving size and compared with Thai-RDI standards to determine the percentage contribution of each nutrient to the daily recommended intake.

F. Statistical Analysis

All experimental data were subjected to ANOVA to assess the effects of different photoperiod regimes on growth, biomass, and light-use efficiency. Post hoc mean separation was applied when treatment effects were significant. The level of significance was defined as $P < 0.05$. Results are expressed as mean values with measures of variability. Statistical computations were performed using Microsoft Excel.

IV. RESULTS

A. Effects of Photoperiod on Morphological Growth

Photoperiod treatments had significant effects on several morphological traits ($p < 0.05$). Plant height (PH), Canopy Width (CW), Leaf Number (NL), Leaf Width (LW), and Stem Diameter (SD) were influenced by the light regime across the 5 photoperiod treatments (Table I).

Plant height increased continuously throughout the experimental period in all treatments; however, elongation growth differed significantly among treatments. Intermittent photoperiod of 4 h light/4 h dark (Trt.4) and 10 min light/10 min dark (Trt.5) produced the greatest final plant height at week 8, with mean values of 19.84 and 16.72 cm, respectively, whereas the long photoperiod (Trt.1; 16/8 h light/dark) showed 15.28 cm height and the short photoperiod (Trt.2; 8/16 h light/dark) showed the lowest height with 11.71 cm. These results indicate that frequent light–dark alternation promotes stem elongation more effectively than continuous exposure to long light.

Canopy Width (CW) differed significantly among treatments ($p < 0.05$). Trt.5 showed the largest canopy

width with 17.18 cm. during the middle growth stages (weeks 5-6), while Trt.1 tended to have the highest CW at the final sampling time with 27.79 cm. Variation in CW among treatments reflects differences in leaf expansion and spatial leaf arrangement, both influenced by photoperiod (Fig. 1).



Fig. 1. *M. crystallinum* plants at 6 weeks of age grown under continuous and intermittent photoperiod regimes. Images show a top view (upper) and an oblique view (Lower). Treatments are arranged from Trt.1 to 5 from left to right.

Leaf Number (NL) responded strongly to photoperiod. The highest value was obtained under Trt.5 (10 min/10 min), reaching 106.27 leaves plant⁻¹ at week 6, followed by Trt.3 (74.12 leaves) and Trt.2. (72.93 leaves). Differences in NL among treatments were highly significant ($p < 0.01$). Increased leaf initiation under short, highly intermittent photoperiods suggests enhanced shoot branching and a greater cumulative photosynthetic area. In contrast, Leaf Width (LW) tended to decrease toward the end of the experiment, particularly under Trt.5, where small leaves were associated with very high leaf numbers. Stem Diameter (SD) increased steadily throughout the experiment in all treatments. Trt.4 (4.65 cm) and Trt.5 (4.2 cm) had the highest SD values at week 8, while long photoperiod (Trt.1) resulted in moderate stem thickness with 3.95 cm. Significant differences were detected mainly during the later growth stages ($p < 0.05$). When averaged across six sampling dates (weeks 3-8), Trt.4 obtained the highest mean PH (9.62 cm), whereas Trt.5 yielded the highest CW (15.58 cm) and NL (100.60 leaves plant⁻¹).

TABLE I
EFFECTS OF PHOTOPERIOD REGIMES ON MORPHOLOGICAL GROWTH TRAITS OF *M. CRYSTALLINUM* L. BASED ON 6-WEEK MEAN VALUES

Treatment	PH	CW	NL	LW ^{ns}	SD ^{ns}
Trt.1	7.80 ^{bc}	12.92 ^c	20.52 ^c	2.95	4.90
Trt.2	6.53 ^c	13.55 ^{bc}	72.93 ^b	2.54	4.46
Trt.3	6.35 ^c	12.58 ^c	74.12 ^b	2.94	4.56
Trt.4	9.62 ^a	14.78 ^{ab}	30.33 ^c	2.86	4.70
Trt.5	8.74 ^{ab}	15.58 ^a	106.27 ^a	2.68	4.64

PH, plant height (cm); CW, canopy width (cm); NL, number of leaves per plant; LW, leaf width (cm); SD, stem diameter (mm). Statistical differences among photoperiod regimes were determined using one-way ANOVA followed by Tukey's HSD test ($P < 0.05$). ns = not significant

Overall, intermittent photoperiods, particularly very short alternating cycles, promoted vegetative growth in ways that differed from continuous photoperiods. Short and frequent light-dark cycles enhanced leaf production, whereas 4/4 h cycles favored stem elongation and structural biomass accumulation.

B. Dry Weight/Fresh Weight Ratio and Chlorophyll Content

Dry weight/fresh weight ratios ranged from 0.0261 to 0.0465 across treatments. The highest ratio was found under Trt.3, whereas the lowest ratios were observed under Trt.4 (Table II). However, statistical analysis indicated no significant differences among treatments ($p > 0.05$). This suggests that plants maintained relatively stable biomass partitioning regardless of photoperiod regime. Total chlorophyll content ranged from 2.14 to 2.17 mg g⁻¹ FW (data not shown), and there were no significant differences among treatments ($p > 0.05$). Neither long nor short continuous photoperiods nor intermittent regimes (8/8 h, 4/4 h, 10 min/10 min) resulted in detectable increases or decreases in chlorophyll concentration. The ability to maintain stable chlorophyll levels across treatments indicates effective physiological acclimation of ice plant to fluctuating light regimes.

TABLE II
COMPARISON OF DRY WEIGHT TO FRESH WEIGHT RATIO (DW/FW) OF *M. CRYSTALLINUM* L. UNDER DIFFERENT PHOTOPERIOD REGIMES

Treatments	Dry Weight / Fresh Weight (g/g) ^{ns}
Trt.1	0.0277
Trt.2	0.0355
Trt.3	0.0465
Trt.4	0.0264
Trt.5	0.0317

A. Effects of Photoperiod on Nutritional Composition

The nutritional composition of the plants was analyzed and compared among the 5 treatments, as shown in Table III. Energy content varied among treatments. Short photoperiod (Trt.2) produced the highest energy value (61 kcal 100 g⁻¹), followed by intermittent 8/8 h (Trt.3; 48 kcal 100 g⁻¹). Intermittent 4/4 h (Trt.4) and 10 min/10 min (Trt.5) showed energy values below 30 kcal 100 g⁻¹.

Variations in energy content were associated mainly with total carbohydrate concentration, which was highest in Trt.2 (10.8 g 100 g⁻¹ FW) and Trt.3 (7.6 g 100 g⁻¹ FW).

Total fat and saturated fat contents were very low (<0.6 g 100 g⁻¹ FW) in all treatments. The highest

values occurred in Trt.2 and Trt.3, whereas Trt.1 and Trt.4 contained less than $0.2 \text{ g } 100 \text{ g}^{-1} \text{ FW}$. These results confirm that ice plant is a low-fat leafy vegetable.

Protein content varied among treatments, with Trt.3 exhibiting the highest protein level ($3.5 \text{ g } 100 \text{ g}^{-1} \text{ FW}$), followed by Trt.2 ($3.2 \text{ g } 100 \text{ g}^{-1} \text{ FW}$) and Trt.5 ($2.8 \text{ g } 100 \text{ g}^{-1} \text{ FW}$). Lower protein levels were observed in Trt.1 and Trt.4. Differences in protein accumulation reflect variation in nitrogen metabolism and synthesis of structural and enzymatic proteins under different photoperiods.

Sodium (Na) and potassium (K) contents differed clearly among treatments. Trt.3 had the highest Na ($322 \text{ mg } 100 \text{ g}^{-1} \text{ FW}$) and K ($1257 \text{ mg } 100 \text{ g}^{-1} \text{ FW}$), followed by Trt.2. Accumulation of these ions is consistent with osmotic adjustment and the halophytic nature of *M. crystallinum*. Moisture content was highest in Trt.4 (90.4%), indicating greater water storage capacity under 4/4 h intermittent photoperiod.

TABLE III
NUTRITIONAL RESPONSES OF *M. CRYSTALLINUM* L. TO
DIFFERENT PHOTOPERIOD REGIMES

Components	Trt.1	Trt.2	Trt.3	Trt.4	Trt.5
Energy (kcal 100 g^{-1})	35	61	48	28	29
Total Fat (g $100 \text{ g}^{-1} \text{ FW}$)	0.2	0.6	0.4	0.1	0.2
Saturated Fat (g $100 \text{ g}^{-1} \text{ FW}$)	0.1	0.2	0.3	0.02	0.1
Carbohydrate (g $100 \text{ g}^{-1} \text{ FW}$)	5.7	10.8	7.6	4.5	4.0
Protein (g $100 \text{ g}^{-1} \text{ FW}$)	2.6	3.2	3.5	2.3	2.8
Sodium (mg $100 \text{ g}^{-1} \text{ FW}$)	193	308	322	206	251
Potassium (mg $100 \text{ g}^{-1} \text{ FW}$)	843	1124	1257	865	975
Moisture (%)	88.75	81.84	83.83	90.40	89.81

Intermittent photoperiods enhanced vegetative growth, particularly leaf number and stem elongation, whereas short photoperiods favored higher carbohydrate and energy accumulation.

C. Effects of Photoperiod on Light-Use Efficiency (LUE)

LUE, expressed as dry biomass produced per unit of incident photosynthetically active radiation, varied significantly among photoperiod treatments despite identical Daily Light Integrals (DLI) across all treatments. This indicates that temporal light-dark distribution, rather than total photon input alone, plays a critical role in determining biomass conversion efficiency. Plants grown under intermittent photoperiod regimes exhibited higher LUE values than those cultivated under continuous long-day or short-day conditions. Among the intermittent treatments, the 4 h light/4 h dark regime demonstrated the highest LUE, followed by the 10 min light/10 min dark treatment. In contrast, the conventional long-day photoperiod showed comparatively lower efficiency. The short-day

treatment resulted in the lowest LUE, consistent with its reduced biomass accumulation. Notably, the enhancement of LUE under intermittent lighting was not strictly proportional to total biomass production. While the 10 min/10 min regime promoted greater leaf biomass accumulation, the 4/4 h treatment achieved superior efficiency in converting supplied photons into dry matter. This suggests that longer uninterrupted dark intervals within intermittent cycles may improve metabolic coordination and carbon utilization efficiency without increasing total light input. Overall, these results demonstrate that photoperiod manipulation can significantly alter light-use efficiency independently of DLI, highlighting intermittent lighting as a promising strategy for improving energy efficiency in controlled-environment plant production systems.

V. DISCUSSION

A. Photoperiod Effects on Plant Height

The photoperiod regime significantly affected plant height in *M. crystallinum*. The 4 h light/4 h dark intermittent treatment produced the tallest plants, whereas the 8 h light/16 h dark regime resulted in the shortest plants. These results indicate that extended daily light availability, when temporally structured by dark intervals, can favor stem elongation. Previous studies have shown that photoperiodic regulation of elongation growth is determined not solely by total light duration but also by the continuity and timing of the dark period, which strongly influences circadian entrainment and downstream growth responses [14], [15]. Periodic dark phases may permit partial recovery of the photosynthetic apparatus and rebalancing of carbon metabolism, thereby supporting sustained cell expansion and elongation growth under artificial lighting conditions. Such recovery processes have been associated with improved coordination between photosynthesis, respiration, and circadian-regulated metabolic fluxes, particularly under non-continuous light regimes [23], [26]. Moreover, circadian clock components, including GIGANTEA and PSEUDO-RESPONSE REGULATORS, have been implicated in the temporal regulation of growth-related processes, including hypocotyl and stem elongation, through interactions with light signaling pathways [10], [27]. Although hormonal parameters were not measured in the present study, previous research has demonstrated that photoperiod and circadian regulation can modulate the biosynthesis and signaling of growth regulators such as auxins and gibberellins, which play key roles in cell elongation [28], [29]. Therefore, the potential involvement of hormonal control in mediating elongation responses under intermittent photoperiods remains plausible but requires further experimental verification.

B. Leaf Number and Canopy Development

Leaf number showed strong responsiveness to the structure of the photoperiod. The 10 min light/10 min dark regime generated the highest leaf number, whereas the short-day treatment produced the fewest leaves. These results suggest that rapid alternation of light and dark periods may enhance leaf initiation and meristematic activity, thereby promoting increased shoot branching and canopy development. Previous studies have demonstrated that photoperiod and circadian regulation influence the activity of the shoot apical meristem and the rate of leaf primordia initiation through coordinated control of carbon availability and clock-regulated growth signals [23], [27]. Frequent light–dark transitions may also improve synchronization between photosynthetic carbon supply and circadian-controlled metabolic demand, supporting continuous organ initiation even under relatively short effective light periods [26]. In CEA systems, similar responses have been observed in leafy vegetables, where intermittent or fragmented lighting regimes increased leaf number and canopy density without proportional increases in individual leaf size [17].

Although increased leaf number likely contributed to higher cumulative photosynthetic surface area and biomass production, individual leaf area was not quantified in the present study. Consequently, potential trade-offs between leaf number and leaf size, which are commonly associated with shifts in carbon partitioning and developmental regulation, cannot be conclusively assessed [29]. However, no statistically significant differences were observed in the dry weight-to-fresh weight ratio or total leaf chlorophyll content among the different photoperiod treatments, which is consistent with previous findings [30].

C. Carbohydrate Accumulation and Cam Metabolism

Carbohydrate content was greatest under The short-day and 8/8 h intermittent regimes, suggesting that relatively longer or well-defined dark periods favor carbon storage in *M. crystallinum*. As a facultative CAM species, *M. crystallinum* exhibits pronounced diel shifts across CAM phases I–IV, in which nocturnal CO₂ fixation and daytime decarboxylation are tightly coordinated with light–dark cycles and circadian regulation [31], [32]. Intermittent photoperiods may alter the timing and duration of CAM phase transitions, particularly Phase II (early-morning transition) and Phase IV (late-afternoon transition), during which both PEPC- and Rubisco-mediated carbon fixation can occur. Modulation of these transitional phases has been shown to influence the balance between malate decarboxylation, carbohydrate synthesis, and carbon export, thereby affecting soluble carbohydrate accumulation and biomass composition [25], [33].

Furthermore, circadian control of CAM-related gene expression, including enzymes involved in malate metabolism and carbohydrate turnover, suggests that temporal restructuring of light–dark cycles could shift carbon partitioning independently of total daily light input [34], [35]. Although these interpretations are consistent with established CAM physiology, they remain inferential in the present study because gas exchange, diel malate fluctuations, and titratable acidity were not directly measured. Direct physiological measurements will therefore be required to confirm the mechanistic basis of carbohydrate accumulation under intermittent photoperiod regimes.

D. Sodium and Potassium Accumulation

Sodium and potassium accumulation peaked under the 8/8 h intermittent regime. Given the halophytic nature of *M. crystallinum*, elevated Na⁺ and K⁺ levels are likely associated with osmotic adjustment and the development of succulence, processes fundamental to salt tolerance in this species [1], [3]. In ice plant, Na⁺ and K⁺ are predominantly sequestered in epidermal bladder cells and vacuoles, contributing to tissue succulence and maintenance of cellular water balance under saline or osmotic stress conditions [3], [36]. Intermittent photoperiods may influence ion accumulation by affecting transpiration dynamics, stomatal behaviour, and circadian-regulated membrane transport activity. Light–dark transitions are known to modulate ion fluxes across plasma and tonoplast membranes, partly through circadian control of transporter expression and activity, including Na⁺/H⁺ antiporters and K⁺ transport systems [37]. Moreover, synchronization between photoperiod structure and circadian regulation may enhance the temporal coordination of ion uptake, long-distance transport, and vacuolar sequestration, thereby promoting ion accumulation without inducing ionic toxicity [13], [23].

Nevertheless, the present results are based on bulk ion measurements, and the underlying regulatory mechanisms remain unresolved. Further studies integrating transporter gene expression, diel ion flux analysis, and compartment-specific ion localization will be required to elucidate how intermittent photoperiods modulate Na⁺ and K⁺ homeostasis in *M. crystallinum*.

E. Implications of Photoperiod Manipulation on Light-Use Efficiency

Light-use Efficiency (LUE) differed significantly among photoperiod treatments despite identical daily light integrals, emphasizing the importance of temporal light–dark distribution in CEA. Intermittent lighting regimes enhanced LUE relative to continuous long-day and short-day conditions, indicating that photoperiod structure influences photon conversion

efficiency beyond total light input. The highest LUE was observed under The 4 h light/4 h dark regime, whereas the 10 min light/10 min dark treatment favored biomass accumulation but showed slightly lower efficiency, suggesting a trade-off between yield maximisation and light-use efficiency. From an engineering perspective, these results highlight intermittent photoperiod manipulation as a practical strategy to improve energy efficiency in plant factories with artificial lighting and other controlled-environment production systems.

E. Overall Integration and Horticultural Implications

Taken together, intermittent photoperiods enhanced several growth parameters relative to continuous short- and long-day lighting, highlighting the role of photoperiod architecture as a tunable control variable in PFAL [14], [38]. The 10 min light/10 min dark regime preferentially increased leaf proliferation. In contrast, the 4 h light/4 h dark regime promoted greater plant height, demonstrating that temporal structuring of light–dark cycles can be aligned with distinct production targets such as canopy density or vertical growth [17], [39]. These results indicate that lighting optimization in PFAL should extend beyond total daily light duration to include the temporal distribution of irradiance [17], [40]. The relative stability of chlorophyll concentration and dry matter percentage across treatments suggests that intermittent photoperiods primarily influenced growth dynamics and biomass partitioning rather than fundamental tissue composition [14], [41].

From a PFAL perspective, this decoupling of morphological control from basic physiological stability is advantageous, as it enables architectural manipulation without compromising product quality attributes [39], [42]. Collectively, the findings support the use of intermittent photoperiod design as a practical strategy for optimizing crop performance and production flexibility in energy-constrained CEA systems [25], [39]. For CEA and PFAL, intermittent photoperiod is a promising management strategy that can maintain plant growth while reducing lighting duration and associated energy demand [25], [40]. Detailed assessments of energy consumption and economic efficiency will, however, be required to fully evaluate its practical advantages under commercial production conditions [38]. Overall, the present findings demonstrate that not only photoperiod length but also the temporal distribution of light–dark cycles influences growth and physiological performance in *M. crystallinum*, consistent with previous reports emphasizing the importance of photoperiod architecture in controlled environments [14], [39]. Future studies integrating CAM-related gas exchange measurements, circadian regulation, and energy-use analysis will further

clarify the mechanistic basis of plant responses to intermittent photoperiods and support evidence-based optimization of PFAL lighting strategies [25], [38].

VI. CONCLUSION

This study demonstrates that temporal structuring of light and dark periods influences growth, canopy development, and nutrient accumulation in *M. crystallinum* under controlled-environment conditions. Intermittent light regimes did not uniformly increase all traits; rather, different architectures of light–dark cycles favored distinct growth outcomes. The 4 h light/4 h dark regime promoted greater plant height, whereas the 10 min light/10 min dark regime enhanced leaf proliferation, indicating that short, frequent alternations particularly stimulate leaf initiation. Dry weight ratio and total chlorophyll content remained relatively stable among light regimes, suggesting that basic tissue composition and photosynthetic pigment levels were largely maintained despite substantial differences in photoperiod structure.

Carbohydrate content was highest under the short-day and 8/8 h intermittent regimes, implying that extended dark periods may facilitate carbon storage processes in this facultative CAM species. Ion accumulation also varied with photoperiod architecture, with the 8/8 h intermittent regime supporting elevated Na^+ and K^+ concentrations, consistent with osmotic adjustment and succulence development characteristic of halophytes. These patterns are compatible with shifts among CAM phases I–IV and associated carbohydrate partitioning, although direct physiological confirmation will require gas-exchange and acidity measurements.

Overall, the findings indicate that not only total day length but also the distribution of light and dark intervals modulates growth and physiological attributes in *M. crystallinum*. Intermittent photoperiod thus represents a promising management option for CEA and PFAL systems, with potential to maintain growth performance while modifying lighting schedules. However, a full assessment of energy-use efficiency, economic performance, and underlying regulatory mechanisms will require further integrative studies that combine CAM dynamics, circadian regulation, and analyses of light-use efficiency.

REFERENCES

- [1] P. Adams et al., "Growth and development of *Mesembryanthemum crystallinum* (Aizoaceae)," *New Phytol.*, vol. 138, no. 2, pp. 171–190, Feb. 1998.
- [2] S. Agaric, "Possibility of desalinization of saline soils by common ice plant (*Mesembryanthemum crystallinum* L.)," *Jpn. J. Trop. Agric.*, vol. 48, pp. 294–298, Dec. 2004.
- [3] S. Agaric et al., "Salt tolerance, salt accumulation, and ionic homeostasis in an epidermal bladder-cell-less mutant of the common ice plant *Mesembryanthemum crystallinum*," *J. Exp. Bot.*, vol. 58, no. 8, pp. 1957–1967, Jun. 2007.

- [4] I. Bouftira et al., "Effect of *Mesembryanthemum crystallinum* extract against DMH-induced colon carcinogenesis in experimental animals," *Int. J. Res. Pharm. Biomed. Sci.*, vol. 3, no. 3, pp. 1038-1043, Jun. 2012.
- [5] I. Bouftira, C. Abdelly, and S. Sfar, "Characterization of cosmetic cream with *Mesembryanthemum crystallinum* plant extract: Influence of formulation composition on physical stability and antioxidant activity," *Int. J. Cosmet. Sci.*, vol. 30, no. 6, pp. 443-452, Dec. 2008.
- [6] R. Ksouri et al., "Influence of biological, environmental and technical factors on phenolic content and antioxidant activities of Tunisian halophytes," *C. R. Biol.*, vol. 331, no. 11, pp. 865-873, Nov. 2008.
- [7] S. Agarie et al., "Potential of the common ice plant, *Mesembryanthemum crystallinum*, as a new high-functional food as evaluated by polyol accumulation," *Plant Prod. Sci.*, vol. 12, no. 1, pp. 37-46, Jan. 2009.
- [8] C. Burana, M. Congtrakultien, and N. Kamthornsiriwimol, "A novel study on the growth and development of *Mesembryanthemum crystallinum* (Aizoaceae) in Thailand," *Int. Sci. J. Eng. Technol. (ISJET)*, vol. 4, no. 1, Jan. 2020.
- [9] C. Burana et al., "Effect of packaging materials on the quality and shelf life of fresh-cut vegetables," *Int. Sci. J. Eng. Technol. (ISJET)*, vol. 9, no. 1, pp. 31-36, Jan. 2025.
- [10] M. Osnato et al., "Photoperiod control of plant growth: Flowering time genes beyond flowering," *Front. Plant Sci.*, vol. 12, p. 805635, Feb. 2021.
- [11] V. M. Röber, T. Schmülling, and A. Cortleven, "The photoperiod: Handling and causing stress in plants," *Front. Plant Sci.*, vol. 12, p. 781988, Jan. 2022.
- [12] J. Liu et al., "Novel insights into the factors influencing rhizosphere reactive oxygen species production and their role in polycyclic aromatic hydrocarbons transformation," *Soil Biol. Biochem.*, vol. 198, p. 109562, Nov. 2024.
- [13] L. Bao et al., "Reactive oxygen species-post-translational modifications-central carbon metabolism regulatory loop," *Front. Plant Sci.*, vol. 16, p. 1637328, Sep. 2025.
- [14] S. R. Adams and F. A. Langton, "Photoperiod and plant growth: A review," *J. Hortic. Sci. Biotechnol.*, vol. 80, pp. 2-10, Jan. 2005.
- [15] D. Vince-Prue, "The duration of light and photoperiodic responses," in *Photomorphogenesis in Plants*, R. E. Kendrick and G. H. M. Kronenberg, Eds. Berlin, Germany: Springer, 1986, pp. 269-305.
- [16] G. Harris and J. Griffin, "Flower initiation in the carnation in response to photoperiod," *Nature*, vol. 191, p. 614, Aug. 1961.
- [17] D. D. Avgoustaki, J. Li, and G. Xydis, "Basil plants grown under intermittent light stress," *Food Control*, vol. 118, p. 107389, Dec. 2020.
- [18] A. I. Velez-Ramirez et al., "Plants under continuous light," *Trends Plant Sci.*, vol. 16, no. 6, pp. 310-318, Jun. 2011.
- [19] H. Poorter et al., "The art of growing plants for experimental purposes," *Funct. Plant Biol.*, vol. 39, no. 11, pp. 821-838, Nov. 2012.
- [20] T. Kozai, G. Niu, and M. Takagaki, *Plant Factory: An Indoor Vertical Farming System for Efficient Quality Food Production*. London, U.K.: Academic Press, 2016.
- [21] S. L. Harmer, "The circadian system in higher plants," *Annu. Rev. Plant Biol.*, vol. 60, pp. 357-377, Jan. 2009.
- [22] K. Greenham and C. R. McClung, "Integrating circadian dynamics with physiological processes in plants," *Nat. Rev. Genet.*, vol. 16, pp. 598-610, Oct. 2015.
- [23] A. N. Dodd et al., "Plant circadian clocks increase photosynthesis," *Science*, vol. 309, pp. 630-633, Jul. 2005.
- [24] N. Ohtake et al., "Continuous irradiation with alternating red and blue light," *HortScience*, vol. 53, no. 12, pp. 1804-1809, Dec. 2018.
- [25] A. M. Borland et al., "Exploiting the potential of plants with CAM," *J. Exp. Bot.*, vol. 60, pp. 1829-1842, Apr. 2009.
- [26] A. Graf et al., "Circadian control of carbohydrate availability," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 107, no. 20, pp. 9458-9463, May 2010.
- [27] K. Nozue et al., "Rhythmic growth explained by coincidence," *Nature*, vol. 448, pp. 358-361, Jul. 2007.
- [28] E. M. Farré, "The regulation of plant growth by the circadian clock," *Plant Biol.*, vol. 14, pp. 401-410, May 2012.
- [29] Y. Kamiya, "Plant hormones: Versatile regulators," *Annu. Rev. Plant Biol.*, vol. 61, pp. 201-225, Jun. 2010.
- [30] J. W. Chen, K. Patloková, and R. Pokluda, "The effect of daily light integral," *Horticulturae*, vol. 10, no. 3, p. 266, Mar. 2024.
- [31] C. B. Osmond, "Crassulacean acid metabolism," *Annu. Rev. Plant Physiol.*, vol. 29, pp. 379-414, Jun. 1978.
- [32] J. C. Cushman and H. J. Bohnert, "Molecular genetics of CAM," *Plant Physiol.*, vol. 113, pp. 667-676, Mar. 1997.
- [33] K. Winter and J. A. M. Holtum, "Facultative CAM plants," *J. Exp. Bot.*, vol. 65, no. 13, pp. 3425-3441, Jul. 2014.
- [34] L. V. Dever et al., "Molecular regulation of CAM photosynthesis," *Plant J.*, vol. 83, pp. 361-374, Jul. 2015.
- [35] J. Hartwell et al., "Emerging model systems for CAM," *Curr. Opin. Plant Biol.*, vol. 31, pp. 100-108, Jun. 2016.
- [36] B. J. Barkla et al., "Ion transport in halophytes," *Plant Cell Environ.*, vol. 32, pp. 112-128, Feb. 2009.
- [37] R. A. Gaxiola, M. G. Palmgren, and K. Schumacher, "Plant proton pumps," *FEBS Lett.*, vol. 581, no. 12, pp. 2204-2214, 2007.
- [38] T. Kozai et al., "Smart plant factories," *Hortic. J.*, vol. 88, pp. 1-15, May 2019.
- [39] M. Olle and A. Viršile, "Effects of LED lighting," *Agric. Food Sci.*, vol. 22, no. 2, pp. 223-234, Jan. 2013.
- [40] C. A. Mitchell et al., "Light-emitting diodes in horticulture," *Hortic. Rev.*, vol. 43, pp. 1-87, Sep. 2015.
- [41] H. Dou et al., "Responses of leafy vegetables to dynamic light," *Sci. Hortic.*, vol. 235, pp. 273-280, Jun. 2018.
- [42] G. Pennisi et al., "Optimal light intensity for indoor cultivation," *Sci. Hortic.*, vol. 272, p. 109508, Oct. 2020.



Kornlawat Tantivit was born in Kathu, Phuket, Thailand, in 1989. He received the B.S. degree in Plant Science (Horticulture) from Maejo University, Chiang Mai, Thailand, in 2011. During his undergraduate studies, he completed an internship at the Horticultural Research Farm at the University of Kentucky in KY, USA. He obtained the M.Sc. degree in Agriculture from Kagawa University, Kagawa, Japan, in 2014, and the Ph.D. degree in Plant Cytogenetics from Ehime University, Matsuyama, Ehime, Japan, in 2017, with support from the Japanese Government (MEXT) Scholarship. Following completion of his doctoral studies, he served as a postdoctoral researcher in the Biochemistry Laboratory at Chulalongkorn University. From 2019 to 2022, he was employed as a Senior Officer and Specialist in Agricultural Innovation at Doikham Food Products Co., Ltd. Since 2023, he has been a lecturer and researcher at the Faculty of Innovative Agricultural Management, Panyapiwat Institute of Management. His research interests include plant genetics, plant physiology, and soilless culture. He is the first author of three

scientific articles in the field of plant cytogenetics. One of his publications in the journal *Cytologia* was selected for the 2017 Wada Memorial Award by the Japan Mendel Society.



Voravit Siripholvat is an Associate Professor and Vice Dean for Academic Affairs at the Faculty of Innovative Agricultural Management, Panyapiwat Institute of Management. He received his B.Sc. in Agriculture (1974) and M.Sc. in Genetics (1977) from Kasetsart University, Thailand, and earned his Doctor of Agriculture in Animal Genetics from Nagoya University, Japan, in 1988. He has extensive experience in higher education, research, and academic administration. At Kasetsart University, Kamphaeng Saen Campus, he served as Deputy Dean and Dean of the Faculty of Agriculture, and as Chair of the Thai Council of Deans of Agriculture, contributing to national policy directions in agricultural education and research. His research focuses on animal genetics and poultry breeding, particularly the Major Histocompatibility Complex (MHC) in chickens, bovine growth hormone polymorphisms, and reproductive biotechnology. After retiring from Kasetsart University, he expanded his work into plant science and sustainable agriculture, including rubber yield optimization and the development of clean agriculture in the Lao PDR.



Nopparat Tatmala is a lecturer and researcher in the Faculty of Agricultural Technology, Rajamangala University of Technology Thanyaburi. She was born on 27th April 1988 in Bangkok Province. She received the B.Sc. (Agricultural Science) from Thaksin University, Thailand. She obtained her M.Sc. (Postharvest Technology) from King Mongkut's University of Technology Thonburi (KMUTT), Bangkok, Thailand. During her Master's, she was a research exchange student at the laboratory of the Center for Environment, Health and Field Science at Chiba University, Japan, supported by the Japan Student Services Organization (JASSO) Scholarship. Her research topic was "Chlorophyll degradation mechanism and its control in lime fruit". She received his Ph.D. in Biotechnology from Thaksin University, Thailand in 2020. During Ph.D. course he has supported by the Thailand Research Fund under the Research and Researchers for Industries (RRI) Scholarship. Her research interests are Postharvest Technology: Postharvest Handling System of Fruits and vegetables; Carotenoid accumulation and Juice Sacs culture *In vitro* in Citrus fruit.

Face Photo and Sketch Synthesis Using CycleGAN: The Role of Residual Blocks and Spectral Normalization

Nuntipat Phisutthangkoon¹, Patikorn Anchuen^{2*}, Thiansiri Luangwilai³,
and Phummipat Daungklang⁴

^{1,4}Department of Computer Science, Navaminda Kasatriyadhiraj Royal Air Force Academy,
Saraburi, Thailand

²Office of Graduate Studies, Navaminda Kasatriyadhiraj Royal Air Force Academy, Saraburi, Thailand

³Mathematics Department, Navaminda Kasatriyadhiraj Royal Air Force Academy, Saraburi, Thailand
E-mail: nuntipat_phis@rtaf.mi.th, patikorn_an@rtaf.mi.th*, Thiansiri.L@gmail.com,
phummipat_d@rtaf.mi.th

Received: August 19, 2025 / Revised: November 11, 2025 / Accepted: January 30, 2026

Abstract—This study investigates the problem of face photo and sketch synthesis using CycleGAN, with a particular focus on architectural choices that affect model performance. A major challenge in this field is the limited availability of paired face photo-sketch datasets, which makes it difficult to develop and evaluate effective models. To address this limitation, this research demonstrates how to generate face sketches and photos from small datasets, offering a practical solution for expanding existing data collections in future applications, particularly in law enforcement and forensic sketch analysis. Four CycleGAN configurations were designed by varying the number of residual blocks in the generator and applying spectral normalization to the discriminator. Experiments were conducted on the CUHK Face Sketch Database, with quantitative evaluations using SSIM and PSNR, as well as qualitative assessments via human visual inspection. The results show that increasing generator depth and incorporating spectral normalization in the discriminator both improve image quality and training stability, although these benefits come at the cost of increased training time. Notably, the configuration with ResNet9 and spectral normalization consistently achieved the highest scores, with SSIM and PSNR values of 0.702 and 18.53 for sketch-to-photo translation, and 0.702 and 17.35 for photo and sketch translation, and produced the most visually convincing results in both tasks. These findings highlight the importance of architectural design choices for effective face photo-sketch translation and provide guidance for future work in related applications.

Index Terms—CycleGAN, Face Photo-Sketch Synthesis, Generative Adversarial Network,

Residual Blocks, Spectral Normalization, Image-To-Image Translation, CUHK Face Sketch Database, SSIM, PSNR

I. INTRODUCTION

Face photo-to-sketch synthesis is a crucial and challenging task in computer vision, with a wide range of real-world applications, including law enforcement, digital entertainment, and forensic art. Being able to accurately translate a face photo into a hand-drawn sketch (and vice versa) can greatly assist in criminal investigations and provide new tools for creative industries. However, generating realistic and high-quality sketches from face photos remains difficult due to the significant differences in style, structure, and texture between photographs and sketches.

Traditional approaches often rely on large paired datasets or hand-crafted features, which may fail to capture fine details and can result in images that look unnatural or overly smooth. In practice, the availability of paired face photo-sketch datasets is very limited, which not only makes it difficult to train effective models but also hinders the creation of new and diverse datasets for research and practical use. Developing methods that can generate convincing photo-sketch pairs from small or unpaired datasets can help expand existing collections and support a broader range of applications in this area.

With the rise of deep learning, models such as convolutional Neural Networks (CNNs) have achieved strong results in fields like image classification [1], [2]. Generative Adversarial Networks (GANs), especially CycleGAN, have become popular for image-to-image translation tasks that do not require paired data. CycleGAN enables learning mappings between two different domains using only unpaired examples, making it suitable for real-world scenarios

where paired face photo-sketch data are limited and enabling the automatic creation of synthetic datasets. Despite their success, standard CycleGAN models still face challenges. In many cases, the generated images may lose important details, or the training process may be unstable, leading to lower-quality results. Recent research [3], [4] suggests that architectural choices—such as increasing the number of residual blocks in the generator or applying spectral normalization in the discriminator—can help improve the quality and stability of GAN-based image translation, but their combined effects have not been fully explored in the context of face photo-sketch synthesis.

To address these gaps, this study aims to systematically compare the impact of generator depth and spectral normalization in the discriminator on the performance of CycleGAN for face photo-sketch synthesis. Four model configurations were designed and tested on a public dataset (the CUHK dataset), using a combination of quantitative metrics and human visual assessment to evaluate the results.

In summary, this work makes the following key contributions. First, we conduct a systematic comparison of four CycleGAN configurations by varying the number of residual blocks in the generator and applying spectral normalization to the discriminator, which has not been extensively examined in prior studies. Second, we provide a comprehensive evaluation framework that integrates quantitative metrics (SSIM and PSNR), human visual assessment, analysis of training stability, comparison with existing methods, and the total training time as a measure of computational cost. Third, our results highlight how specific architectural choices influence model behavior under limited-data conditions, offering practical guidance for real-world applications. Finally, the proposed approach enables the generation of additional face photo-sketch pairs from small datasets, contributing to future dataset expansion and research development.

II. RELATED WORK

The problem of translating between face photos and sketches has attracted considerable attention in computer vision research. Early approaches relied on traditional image processing techniques such as Principal Component Analysis (PCA), eigenfaces, or patch-based synthesis [5]-[11]. These methods required carefully paired datasets and handcrafted features, and although they produced recognizable sketches, they often struggled to preserve fine details or generate realistic texture.

With the rise of deep learning, Generative Adversarial Networks (GANs) significantly improved image translation performance. Methods such as pix2pix [9] demonstrated the effectiveness of adversarial learning but still relied on paired datasets,

which are difficult to obtain for photo-sketch translation. CycleGAN [3] addressed this limitation through a cycle-consistency loss, enabling translation using only unpaired data and making it one of the most influential frameworks for both photo-to-sketch and sketch-to-photo synthesis. Extensions such as the Identity-Aware CycleGAN (IACycleGAN) [12] introduced an identity-preserving perceptual loss using a face-recognition network to enhance feature consistency, while FCN-based approaches, e.g., [13], used generative-discriminative learning to extract facial structures but often produced blurrier results and lacked fine detail.

Despite these advances, challenges remain regarding the quality and stability of GAN-based synthesis. Several studies have proposed architectural improvements to enhance the expressiveness and stability of CycleGAN models. Increasing the number of residual blocks improves the generator's ability to capture complex facial patterns, whereas applying spectral normalization [4] stabilizes discriminator learning by constraining gradient behavior. Evaluating the quality of synthesized images typically involves SSIM and PSNR metrics to assess structural similarity and signal fidelity, while human visual assessment remains important for identifying perceptual artifacts.

More recent work has explored strategies to increase robustness and realism. A semi-supervised CycleGAN [14] jointly leveraged paired and unpaired samples to improve identity preservation under real-world conditions. DiffFaceSketch [15] introduced a sketch-guided latent diffusion model capable of generating high-fidelity face images with sharper textures and greater realism than traditional GAN-based methods. These developments reflect the increasing emphasis on identity retention, structural fidelity, and robustness in face photo-sketch synthesis.

While substantial progress has been made, there is still limited research that systematically evaluates how specific architectural components—particularly generator depth and the use of spectral normalization—jointly affect CycleGAN performance. Moreover, few studies combine quantitative metrics with human visual analysis to assess both structural accuracy and perceptual realism. This study addresses this gap by providing a comprehensive comparison of different CycleGAN configurations for face photo-sketch translation.

III. METHODOLOGY

This section details the experimental setup for face photo-to-sketch and sketch-to-photo synthesis, including dataset preparation, model architectures, training procedures, evaluation metrics, and the hardware/software environment required to reproduce the study.

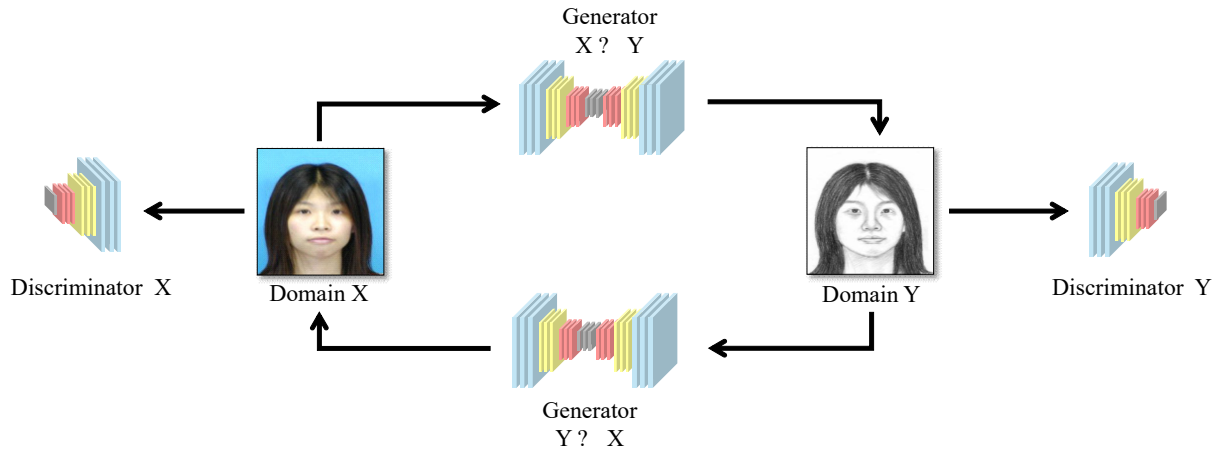


Fig. 1. The architecture of CycleGAN for face photo-to-sketch and sketch-to-photo synthesis. The model consists of two generators ($X \rightarrow Y$ and $Y \rightarrow X$) and two discriminators (X and Y), enabling bidirectional translation between the photo domain (Domain X) and the sketch domain (Domain Y) using unpaired training data and cycle-consistency loss.

A. Dataset

All experiments in this study were conducted using the CUHK Face Sketch Database (CUFS) [6], a well-established benchmark for face photo-to-sketch synthesis. The CUFS dataset comprises 188 image pairs, each containing a real face photograph and its corresponding hand-drawn sketch. In this work, we define domain X as the set of real face photos, and domain Y as the set of hand-drawn face sketches. This domain notation is adopted throughout the paper to describe the translation tasks and the corresponding outputs. The dataset was constructed and made publicly available by the Multimedia Laboratory of the Chinese University of Hong Kong, and can be accessed through their official website.

To ensure consistency in training and evaluation, all images were resized to 128×128 pixels and normalized to the range $[-1, 1]$. The 128×128 resolution is commonly used in CycleGAN studies because it preserves essential facial structure while keeping the computational cost low. Normalizing to $[-1, 1]$ follows standard GAN practice, helping stabilize training and improving convergence. The preprocessing was implemented via a sequence of transformations, including resizing, conversion to tensor format, and normalization. Importantly, all images were processed and maintained in RGB color format to ensure that the full range of visual information available in both photographs and sketches was utilized during training and evaluation.

For model training and evaluation, the dataset was partitioned into training and testing subsets via a random split. In total, 70% of the image pairs (131 pairs) were allocated for training, and the remaining 30% (57 pairs) were reserved for testing. A fixed

random seed controlled the randomization process to guarantee reproducibility of the experimental results. This train/test split ensures a robust and unbiased assessment of model performance across all configurations studied in this work.

B. Network Architecture

The network architecture employed in this study is fundamentally based on the CycleGAN framework, which is designed for unpaired image-to-image translation tasks. The architecture of the proposed framework is illustrated in Fig. 1.

To investigate the effects of architectural modifications, four experimental configurations—referred to as Configs A, B, C, and D—were devised, as shown in Table I, which summarizes their structural differences and highlights how each variation influences the overall translation process.

TABLE I
SUMMARY OF EXPERIMENTAL CONFIGURATIONS

Configs	Generator	Discriminator	Description
A	ResNet-6	Standard	Baseline (6 Residual Blocks, no SN)
B	ResNet-9	Standard	More Residual Blocks (9), no SN
C	ResNet-6	+SpectralNorm	6 Residual Blocks, Discriminator with SN
D	ResNet-9	+SpectralNorm	9 Residual Blocks, Discriminator with SN

Source: SN = Spectral Normalization

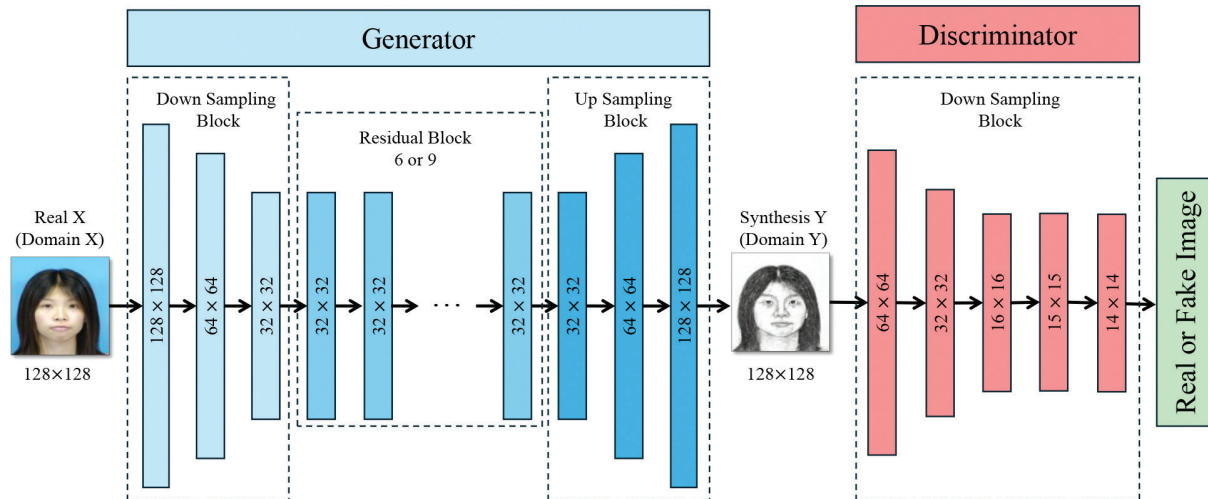


Fig. 2. The structure of the generator and the discriminator

Each configuration systematically varies two key components of the CycleGAN architecture: The number of residual blocks in the generator and the use of spectral normalization in the discriminator. Specifically, Configs A and B utilize standard discriminators, with their generators comprising 6 and 9 residual blocks, respectively. Configs C and D extend this design by incorporating spectral normalization into the discriminator, again paired with generators featuring either 6 (C) or 9 (D) residual blocks. This controlled experimental design enables a focused analysis of how these architectural choices affect the quality and stability of face photo-sketch synthesis.

Fig. 2 shows the structure of the generator and the discriminator. The generator in all configurations follows a ResNet-based structure, closely aligned with the original CycleGAN design. The input image, represented in RGB color space, is first processed by a convolutional layer with a 7×7 kernel, followed by instance normalization and a ReLU activation function to extract low-level features. Subsequently, the feature maps are progressively downsampled through two convolutional layers with a stride of 2, each followed by instance normalization and ReLU, thereby reducing the spatial dimensions while increasing feature abstraction. At the core of the generator, a sequence of either six or nine residual blocks (depending on the configuration) enables the model to capture complex image transformations. Each residual block consists of two convolutional layers with 3×3 kernels, each followed by instance normalization and ReLU, with skip connections facilitating the flow of information and gradients, thereby allowing the network to maintain structural coherence while learning more refined and expressive feature representations.

The residual block can be mathematically formulated as

$$y = \mathcal{F}(x, \{W_i\}) + x \quad (1)$$

where x is the input, $\mathcal{F}(x, \{W_i\})$ represents the residual mapping implemented by two convolutional layers (with instance normalization and ReLU), and W_i are the weight matrices of these layers.

After the residual blocks, the architecture performs two upsampling stages using transposed convolutional layers with a stride of 2, each followed by instance normalization and ReLU, thereby restoring the spatial resolution to the original input size. The final output is produced via a 7×7 convolutional layer with a Tanh activation function, mapping the feature representation back to an RGB image with pixel values in the range $[-1, 1]$. ReLU activations are employed throughout the generator, except at the output layer.

The discriminator adopts a PatchGAN architecture, which is well-suited for assessing the realism of local image patches rather than the entire image. This design comprises a series of convolutional layers with 4×4 kernels. The initial layer reduces spatial dimensions and is followed by a LeakyReLU activation. Subsequent layers further downsample the input, each incorporating instance normalization (except the first layer) and LeakyReLU activations. The final output is a one-channel patch map indicating the probability of each patch being real or fake. In configurations C and D, every convolutional layer in the discriminator is augmented with spectral normalization, a regularization technique known to enhance GAN training stability by constraining each layer's Lipschitz constant.

Specifically, spectral normalization is defined as

$$\bar{W} = \frac{W}{\sigma(W)} \quad (2)$$

where W is the weight matrix of a layer and $\sigma(W)$ denotes its largest singular value (spectral norm).

By changing only the number of residual blocks in the generator and whether spectral normalization is used in the discriminator, we can be confident that any differences in model performance come directly

from these design choices. This clearly shows how important these two factors are for face photo-sketch synthesis.

C. Training Details

The training procedure in this study employs a combination of adversarial, cycle-consistency, and identity losses, which jointly guide the generator and discriminator during optimization. The adversarial component is implemented using mean squared error (MSE), a choice widely adopted in CycleGAN-based models for its stability compared with binary cross-entropy. Cycle-consistency loss ($\lambda_{\text{cycle}} = 10$) ensures that translations between the two domains remain structurally coherent, while the identity loss ($\lambda_{\text{identity}} = 5$) helps preserve domain-specific characteristics, both computed using L1 distance. Optimization is carried out using the Adam optimizer with a learning rate of 0.0002 and momentum parameters $(\beta_1, \beta_2) = (0.5, 0.999)$. These settings follow established best practices in GAN training and were chosen for their proven stability in unpaired image-to-image translation tasks. A linear decay scheduler is applied after epoch 100 to encourage smooth convergence. Each model is trained for 200 epochs using a batch size of 1, which is standard in CycleGAN due to memory constraints and the patch-based discriminator design. The training data are reshuffled at the start of every epoch to reduce sampling bias and improve generalization. The complete set of training parameters and hyperparameters used in this study is summarized in Table II.

In addition to these design choices, several aspects of the CycleGAN framework inherently mitigate the risk of overfitting. The cycle-consistency and identity losses act as implicit regularizers by enforcing structural fidelity across translation cycles, preventing the generator from memorizing individual samples. Furthermore, the use of spectral normalization in Configurations C and D constrains the discriminator's behavior and promotes more stable gradients, which helps avoid discriminator overfitting. The combination of a small batch size, data shuffling, and a decaying learning rate also introduces stochasticity into the training process, reducing the likelihood that the model fits noise or sample-specific artifacts. These factors collectively contribute to stable training and help maintain generalization despite the relatively small dataset.

TABLE II
KEY TRAINING PARAMETERS AND HYPERPARAMETERS

Parameter	Value/Setting	Description
Lambda_Cycle	10.0	Cycle Consistency Loss Weight
Lambda_Identity	5.0	Identity Loss Weight
Optimizer	Adam	Both G and D
Learning Rate	0.0002	Initial
Adam Betas	(0.5, 0.999)	-
LR Scheduler	LambdaLR	Linear Decay After Epoch 100
Batch Size	1	-
Image Size	128 x 128	-
Epochs	200	-
Residual Blocks	6 or 9	Per Config
Normalize	mean=0.5, std=0.5	Input/Output in [-1, 1]

For model management and monitoring, the weights of the generator and discriminator are saved as checkpoints every 20 epochs. Loss values for both individual batches and per-epoch averages are recorded in CSV files, facilitating detailed post hoc analysis and visualization. Furthermore, sample images generated by the model are saved at 20-epoch intervals, enabling ongoing qualitative assessment of the synthesis results throughout training. Fig. 3 presents illustrative examples of the synthesis process, highlighting the transformation between the photo and sketch domains.

All experimental methods in this study were implemented in Python 3 using PyTorch 2.7 and executed on a workstation equipped with an Intel Core i5-12600KF CPU, an NVIDIA GeForce RTX 3060 GPU with 12 GB of VRAM, and 32 GB of DDR4 RAM. Supporting libraries included torchvision, Pillow, numpy, pandas, matplotlib, and tqdm.

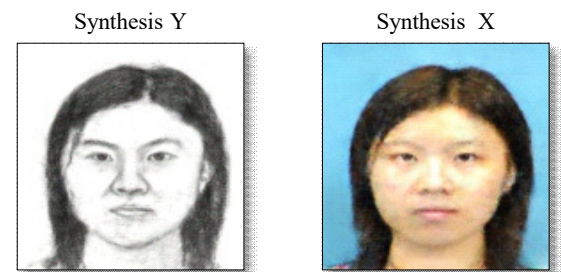


Fig. 3. Illustrative examples of the synthesis process. The left image (Synthesis Y) shows a synthesized face sketch generated from a real photo, while the right image (Synthesis X) shows a synthesized face photo generated from a real sketch.

D. Evaluation Metrics

The effectiveness of the proposed models was quantitatively evaluated using established image similarity metrics on the independent test set. The accuracy and perceptual quality of the synthesized images were assessed using two principal measures: The Peak Signal-to-Noise Ratio (PSNR) [16] and the Structural Similarity Index (SSIM) [17]. PSNR provides an objective evaluation of the pixel-level similarity between the generated images and their corresponding ground-truth images, with higher values indicating a closer similarity and lower distortion. Meanwhile, SSIM measures the similarity of brightness, contrast, and structure, which reflect how humans perceive image quality, and a higher SSIM score accordingly indicates greater perceptual similarity between the compared image pairs.

For the evaluation, each generated image, either a synthesized face photo or sketch, was paired with its corresponding real image based on the same index. (for example, synthesisX_xxx.png compared with realX_xxx.png, and synthesisY_xxx.png with realY_xxx.png, where domain X denotes real face photos and domain Y denotes hand-drawn sketches). We calculated both PSNR and SSIM for every image pair using standard tools from skimage. Metrics library to make sure the results were consistent and reliable. Then we showed the average PSNR and SSIM scores for each experiment, making it easy to see which model performed best.

IV. RESULTS AND DISCUSSION

To compare the performance of each model configuration, we examined both the training losses and the evaluation metrics, specifically SSIM and PSNR, on the test set, which consists of 57 previously unseen image pairs. The results are presented for both directions, photo-to-sketch (Domain Y) and sketch-to-photo (Domain X), and are reported across several training epochs.

A. Training Loss

Table III summarizes the generator and discriminator losses and the total training time for each configuration, reported as mean values with corresponding standard deviations (Mean $\pm$ SD). In this context, G_loss_avg refers to the average total generator loss, which combines adversarial, cycle-consistency, and identity losses, calculated across all test batches and epochs. $D_X_loss_avg$ and $D_Y_loss_avg$ represent the average discriminator losses for domains X and Y, respectively, computed over all test batches. The table also shows the total training time in minutes for each model, allowing for a direct comparison of computational efficiency.

TABLE III
A SUMMARY OF THE AVERAGE GENERATOR AND DISCRIMINATOR LOSSES ($\pm$ SD) AND TOTAL TRAINING TIME FOR EACH CONFIGURATION

Configs	$G_loss_avg \pm SD$	$D_X_loss_avg \pm SD$	$D_Y_loss_avg \pm SD$	Total Training Time (m)
A	2.69 (± 0.49)	0.02 (± 0.04)	0.01 (± 0.04)	38.36
B	2.71 (± 0.49)	0.01 (± 0.04)	0.01 (± 0.03)	52.32
C	2.62 (± 0.46)	0.01 (± 0.03)	0.01 (± 0.03)	43.93
D	2.57 (± 0.46)	0.02 (± 0.04)	0.01 (± 0.03)	51.77

Note: SN = Spectral Normalization

To show how the loss values change during training, Figs. 4 to 6 illustrate the trends of G_loss_avg , $D_X_loss_avg$, and $D_Y_loss_avg$ over 200 training epochs for each model. Fig. 6. Comparison of discriminator Y loss average across config configurations. All configurations show a clear decrease in loss values as training progresses, which suggests stable learning for both the generator and discriminator networks.

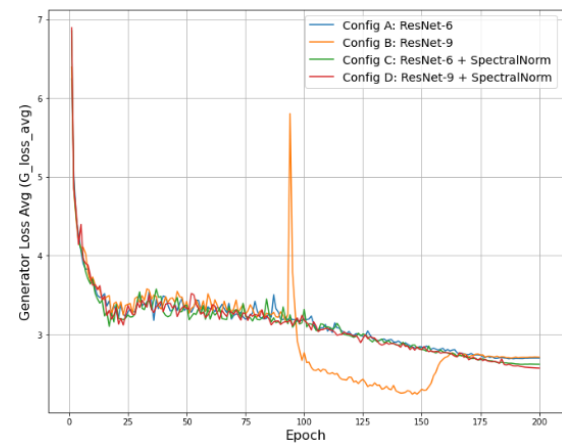


Fig. 4. Comparison of generator loss average across configs

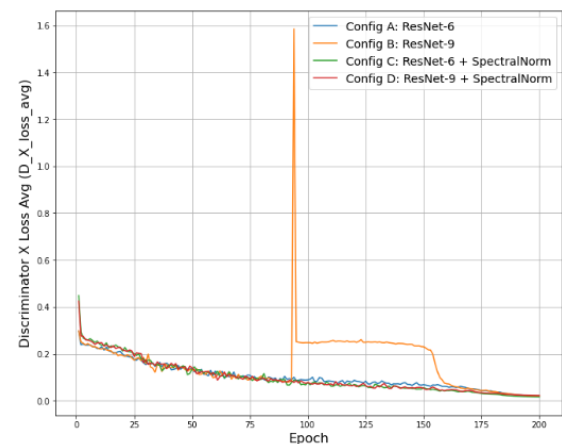


Fig. 5. Comparison of discriminator X loss average across configs

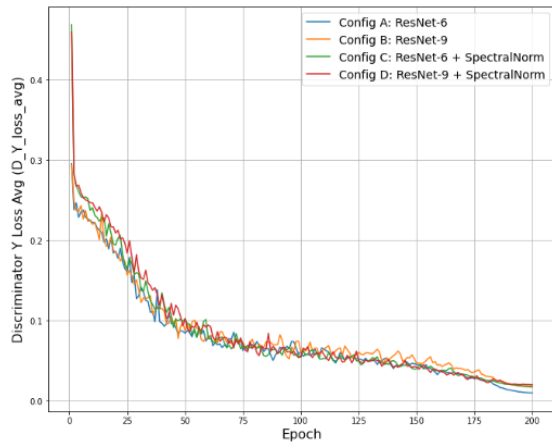


Fig. 6. Comparison of discriminator Y loss average across configs

However, some distinct patterns can be observed among the different configurations. Notably, as shown in Fig. 4 and Fig. 5, Config B (ResNet9 without SpectralNorm) exhibits a sharp spike in both generator and discriminator X losses around epoch 100, indicating a momentary instability during training. This fluctuation is not observed in the other configurations, which remain relatively stable throughout. However, it is noteworthy that Config B can recover from this instability, and its loss values gradually return to levels comparable to those of the other models by around epoch 160, demonstrating the model's ability to adapt and stabilize as training progresses.

From the summary table (Table III), it is evident that Config D (ResNet9 with SpectralNorm) achieves the lowest generator loss (2.57), indicating that this setup helps the generator optimize its objective most effectively. In contrast, Config B reports the highest generator loss (2.71), which may suggest lower stability or less effective learning than the other settings. For the discriminator losses, all configurations produce low values, but those that incorporate SpectralNorm (Configs C and D) tend to have slightly higher $D_Y_loss_avg$, as visualized in Fig. 6. This may indicate that the generator finds it more challenging to fool the discriminator when SpectralNorm is applied, potentially helping the discriminator to learn more robust and discriminative features. As a result, this added difficulty can lead to a better balance between the generator and discriminator during training, which is important for achieving stable and high-quality image synthesis.

In terms of computational efficiency, the results show that deeper architectures and the inclusion of SpectralNorm increase training time. For example, Config D (ResNet9+SN) required the longest training time (51.77 minutes), while Config A (ResNet6) completed training the fastest (38.36 minutes). This demonstrates a trade-off between model complexity and computational cost. While more advanced

configurations (such as Config D) deliver better performance in terms of loss values and image quality, they also demand more processing time. Therefore, the choice of model configuration should consider both the desired output quality and available computational resources.

The loss curves further reveal that while all models converge to relatively low loss values, Config D consistently demonstrates the most stable and lowest losses overall. This pattern supports the idea that using both a deeper generator (ResNet9) and SpectralNorm in the discriminator leads to more effective and stable training, resulting in better model performance in later sections.

B. SSIM and PSNR (Sketch-to-Photo: Domain X)

When evaluating the performance across all epochs, it is clear that Config D (ResNet9 with SpectralNorm) consistently produces the highest or near-highest values for both SSIM and PSNR. Additionally, this selection is based entirely on the summary statistics from the full test set of 57 image pairs. At epoch 160, this configuration achieves the best scores with an SSIM of 0.702 and a PSNR of 18.54. These results indicate that the images generated by Config D at this point are the most similar to and of the highest quality among the ground-truth photos.

Epoch 160 was chosen for further analysis because it represents the peak performance in both the average and the standard deviation of SSIM and PSNR, as shown in Table IV. The decision to focus on Config D was made because it demonstrated the most consistent superiority across all epochs according to these metrics.

TABLE IV
MEAN SSIM AND MEAN PSNR VALUES FOR EACH MODEL CONFIGURATION AND EPOCH ON THE SKETCH-TO-PHOTO (DOMAIN X) FROM THE COMPLETE TEST SET (N = 57 IMAGE PAIRS)

Epoch	Configs	Mean SSIM $\pm$ SD	Mean PSNR $\pm$ SD
140	A	0.6903 ($\pm$ 0.0448)	18.03 ($\pm$ 2.04)
	B	0.6949 ($\pm$ 0.0544)	18.04 ($\pm$ 2.64)
	C	0.7017 ($\pm$ 0.0454)	18.40 ($\pm$ 2.27)
	D	0.6999 ($\pm$ 0.0479)	18.38 ($\pm$ 2.53)
160	A	0.6926 ($\pm$ 0.0476)	18.43 ($\pm$ 2.54)
	B	0.7115 ($\pm$ 0.0474)	18.36 ($\pm$ 2.57)
	C	0.7011 ($\pm$ 0.0459)	18.45 ($\pm$ 2.30)
	D	0.7023 ($\pm$ 0.0479)	18.53 ($\pm$ 2.52)
180	A	0.6862 ($\pm$ 0.0489)	18.30 ($\pm$ 2.52)
	B	0.6946 ($\pm$ 0.0492)	18.24 ($\pm$ 2.47)
	C	0.6902 ($\pm$ 0.0473)	18.39 ($\pm$ 2.24)
	D	0.6958 ($\pm$ 0.0493)	18.45 ($\pm$ 2.50)
200	A	0.6809 ($\pm$ 0.0493)	18.28 ($\pm$ 2.48)
	B	0.6848 ($\pm$ 0.0507)	18.17 ($\pm$ 2.47)
	C	0.6851 ($\pm$ 0.0478)	18.35 ($\pm$ 2.23)
	D	0.6874 ($\pm$ 0.0493)	18.39 ($\pm$ 2.49)

Note: SN = Spectral Normalization

To further examine the robustness and reliability of this result, Fig. 7 shows the histograms of SSIM and PSNR values for Config D at epoch 160. The histogram plots reveal that most SSIM and PSNR values are concentrated within a narrow range, with very few outliers. This concentration indicates stable model performance across all 57 test pairs, suggesting that the model not only performs well on average but also produces high-quality results for nearly all test images. Therefore, these findings can be considered both trustworthy and generalizable.

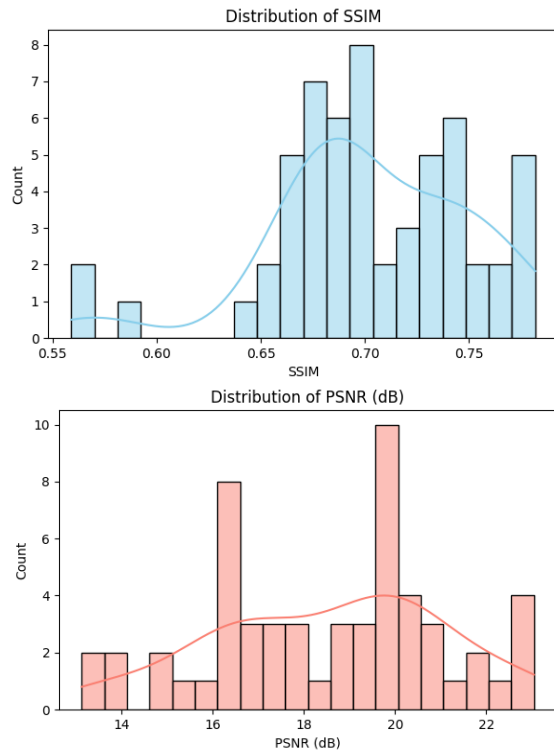


Fig. 7. Distribution of SSIM (top) and PSNR (bottom) values for Config D (ResNet9 with SpectralNorm) at epoch 160 on the sketch-to-photo (Domain X) translation task

Config C (ResNet6 with SpectralNorm) also performs well, often approaching or slightly exceeding the performance of Configs A and B (which do not use SpectralNorm), especially on SSIM. Overall, these results highlight the effectiveness of SpectralNorm in preserving structural similarity and details in the generated images.

C. SSIM and PSNR (Photo-to-Sketch: Domain Y)

For the photo-to-sketch translation task (Domain Y), the evaluation shows a pattern similar to that observed in the sketch-to-photo direction. As presented in Table V, Config D (ResNet9 with SpectralNorm) again achieves the highest SSIM (0.702) and PSNR (17.35) at epoch 160, standing out as the best performer among all configurations. This suggests that using a deeper generator together with SpectralNorm helps the model produce sketches that are more similar to

the real hand-drawn sketches in terms of both structure and pixel quality.

TABLE V
MEAN SSIM AND MEAN PSNR VALUES FOR EACH MODEL CONFIGURATION AND EPOCH ON THE PHOTO-TO-SKETCH (DOMAIN Y) FROM THE COMPLETE TEST SET (N = 57 IMAGE PAIRS)

Epoch	Configs	Mean SSIM $\pm$ SD	Mean PSNR $\pm$ SD
140	A	0.6878 ($\pm$ 0.0448)	16.05 ($\pm$ 2.42)
	B	0.6916 ($\pm$ 0.0537)	16.70 ($\pm$ 3.03)
	C	0.7029 ($\pm$ 0.0526)	17.00 ($\pm$ 3.10)
	D	0.6969 ($\pm$ 0.0496)	16.75 ($\pm$ 2.74)
160	A	0.6934 ($\pm$ 0.0494)	17.18 ($\pm$ 2.85)
	B	0.6909 ($\pm$ 0.0537)	16.85 ($\pm$ 3.23)
	C	0.6962 ($\pm$ 0.0556)	16.70 ($\pm$ 3.21)
	D	0.7022 ($\pm$ 0.0524)	17.35 ($\pm$ 2.95)
180	A	0.6798 ($\pm$ 0.0491)	16.93 ($\pm$ 3.00)
	B	0.6847 ($\pm$ 0.0546)	16.71 ($\pm$ 3.12)
	C	0.6921 ($\pm$ 0.0561)	16.74 ($\pm$ 3.25)
	D	0.6996 ($\pm$ 0.0546)	17.29 ($\pm$ 2.92)
200	A	0.6726 ($\pm$ 0.0512)	16.91 ($\pm$ 2.99)
	B	0.6784 ($\pm$ 0.0567)	16.67 ($\pm$ 3.07)
	C	0.6838 ($\pm$ 0.0589)	16.66 ($\pm$ 3.28)

Note: SN = Spectral Normalization

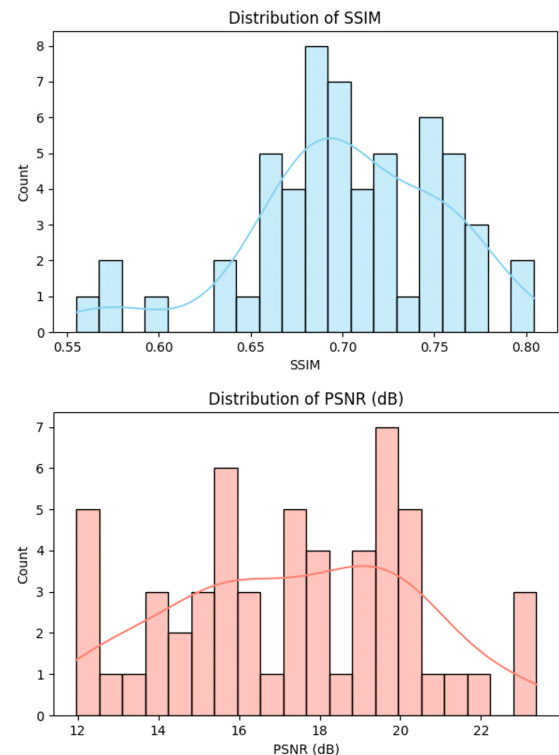


Fig. 8. Distribution of SSIM (top) and PSNR (bottom) values for Config D (ResNet9 with SpectralNorm) at epoch 160 on the sketch-to-photo (Domain Y) translation task.

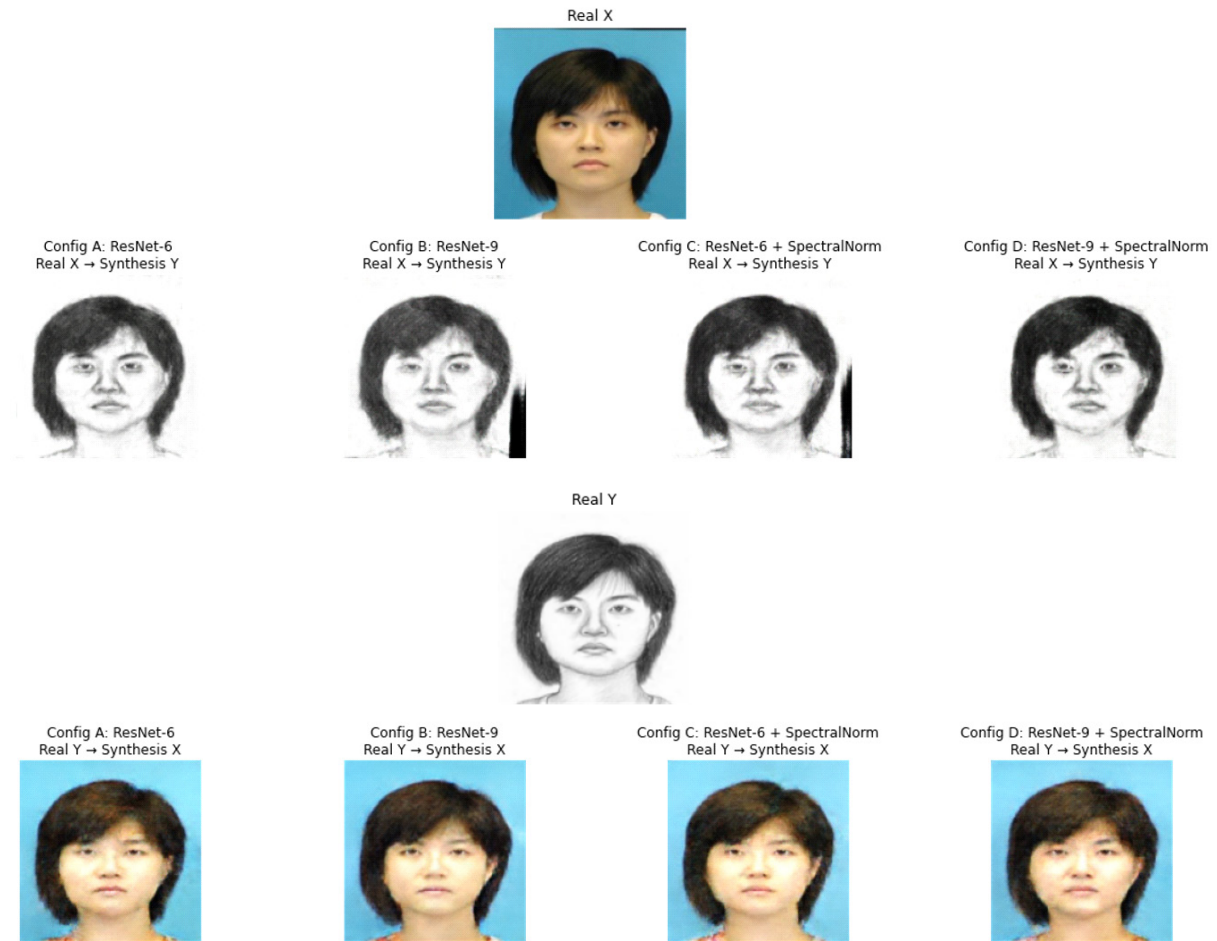


Fig. 9. Sample results of photo-to-sketch (top rows) and sketch-to-photo (bottom rows) translation for two different test subjects. Each column shows results from a different model configuration. The real sketch (Real Y) and real photo (Real X) are shown for reference.

The histogram plots in Fig. 8 further confirm this finding. The SSIM and PSNR values for Config D at epoch 160 are mostly concentrated within a narrow range, indicating that the model performs consistently well across all 57 test images. Only a few results fall outside the main range, which means the model rarely fails on individual test samples.

Overall, these results reinforce the benefits of using SpectralNorm in the discriminator and a deeper generator architecture for photo-to-sketch synthesis. The combination leads to both higher average performance and greater consistency, which are desirable characteristics for real-world face sketch generation applications.

D. Performance Comparison with Existing Methods

To further contextualize the proposed approach

within existing research, we compared the performance of our best configuration (ResNet-9 with Spectral Normalization) against several representative models commonly used for face photo-sketch synthesis, including cGAN-based translation [9], CycleGAN [3], the Identity-Aware CycleGAN (IACycleGAN) [12], and the FCN-based generative-discriminative framework [13]. Table VI reports the SSIM scores for both translation directions: sketch-to-photo and photo-to-sketch.

Across both tasks, the proposed model delivers competitive or superior performance. For sketch-to-photo synthesis, our method (SSIM = 0.7023) surpasses CycleGAN, cGAN, and the FCN approach, while performing comparably to IACycleGAN (SSIM = 0.7122), which benefits from an identity-preserving loss derived from a face-recognition network.

TABLE VI
COMPARISON OF SSIM VALUES ACROSS FACE PHOTO-SKETCH SYNTHESIS MODELS

	cGAN [9]	CycleGAN [3]	IACycleGAN [12]	FCN [13]	ResNet-9+SN (Ours)
Sketch-to-photo	0.6332	0.6313	0.7122	0.6884	0.7023
Photo-to-sketch	0.4938	0.4961	0.6495	0.5742	0.7022

For photo-to-sketch translation, the proposed configuration achieves the highest SSIM score. This suggests that combining a deeper generator with spectral normalization improves the model's ability to capture structural and textural characteristics essential for high-quality sketch generation.

E. Visual Comparison of Synthesized and Real Images

To complement the quantitative evaluation, Fig. 9 presents examples of both photo-to-sketch and sketch-to-photo translation results for different configurations. These visual comparisons help illustrate differences in image quality that are sometimes easier for the human eye to appreciate than numeric scores alone.

For the photo-to-sketch task (Domain Y), images generated by Config D (ResNet9 with SpectralNorm) are generally the most faithful to real sketches. Config D produces images with sharper lines and clearer details, especially around areas such as the hair ends, which closely resemble those in the real sketches. The generated sketches from Config D also show less blurring than those from the other configurations, which often produce softer or less well-defined edges. The overall face structure, including the hair outline, is more consistent with the real sketch, making the results look more natural and visually pleasing.

For the sketch-to-photo task (Domain X), at first glance, all configurations produce photo-like images that appear quite similar. However, upon closer inspection, some differences become noticeable. In particular, Config D stands out for its ability to generate more accurate and detailed features, especially in areas such as hair texture, hair ends, and skin tone. For example, in the red-circled areas shown in Fig. 10, the hair edges generated by Config D are more consistent with the original photo, and the skin tones are more natural and less patchy than those produced by the other configurations. These subtle details contribute to a more realistic and visually appealing result.

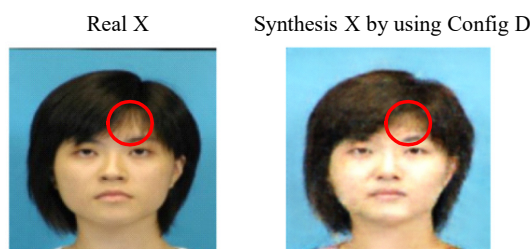


Fig. 10. Close-up comparison of hair detail in sketch-to-photo (Domain X) synthesis for two selected configurations. The highlighted areas show that Config D (ResNet9 + SpectralNorm) preserves hair edges and texture more accurately than other models.

A deeper examination of the model behavior shows that the architectural choices in this study directly influence both training stability and the visual

quality of the generated images. Config D performs best because its deeper generator (ResNet-9) provides a larger receptive field, allowing it to capture global facial structures and fine details such as contours, shading, and hair textures. Spectral normalization further stabilizes the discriminator and reduces overfitting, which is especially important for the small CUHK dataset. Consequently, Config D converges more smoothly, avoids the loss spikes observed in Config B, and consistently produces clearer, more accurate facial features. The narrow SSIM and PSNR distributions also indicate strong generalization despite limited data.

From a practical standpoint, the proposed approach is lightweight and easy to deploy, as it requires neither paired datasets nor external identity-recognition modules. This makes it suitable for real-world contexts such as law enforcement and forensic applications, where paired photo-sketch data are scarce. In addition, the ability to generate high-quality synthetic photo-sketch pairs from small datasets offers a useful pathway to expand training data for future applications.

V. CONCLUSION

This study investigated the use of CycleGAN for translating between face photos and sketches, focusing on comparing different architectural configurations. Four model variants were examined, differing in the number of residual blocks in the generator and the use of spectral normalization in the discriminator. The CUHK Face Sketch Database was used as the benchmark, and the models were evaluated both quantitatively and qualitatively.

The experimental results showed that increasing the depth of the generator (from ResNet6 to ResNet9) and applying spectral normalization to the discriminator led to improved performance on both photo-to-sketch and sketch-to-photo tasks. Specifically, the config D with ResNet9 and spectral normalization consistently achieved the highest SSIM and PSNR scores, indicating that the synthesized images were more similar to the real ones in both structure and visual quality. These findings were further supported by human visual assessment, which showed that this configuration preserved finer details, such as hair edges and skin tones, more effectively than the other models.

While the findings are promising, this study has certain limitations. The experiments were conducted on a relatively small dataset of 188 image pairs, which limits the diversity of facial attributes and sketching styles and may affect generalization to more varied real-world cases. In addition, the evaluation was based on a single benchmark dataset; testing the model on larger or more diverse datasets would further strengthen the conclusions.

Overall, this research highlights the importance of careful architectural choices, especially the combination of a deeper generator and spectral normalization in the discriminator, for improving the quality of face photo-sketch synthesis. The approach used in this study can serve as a reference for future work in related fields, such as law enforcement, forensic sketch analysis, and any application that benefits from reliable photo-sketch translation.

REFERENCES

- [1] T. Valeephakon, K. Orkphol, and P. Chaihuadjaroen, "Deep convolutional neural networks based on VGG-16 transfer learning for abnormalities peeled shrimp classification," *Int. Sci. J. Eng. Technol.*, vol. 6, no. 2, pp. 13-23, Dec. 2022.
- [2] N. Hongcharoen, P. Sanguansat, and S. Marukatat, "Fluke eggs detection and classification using deep convolutional neural network," *Int. Sci. J. Eng. Technol.*, vol. 6, no. 2, pp. 40-53, Dec. 2022.
- [3] J. -Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 2223-2232, <https://doi.org/10.1109/ICCV.2017.244>.
- [4] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," in *Proc. Int. Conf. Learn. Representations (ICLR)*, 2018, pp. 1-4.
- [5] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cogn. Neurosci.*, vol. 3, no. 1, pp. 71-86, 1991.
- [6] X. Wang and X. Tang, "Face photo-sketch synthesis and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 11, pp. 1955-1967, Nov. 2009.
- [7] A. Irfan, R. Mudassar, S. Muhammad, K. Seifedine, and R. Seungmin, "Union is strength: Improving face sketch synthesis by fusing outcomes of fully convolutional networks and random sampling locality constraint," *Alexandria Eng. J.*, vol. 61, no. 12, pp. 10727-10741, Apr. 2022.
- [8] M. Anvesha, S. Alistair, B. Marija, and J. Hossein, "Rhi3D-Gen: Analyzing rhinophyma using 3D face models and synthetic data," *Intell.-Based Med.*, vol. 8, Art. no. 100124, Nov. 2023.
- [9] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 1125-1134.
- [10] X. Gao, N. Wang, D. Tao, and W. Liu, "Face sketch-photo synthesis and retrieval using sparse representation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 8, pp. 1213-1226, Aug. 2012.
- [11] L. Wang, V. Sindagi, and V. Patel, "High-quality facial photo-sketch synthesis using multi-adversarial networks," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, 2018, pp. 83-90.
- [12] Y. Fu, W. Deng, J. Du, and J. Han, "Identity-aware CycleGAN for face photo-sketch synthesis and recognition," *Pattern Recognit.*, vol. 102, Art. no. 107249, Jun. 2020, <https://doi.org/10.1016/j.patcog.2020.107249>.
- [13] B. Cao, N. Wang, J. Li, Q. Hu, and X. Gao, "Face photo-sketch synthesis via full-scale identity supervision," *Pattern Recognit.*, vol. 124, Art. no. 108446, Apr. 2022, <https://doi.org/10.1016/j.patcog.2021.108446>.
- [14] C. Chaofeng, L. Wei, T. Xiao, and K. W. K. Kwan-Yee, "Semi-supervised Cycle-GAN for face photo-sketch translation in the wild," *Comput. Vis. Image Understand.*, vol. 235, Art. no. 103775, Oct. 2023, <https://doi.org/10.1016/j.cviu.2023.103775>.
- [15] Y. Peng, C. Zhang, H. Xu, T. Fukumoto, and K. Maeno, "DiffFaceSketch: High-fidelity face image synthesis with sketch-guided latent diffusion model," *arXiv*, 2023, [Online]. Available: <https://doi.org/10.48550/arXiv.2302.06908> [Accessed: Aug. 19, 2025].
- [16] T. Wang, Z. Hu, and Y. Guo, "An efficient lightweight network for image denoising using progressive residual and convolutional attention feature fusion," *Sci. Rep.*, vol. 14, Art. no. 9544, Apr. 2024, <https://doi.org/10.1038/s41598-024-60139-x>.
- [17] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600-612, Apr. 2004, <https://doi.org/10.1109/TIP.2003.819861>.



Nuntipat Phisutthangkoon received the B.S. degree in Applied Mathematics in 2011 and the M.S. degree in Computer Science in 2013, and the Ph.D. degree in Computer Science from King Mongkut's Institute of Technology, Ladkrabang (KMITL), Bangkok, Thailand. His research interests include Machine learning, parallel algorithms, and parallel and distributed computing.



Patikorn Anchuen received the B.E. degree in telecommunication engineering from the King Mongkut's Institute of Technology, Ladkrabang, Thailand, in 2014, and the M.E. degree in telecommunication engineering and the Ph.D. degree in telecommunication and computer engineering from Suranaree University of Technology, Thailand, in 2017 and 2020, respectively. He is currently a Lecturer with the Office of Graduate Studies at the Navaminda Kasatriyadhiraj Royal Air Force Academy in Thailand. His research interests include wireless communication, Artificial Intelligence (AI), Machine Learning (ML), and Artificial Neural Networks

Mobile Networks, Quality of Experience (QoE), and 5G communications. (ANN), Deep Reinforcement Learning (DRL), Genetic Algorithm (GA), Particle Swarm Optimization (PSO).



Thiansiri Luangwilai was born in 1982 in Nakhon Pathom, Thailand. He received a B.Sc. in computer science and mathematics from the University of New South Wales at the Australian Defense Force Academy in Canberra, Australia, in 2005, and a first-class honors in applied mathematics from the same university in 2006. He received a Ph.D. in Applied Mathematics and Statistics in 2012. His current research interests are in applied mathematics, mathematical modeling, industrial mathematics, numerical analysis and algorithms, bifurcation analysis, and dynamic bifurcation analysis.



Phummipat Daungklang was born in 1979 in Thailand. He received a B.Sc. in computer science from the Navaminda Kasatriyadhiraj Royal Air Force Academy in Bangkok in 2000. His current research interests are Artificial Intelligence (AI), Machine Learning (ML), Artificial Neural Networks (ANN), and Geographic Information Systems

A Hybrid of Syntactic Structure and Contextual Semantics for Enhanced Product Review Sentiment Analysis

Damrong Sattayawaksakul¹ and Pimpa Cheewaparakobkit^{2*}

¹Faculty of Business Administration, Asia-Pacific International University, Saraburi, Thailand

²Faculty of Information Technology, Asia-Pacific International University, Saraburi, Thailand

E-mail: damsat@apiu.edu, pimpa@apiu.edu*

Received: June 8, 2025 / Revised: January 7, 2026 / Accepted: January 30, 2026

Abstract—With the rapid growth of e-commerce platforms, analyzing customer reviews has become essential for understanding product perception and supporting data-driven decision-making. Traditional sentiment analysis methods often rely solely on sequential text representations, which may overlook the syntactic structures and contextual nuances present in customer feedback. This study aims to develop an interpretable hybrid framework that effectively captures both semantic meaning and syntactic dependencies to enhance sentiment prediction accuracy. The proposed model combines contextual embeddings from SentBERT with syntactic features extracted via Graph Feature Fusion. The process involves preprocessing customer reviews, extracting aspect-based semantic representations using SentBERT, and constructing syntactic graphs via dependency parsing. These features are merged in a fusion layer and refined using supervised contrastive learning to improve class separability in the sentiment space. SHAP Explainable AI is integrated to provide human-interpretable explanations for the sentiment predictions. The hybrid model achieves a sentiment classification accuracy of 97%, outperforming baseline methods including BERT (93%), GCN (88%), and classical machine learning algorithms. These findings highlight the effectiveness of integrating syntactic and contextual features in sentiment analysis. The framework can be applied to real-world e-commerce platforms to enhance automated review analysis, improve customer service insights, and support product development strategies.

Index Terms—Sentiment Analysis, Graph Feature Fusion, Product Review Analysis, Contextual Embedding

I. INTRODUCTION

In the modern digital landscape, e-commerce has revolutionized consumer behavior, with online product reviews becoming a significant factor in purchase decisions [1]. These reviews serve as a valuable source of information for both consumers and businesses, providing insights into customer satisfaction and product quality. However, the vast quantity of reviews on platforms like Amazon, Lazada, Twitter, and Shopee presents a challenge for extracting meaningful insights. The complexity of natural language, coupled with the sheer volume of user-generated content, necessitates the adoption of automated techniques such as sentiment analysis and review summarization powered by Artificial Intelligence (AI) [2], [3]. Sentiment analysis, a subfield of Natural Language Processing (NLP), has gained significant traction for its ability to automatically analyze, interpret, and categorize large volumes of text data [4]. Businesses increasingly rely on sentiment analysis to gauge customer opinions and facilitate data-driven decision-making. Classical machine learning algorithms, such as Naïve Bayes [5] and Support Vector Machine [6], excel at polarity detection but struggle with context-aware tasks. For instance, Karina and Sriani achieved 73% accuracy in classifying skincare reviews using Naïve Bayes, yet their model failed to capture mixed sentiments or attribute feedback to specific product aspects (e.g., packaging, texture) [7]. Similarly, SVM-based studies highlight trade-offs between precision and scalability, particularly when handling slang, abbreviations, or non-linear sentiment patterns [8]. These limitations are exacerbated in e-commerce, where reviews often contain conflicting opinions (e.g., “Great camera but poor battery life”) and domain-specific terminology. Recent breakthroughs in deep learning, particularly transformer-based models like BERT [9] and GPT

[3], have significantly advanced the state-of-the-art by leveraging self-attention mechanisms to capture long-range dependencies in text [10]. Nonetheless, these models often focus on holistic sentiment classification and may neglect syntactic structures and aspect-specific nuances, limiting interpretability and granular sentiment detection

Recent research has explored the integration of Graph Convolutional Networks (GCNs) into sentiment analysis frameworks. GCNs are adept at modeling complex relationships in data by representing text as graphs, where nodes correspond to words and edges capture syntactic dependencies. For instance, Li et al. proposed an edge-enhanced Graph Convolutional Network (GCN) model. This model enhances aspect-based sentiment classification by preserving feature integrity in syntactic graphs. While it improves syntactic structure preservation over traditional methods, its higher computational complexity and reliance on accurate syntactic parsing make it susceptible to noisy input. [11].

In another study by Hu and Li proposed a bi-channel GCN model that integrates phrase structure trees and hierarchical phrase matrices to enhance syntactic structural information. Although this approach improves sentiment classification accuracy, it introduces additional complexity to graph construction and may pose challenges for real-time applications [12].

To address these limitations, this study proposes a hybrid sentiment analysis framework that integrates semantic embeddings with syntactic structures to enhance both performance and interpretability. The key contributions of this research are as follows:

- 1) Context-Enriched Aspect Term Extraction: We proposed SentBERT for fine-grained identification of product-specific aspects (e.g., camera, battery life), allowing the model to distinguish sentiments directed at individual features.

- 2) Graph-Based Syntactic Modeling: A Graph Feature Fusion (GFF) module constructed via dependency parsing captures grammatical dependencies, facilitating a structural understanding of how sentiment-bearing words relate to specific product aspects.

- 3) Unified Feature Representation via Fusion Layer: Semantic and syntactic features are merged into a unified vector space, ensuring a more comprehensive sentiment representation.

- 4) Supervised Contrastive Learning for Robust Classification: This strategy enhances class separation by minimizing intra-class variance and maximizing inter-class distance, leading to a more discriminative feature space.

- 5) Explainable AI with SHAP: The framework

incorporates SHAP to generate interpretable visual explanations, highlighting which words contribute most significantly to sentiment predictions, enhancing transparency and user trust.

The remainder of this paper is structured as follows. Section II reviews related work on sentiment analysis methodologies. Section III details the proposed hybrid framework. Section IV presents the experimental results and performance evaluation of the proposed method. Finally, Section V offers the conclusions of this study.

II. LITERATURE REVIEW

The analysis of customer reviews in e-commerce has been extensively studied, leading to various methodologies for sentiment analysis and review summarization. Early approaches focused on sentiment analysis, aiming to classify reviews as positive, negative, or neutral. For instance, Peal et al., proposed a method to summarize customer reviews by extracting pros and cons, providing a concise overview of product feedback [13]. However, this approach primarily relied on predefined sentiment lexicons, which may not capture the nuanced expressions in user reviews. Building upon sentiment analysis, researchers have explored topic modeling to enhance review summarization. Kumar et al. employed topic modeling alongside sentiment analysis to summarize online customer reviews, enabling a more structured understanding of user opinions [14]. While effective in identifying prevalent themes, this method may struggle with the dynamic and evolving nature of language used in reviews.

The integration of deep learning techniques has further advanced sentiment analysis in e-commerce. Yang *et al.* introduced a model combining a sentiment lexicon with a Convolutional Neural Network (CNN) and an attention-based Bidirectional Gated Recurrent Unit (BiGRU) to analyze Chinese e-commerce product reviews [15]. This approach improved the accuracy of sentiment classification by capturing complex linguistic patterns. However, deep learning models often require large labeled datasets and are computationally intensive.

Classical machine learning algorithms, such as Naive Bayes (NB), Support Vector Machines (SVM), Decision Tree (DT), and Logistic Regression (LR), have been widely applied to sentiment classification tasks. Suryani et al. demonstrated the use of Naive Bayes for sentiment analysis on Twitter data related to electric cars [15]. Similarly, Sulthana et al. applied SVM to classify movie reviews [16]. However, these methods often rely on bag-of-words representations, which ignore word order and contextual dependencies. While offering computational

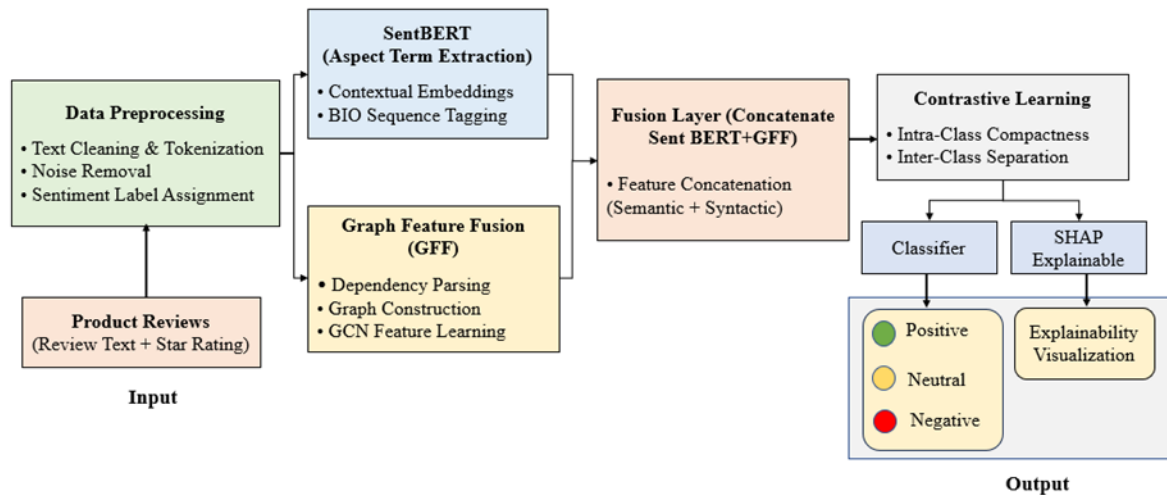


Fig. 1. The overview of the proposed method

While efficient and easy to implement, these models struggle with nuanced language, such as sarcasm, irony, and negation, which are prevalent in online reviews [17]. In the e-commerce domain, where reviews often contain complex product-specific jargon and multiple sentiments towards different aspects (e.g., “The battery life is great, but the camera is disappointing”), these methods often fall short. Furthermore, these models treat all words equally, failing to capture the relative importance of different terms in expressing sentiment. The challenge lies in adapting these techniques to understand the context and intricacies of online reviews.

Aspect-based sentiment analysis has also been a focal point in recent research. Wahyudi and Kusumaningrum investigated the performance of sentiment analysis on e-commerce user reviews using the Latent Dirichlet Allocation (LDA) algorithm, aiming to determine sentiment across different product aspects. [18]. While this approach allows for a more granular analysis, it may struggle to accurately identify aspects in reviews that contain implicit or ambiguous expressions.

Advancements in deep learning, particularly transformer-based models such as BERT [9] and its variants [3], have shown promising results in capturing contextual information to improve sentiment analysis. BERT’s bidirectional contextual understanding, achieved through self-attention mechanisms, allows it to model long-range dependencies in text. However, directly applying BERT to long documents, such as lengthy product reviews, can be computationally expensive due to the quadratic complexity of the attention mechanism.

Furthermore, while BERT captures contextual nuances, it does not inherently focus on specific product features, resulting in a less granular understanding of customer opinions. Recent studies have turned to Graph Convolutional Networks

(GCNs) to incorporate syntactic structure into sentiment analysis. GCNs treat sentences as graphs, where words are nodes and syntactic dependencies form the edges, allowing the model to propagate sentiment through syntactic paths.

III. PROPOSED METHOD

This research presents a hybrid framework for sentiment analysis of mobile phone product reviews that integrates semantic and structural information to improve classification accuracy and model interpretability. The proposed method begins with preprocessing raw review texts, followed by dual feature-extraction paths: semantic embeddings via SentBERT for aspect-term understanding and syntactic relationships via Graph Feature Fusion (GFF) using dependency parsing. These two complementary feature sets are then combined in a Fusion Layer to form a unified representation that captures both context and structure. To enhance the distinctiveness of sentiment categories—positive, neutral, and negative, a supervised contrastive learning strategy is applied, refining the feature space and improving classification robustness. A sentiment classifier then predicts the emotional polarity of the review, while a SHAP Explainable module generates interpretable explanations by highlighting which words most influenced the prediction. This architecture supports both high-performance sentiment analysis and human-centered explainability, making it well-suited for applications in e-commerce and customer feedback systems. The detailed workflow of the proposed method is illustrated in Fig. 1.

A. Data Preprocessing

The Data Preprocessing module transforms raw customer reviews into a clean, structured format for analysis. First, text in the “Comment” field undergoes standard NLP steps: Lowercasing, removal of

unnecessary punctuation, cleaning of HTML tags, special characters, and emojis, whitespace normalization, and optional spelling correction. Next, the “Stars” field is converted into sentiment labels—Positive (4-5 stars), Neutral (3 stars), and Negative (1-2 stars)—to serve as ground truth for training and evaluating the sentiment analysis model.

B. SentBERT (Aspect Term Extraction)

The SentBERT module, as shown in Fig. 2, plays a crucial role in the proposed framework by performing Aspect Term Extraction (ATE), which involves identifying specific entities or product features mentioned in customer reviews. In mobile phone reviews, common aspect terms include “battery life,”

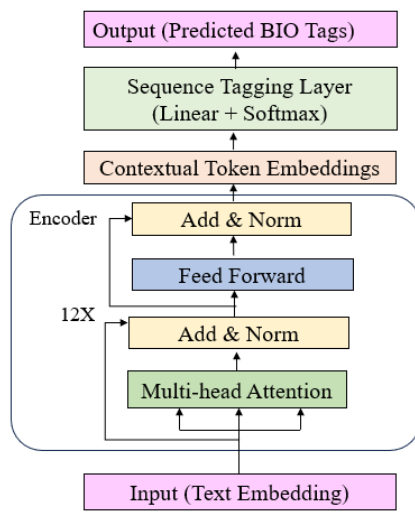


Fig. 2. The SentBERT architecture

“camera,” “price,” “display,” “performance,” and “design.” Accurate identification of these terms is essential for fine-grained sentiment analysis. SentBERT, a sentence-level variant of the BERT (Bidirectional Encoder Representations from Transformers) architecture [9], is adopted and fine-tuned for this sequence labeling task. Built on transformer encoder layers, SentBERT leverages multi-head self-attention, Add & Norm, and feed-forward sublayers to produce contextualized embeddings for each token in a sentence. These embeddings capture both semantic meaning and the relationships among words, making them ideal for identifying aspect terms in unstructured text. The self-attention mechanism, crucial for enabling the model to focus on important words within a sentence, computes attention weights using the following formula (1):

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

where Q , K , and V represent the query, key, and value matrices, respectively, and d_k is the dimensionality of the key vectors.

1) Problem Formulation as Sequence Tagging

To perform aspect term extraction using SentBERT, the input review text is prepared for token-level classification. This involves defining a structured tagging scheme and fine-tuning the model for sequence labeling. Given a customer review sentence $S=[w_1, w_2, \dots, w_n]$, where w_i represents the i -th token (word or sub-word unit), the objective of ATE is to assign a label l_i to each token w_i . This label indicates whether the token is part of an aspect term and, if so, its position within that term. We adopt the widely recognized BIO (Beginning, Inside, Outside) tagging scheme [19] for this purpose:

- **B-ASP**: Denotes the **B**eginning of an **ASP**ect term.

- **I-ASP**: Denotes an **I**nside token of an **ASP**ect term.

- **O**: Denotes an **O**utside token, meaning it is not part of an aspect term.

For instance, consider the review sentence: “The **battery life** is excellent, but the **camera** is disappointing”. The corresponding BIO labels would be: O, B-ASP, I-ASP, O, O, O, O, B-ASP, O, O.

2) SentBERT as a Sequence Tagger

The BERT model is used as a sequence tagger because it generates rich, contextualized word embeddings by considering the entire input sequence. This capability is crucial for accurately identifying aspect terms, which often depend on surrounding words for their meaning. The operational flow of the SentBERT-based sequence tagger is as follows:

• Input Representation

Each review sentence is broken into words (or word parts) using BERT’s WordPiece tokenizer. Special markers, (CLS) at the start and (SEP) at the end, are added to help SentBERT understand the text. These words and markers are then converted into numerical embeddings that combine information about the words themselves, their segment (which indicates whether they belong to different sentences), and their order in the sentence.

• Contextual Embedding Generation

SentBERT processes the input words by feeding them into its multi-layer transformer encoder. This encoder uses a multi-head self-attention mechanism, enabling each word to “look” at all other words in the sentence to understand their deep contextual relationships. The output is a sequence of contextualized hidden states, where each state is a rich representation of its corresponding input word within the context of the entire sequence.

• Sequence Tagging Layer

After SentBERT processes the word contexts, a special “tagging” layer is added on top. This layer takes SentBERT’s output for each word and essentially “votes” on what kind of tag it should get (like “beginning of aspect (B-ASP),” “inside aspect

(I-ASP),” or “other O “). It calculates a probability for each possible tag, and the tag with the highest probability is chosen as the final label for that word. A linear layer followed by softmax is added to project each hidden state into label scores. The formula is shown in (2).

$$P(l_j|h_i) = \text{softmax}(Wh_i + b) \quad (2)$$

The variable W represents the weight matrix, a set of adjustable values that the model learns to combine information effectively. b is the bias term used to fine-tune the output. h_i refers to the hidden state (embedding) of token w_i , which represents the rich, contextual understanding of the word w_i that SentBERT has generated. $P(l_j|h_i)$ signifies the probability of assigning label l_j to token w_i , given its hidden state h_i . In simpler terms, it indicates how likely it is that this specific tag (l_j) should be given to this particular word (w_i), based on what the model knows about it (h_i). Finally, the label with the highest probability is selected for each token.

C. Graph Feature Fusion

The Graph Feature Fusion module plays a critical role in transforming each review sentence into a structured representation that captures the grammatical relationships between words. Once aspect terms are extracted, it becomes essential to understand how these terms relate to sentiment-bearing expressions within the sentence. To achieve this, the system utilizes dependency parsing, a syntactic analysis technique that identifies the grammatical structure of a sentence by determining which words are connected and the nature of their relationships—such as subject, object, or modifier.

In this process, each word in the sentence is treated as a node, and the syntactic dependencies between words are represented as directed edges connecting those nodes. For example, in the sentence “The battery life is amazing,” the parser may determine that “battery” modifies “life,” “life” is the subject of the verb “is,” and “amazing” is the complement of “is.” These relationships form a sentence-level dependency graph that preserves both the grammatical and semantic structure of the input.

The resulting sentence-level graph is fed into a Graph Convolutional Network (GCN), enabling the model to analyze sentiment propagation across the sentence structure. To enhance learning, each GCN layer applies the ReLU (Rectified Linear Unit) activation function, defined as $\text{ReLU}(x) = \max(0, x)$, which introduces non-linearity and helps the network focus on meaningful features [20].

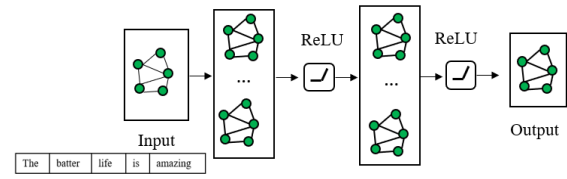


Fig. 3. Graph Feature Fusion

As shown in Fig. 3, this module visually maps syntactic connections between words, forming a graph that enables the model to relate opinions (e.g., “amazing”) to aspects (e.g., “battery life”) more effectively. This approach strengthens aspect-based sentiment classification by preserving grammatical context.

D. Fusion Layer

The Fusion Layer Module (Concatenate SentBERT+ GFF) serves as a critical integration point within the system, combining the rich semantic understanding from SentBERT with the structural grammatical insights from the Graph Feature Fusion (GFF) module. This module operates by concatenating the output representations from both SentBERT, which is responsible for aspect term extraction, and the Graph Feature Fusion module. The concatenation process creates a more comprehensive and robust feature vector for each data point or word. This unified representation is designed to capture both the deep contextual meaning of the text and the explicit relationships between words, as identified by the dependency parser. By bringing together these two distinct yet complementary types of information, the Fusion Layer provides richer input to subsequent stages, such as Contrastive Learning, enabling the model to achieve a more nuanced and accurate understanding of sentiment within the context of specific aspects of product reviews.

To illustrate the effect of the Fusion Layer, consider the review sentence “The battery life is excellent, but the camera is disappointing.” SentBERT generates contextual semantic embeddings that capture the positive polarity of “excellent” and the negative polarity of “disappointing,” while the GFF module encodes syntactic dependencies that link these sentiment-bearing words to their corresponding aspects (“battery life” and “camera”). After concatenation, the resulting fused representation jointly preserves sentiment polarity, aspect context, and grammatical structure within a single feature vector, enabling downstream modules to distinguish aspect-level sentiments more effectively than using semantic or syntactic features alone.

E. Contrastive Learning

The Contrastive Learning Module is designed to improve the model's ability to learn more discriminative and robust feature representations for sentiment classification. After extracting semantic features using SentBERT and structural features via Graph Feature Fusion (GFF), the concatenated embeddings are passed into this module, which applies a supervised contrastive loss. The objective is to pull embeddings of reviews with the same sentiment label closer together in the feature space, while pushing those with different sentiment labels apart. This technique enhances the separability of sentiment classes—positive, neutral, and negative—especially when reviews contain ambiguous or conflicting expressions. By reinforcing class-specific similarity and inter-class distinction, the Contrastive Learning Module helps the classifier make more accurate predictions and generalize better to unseen data.

The Supervised Contrastive Loss [21] is used to train the model by comparing the similarity of samples in a batch of size N . The overall loss for the batch is calculated by summing the individual losses for each sample as defined in (3).

$$L_{batch} = \sum_{i=1}^N L_i \quad (3)$$

Here, L_i represents the loss for the i -th sample, also called the anchor sample. The individual loss L_i is calculated as defined in (4).

$$L_i = \sum_{j \in P(i)} -\log \left[\frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k \in A(i)} \exp(\text{sim}(z_i, z_k)/\tau)} \right] \quad (4)$$

Where z_i represent the embedding (vector representation) of the i -th sample. y_i is its ground-truth sentiment label of the i -th sample (e.g., positive, neutral, negative). The function $\text{sim}(u, v)$ measures the similarity between two embedding vectors u and v . Typically, cosine similarity is used, defined as in (5).

$$\text{sim}(u, v) = \frac{u \cdot v}{\|u\| \cdot \|v\|} \quad (5)$$

The parameter $\tau > 0$ is a temperature scalar that scales the similarity scores. Smaller τ values make the model more sensitive to differences between samples. $P(i)$ denotes the set of positive samples. $A(i)$ represents the set of all other samples in the batch.

F. SHAP Explainable

The SHAP Explainable Module enhances the interpretability of the model's sentiment predictions by attributing importance scores to individual input

tokens. After the sentiment classifier generates an output—whether positive, neutral, or negative—this module applies SHAP (Shapley Additive Explanations) [22] to analyze how much each word in the product review contributed to the final prediction. Grounded in game theory, SHAP calculates the marginal contribution of each word by comparing the model's predictions with and without that word across various subsets. This results in a ranked list of tokens, each associated with a SHAP value indicating its positive or negative influence. The output is then visualized using color-coded summary plots, heatmaps, or bar charts, providing users with transparent and intuitive explanations. By incorporating this module, the framework supports explainable AI, allowing stakeholders to better understand the rationale behind each sentiment decision, which is especially valuable in customer feedback analysis and decision support systems.

For example, in the review “The battery life is terrible, but the camera is great,” SHAP may assign strong negative importance to “terrible” and strong positive importance to “great,” revealing their opposing effects on sentiment prediction. These attributions are visualized using heatmaps or bar plots to help users understand which terms drive the model's decisions. This module enhances model transparency, enabling better trust, error analysis, and actionable insights from product review data. The predictive formula constructed by the SHAP algorithm [22] is illustrated in (6).

$$y_i = y_{base} + y(x_i^1) + y(x_i^2) + \dots + y(x_i^j) \quad (6)$$

Where y_i is the i^{th} predicted value, y_{base} is the base-line value, while y_j is the SHAP value of the j^{th} feature in the i^{th} sample.

G. Implementation Details

The proposed framework was implemented in Python 3.7 and executed on an NVIDIA GeForce RTX 4060 GPU. Semantic features were extracted using a fine-tuned SentBERT (BERT-base, 12 layers, 768 hidden units), trained for aspect term extraction with a BIO tagging scheme using a learning rate of $2e-5$ and a batch size of 32. Syntactic features were obtained via dependency parsing, where each sentence was represented as a graph and processed using a layer Graph Convolutional Network with ReLU activation and 128 hidden units. The semantic and syntactic embeddings were concatenated in a fusion layer and optimized using supervised contrastive learning with cosine similarity and a temperature parameter of 0.07. Final sentiment classification was performed using a softmax layer, and SHAP was applied to provide word-level interpretability of the model's predictions.

IV. EXPERIMENTAL RESULTS

This section presents the experimental setup and results for the proposed hybrid sentiment analysis framework, designed to evaluate product reviews on Amazon. The experiments aimed to assess the model's ability to classify customer sentiment across multiple mobile phone brands using key performance metrics, including accuracy, precision, recall, and F1 Score. Results were compared against various baseline machine learning and deep learning models to demonstrate the effectiveness of the proposed method.

A. Data Collection

This study used a publicly available dataset from Amazon to perform sentiment analysis (SA). This dataset contains 10,500 customer reviews across eight mobile phone brands: Apple, Samsung, Motorola, OnePlus, Xiaomi, Google, LG, and Huawei. It comprises 40 models, including the Apple iPhone 15, Samsung Galaxy S24 Ultra, and Google Pixel 8. The dataset includes several attributes, such as brand, model, star rating, and review text, forming a robust basis for comprehensive analysis in this research. The dataset can be accessed via the following link: <https://rb.gy/rnrv3e>

B. Experimental Evaluation

The dataset was split into an 80% training set and a 20% test set. Baseline models—including Naïve Bayes, Decision Tree, Logistic Regression, Random Forest, Support Vector Machine, BERT-only, and GCN-only—were implemented using standard configurations, and deep learning baselines were fine-tuned under the same training and evaluation conditions to ensure fair comparison. Model performance was evaluated using accuracy, F1-score, recall, and precision. Accuracy measures the overall correctness of a model's predictions. High accuracy indicates that the model makes correct predictions most of the time. F1-score provides a balanced measure of precision and recall. A high F1-score indicates that the model is achieving a good balance between precision and recall. Precision measures the model's ability to correctly identify only positive instances. High precision means that when the model predicts a positive instance, it is likely to be correct. Recall measures the model's ability to correctly identify all positive instances. A high recall means the model can identify most positive instances. These formulas are shown in (7)-(10).

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

$$\text{F1-Score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (8)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (9)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (10)$$

The performance comparison shown in Table I demonstrates that the proposed method significantly outperforms traditional and deep learning-based baselines. Models such as Decision Tree, Naive Bayes, Random Forest, Support Vector Machine, Logistic Regression, GCN-only, and BERT-only yielded lower performance scores. In contrast, the proposed method achieved 97% in accuracy, precision, recall, and F1-score, reflecting superior sentiment classification capability. These results confirm that integrating SentBERT, Graph Feature Fusion, contrastive learning, and SHAP-based explainability yields a more robust and interpretable sentiment analysis model.

TABLE I
PERFORMANCE COMPARISON OF THE PROPOSED
METHOD WITH OTHER APPROACHES

Models	Accuracy	Precision	Recall	F1-Score
Decision Tree	0.84	0.84	0.84	0.84
Naive Bayes	0.84	0.85	0.84	0.81
Random Forest	0.88	0.89	0.88	0.87
GCN-only	0.88	0.88	0.88	0.88
Support Vector Machine	0.89	0.89	0.89	0.89
Logistic Regression	0.91	0.91	0.91	0.91
BERT-only	0.93	0.93	0.93	0.93
Proposed Method	0.97	0.97	0.97	0.97

Fig. 4 and Fig. 5 illustrate the output of the SHAP Explainable Module, which visualizes the importance of different product attributes across sentiment classes.

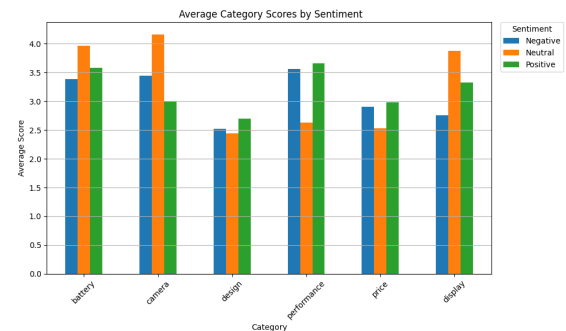


Fig. 4. Average Aspect Scores by Sentiment Category

In Fig. 4, a grouped bar chart shows the average sentiment scores—positive, neutral, and negative—for six key aspects of mobile phones: battery, camera, design, performance, price, and display. This visualization highlights how specific features are associated with different emotional polarities. For instance, camera and display tend to receive higher scores in neutral reviews, while design shows relatively low scores across sentiments, indicating common user dissatisfaction or indifference.

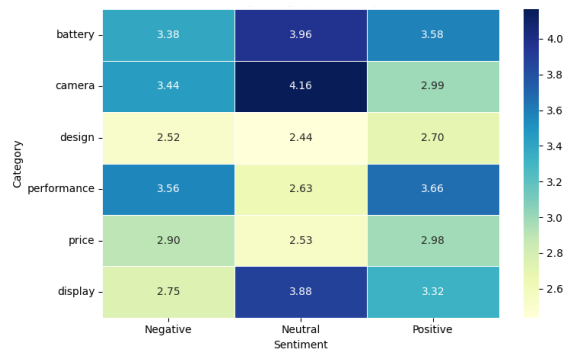


Fig. 5. Heatmap of average aspect scores across sentiment categories

Complementing this, Fig. 5 presents a heatmap showing the same average category scores but in a more intuitive, color-coded format. The deeper the color intensity, the higher the average sentiment score. This allows users and analysts to quickly identify which aspects significantly influence sentiment predictions and how those influences vary across different categories. Together, these visualizations provide interpretable insights into how feature-level contributions affect the model's decisions, thereby supporting transparency and trust in sentiment analysis outcomes.

V. DISCUSSION AND CONCLUSION

This study successfully developed and evaluated a novel hybrid framework for sentiment analysis of mobile phone reviews, effectively integrating semantic embeddings from SentBERT with syntactic structures extracted through Graph Feature Fusion (GFF). The proposed method demonstrated a significant performance advantage over traditional machine learning and deep learning baselines, consistently achieving 97% across key evaluation metrics—accuracy, precision, recall, and F1-score. This superior outcome highlights the effectiveness of combining diverse linguistic representations for robust sentiment classification.

The achieved performance can be attributed to several synergistic architectural components. Firstly, the SentBERT module's capability for Aspect Term Extraction (ATE) was instrumental in enabling a fine-grained understanding of product-specific sentiments.

This aspect-level granularity is crucial in analyzing e-commerce reviews, which often contain complex expressions or conflicting opinions about different features. Secondly, the Graph Feature Fusion (GFF) module leveraged dependency parsing to capture the intricate grammatical relationships and syntactic dependencies between words. This structural understanding enabled the model to effectively map sentiment-bearing terms to specific product aspects, a common challenge for methods that rely solely on sequential text representations. The subsequent Fusion Layer then seamlessly integrated these rich semantic and structural features, forming a comprehensive input for the classifier. Furthermore, the application of supervised contrastive learning significantly enhanced the model's discriminative power by refining the feature space and promoting greater separability among positive, neutral, and negative sentiment classes. Finally, the integration of SHAP Explainable AI provided critical interpretability, offering transparent insights into which words and aspects most significantly influenced the sentiment predictions, as visually corroborated by Fig. 4 and Fig. 5. This transparency not only fosters user trust but also provides actionable insights for businesses. This framework holds significant potential for real-world applications in e-commerce, offering enhanced automated review analysis capabilities, deeper insights for customer service, and valuable support for product development strategies. Despite its demonstrated high performance and interpretability, the proposed framework is not without limitations. Its dependence on accurate dependency parsing renders it susceptible to grammatical errors or informal language prevalent in user-generated content, potentially leading to inaccuracies in graph construction. Additionally, the computational demands of combining SentBERT, GCN-based processing and supervised contrastive learning are substantial, potentially limiting real-time deployment in large-scale applications. Lastly, while highly effective for mobile phone reviews, the model's domain specificity suggests that its generalizability to other diverse domains (e.g., hospitality, software reviews) may require further adaptation and retraining. Future research will therefore focus on optimizing model efficiency to support real-time sentiment analysis by reducing inference latency and computational cost. Cross-domain and cross-platform evaluations will be conducted to assess generalizability and robustness across diverse review types.

Further extensions may include multilingual sentiment analysis, dynamic and adaptive aspect extraction for emerging product features, and lightweight graph representations to improve scalability. Incorporating online or continuous learning mechanisms to adapt to evolving language

use in customer reviews is also a promising direction for future work.

Funding: This research was supported by Asia-Pacific International University, Thailand, under grant number: RRDC 2024-222.

B Econ in Business Economics. His research interests focus on business administration, management, marketing, and finance.

REFERENCE

- [1] K. Qiu and L. Zhang, "How online reviews affect purchase intention: A meta-analysis across contextual and cultural factors," *Data Inf. Manag.*, vol. 8, no. 2, p. 100058, Jun. 2024.
- [2] V. Shrirame, J. Sabade, H. Soneta, and M. Vijayalakshmi, "Consumer behavior analytics using machine learning algorithms," in *Proc. 2020 IEEE Int. Conf. Electron., Comput. Commun. Technol. (CONECCT)*, 2020, pp. 1-6, <https://doi.org/10.1109/CONECCT50063.2020.9198562>.
- [3] T. Brown et al., "Language models are few-shot learners," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2020, pp. 1877-1901.
- [4] J. J. Rahman, T. M. A. Riaz, M. Partha, K. M. Mohsin, N. Kamruddin, and M. F. Mridha, "Recent advancements and challenges of NLP-based sentiment analysis: A state-of-the-art review," *Nat. Lang. Process. J.*, p. 100059, Mar. 2024.
- [5] C. Dewi, R.-C. Chen, H. J. Christanto, and F. Cauteruccio, "Multinomial naïve Bayes classifier for sentiment analysis of internet movie database," *Vietnam J. Comput. Sci.*, vol. 10, no. 4, pp. 485-498, Jan. 2023, <https://doi.org/10.1142/S2196888823500100>.
- [6] I. A. K. Shaikh, R. P. Pujar, S. P. Kishore, S. Ragamayi, P. V. Krishna, and A. B. Nadaf, "A novel approach for e-commerce system for sale prediction with denoised autoencoder and SVM-based approach," in *Proc. 2023 Int. Conf. Sustain. Comput. Smart Syst. (ICSCSS)*, 2023, pp. 1684-1689.
- [7] N. Karina and Sriani, "Sentiment analysis of skincare products using the naïve Bayes method," *J. Inf. Syst. Informatics*, vol. 6, no. 3, pp. 1663-1676, Sep. 2024, <https://doi.org/10.51519/journalisi.v6i3.817>.
- [8] T. A. Khan, R. Sadiq, Z. Shahid, M. M. Alam, and M. B. Mohd Su'ud, "Sentiment analysis using support vector machine and random forest," *J. Informatics Web Eng.*, vol. 3, no. 1, pp. 67-75, Feb. 2024, <https://doi.org/10.33093/jiwe.2024.3.1.5>.
- [9] J. Devlin, M. -W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, 2019, pp. 4171-4186.
- [10] A. Vaswani et al., "Attention is all you need," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 5998-6008.
- [11] C. Li, D. Cheng, and Y. Morimoto, "Leveraging deep neural networks for aspect-based sentiment classification," *arXiv*, pp. 1-19, Mar. 2025, <https://doi.org/10.48550/arXiv.2503.12803>.
- [12] J. Hu and Y. Li, "Aspect-level sentiment analysis of bi-graph convolutional networks based on enhanced syntactic structural information," *J. Comput. Commun.*, vol. 13, no. 1, pp. 72-89, Jul. 2025, <https://doi.org/10.4236/jcc.2025.131006>.
- [13] M. Peal, M. S. Hossain, and J. Chen, "Summarizing consumer reviews," *J. Intell. Inf. Syst.*, vol. 59, no. 1, pp. 193-212, Feb. 2022, <https://doi.org/10.1007/s10844-022-00694-9>.
- [14] K. S. Kumar, M. Priyanka, M. Rishitha, D. D. Teja, and N. Madhuri, "Text summarization with sentiment analysis," *Int. J. Innov. Res. Comput. Sci. Technol.*, vol. 9, no. 4, Jul. 2021.
- [15] L. Yang, Y. Li, J. Wang, and R. S. Sherratt, "Sentiment analysis for e-commerce product reviews in Chinese based on sentiment lexicon and deep learning," *IEEE Access*, vol. 8, pp. 23522-23530, Jan. 2020.
- [16] A. R. Sulthana, K. Jaithunbi, H. Harikrishnan, and V. Vijayakumar, "Sentiment analysis on movie reviews dataset using support vector machines and ensemble learning," *Int. J. Inf. Technol. Web Eng.*, vol. 17, pp. 1-23, Jan. 2022.
- [17] D. Šandor and M. Bagić Babac, "Sarcasm detection in online comments using machine learning," *Inf. Discov. Deliv.*, vol. 52, no. 2, pp. 213-226, Mar. 2024, <https://doi.org/10.1108/IDD-01-2023-0002>.
- [18] E. Wahyudi and R. Kusumaningrum, "Aspect-based sentiment analysis in e-commerce user reviews using latent Dirichlet allocation (LDA) and sentiment lexicon," in *Proc. 3rd Int. Conf. Informatics Comput. Sci. (ICICoS)*, 2019.
- [19] L. Luo, C. -H. Wei, P. -T. Lai, R. Leaman, Q. Chen, and Z. Lu, "AIONER: All-in-one scheme-based biomedical named entity recognition using deep learning," *Bioinformatics*, vol. 39, no. 5, pp. 1-9, May 2023, <https://doi.org/10.1093/bioinformatics/btad310>.
- [20] M. A. Sejan, M. H. Rahman, M. A. Aziz, J. -I. Baik, Y. -H. You, and H. -K. Song, "Graph convolutional network design for node classification accuracy improvement," *Mathematics*, vol. 11, no. 17, p. 3680, Aug. 2023.
- [21] P. Khosla et al., "Supervised contrastive learning," *arXiv*, 2020. [Online]. Available: <https://arxiv.org/abs/2004.11362> [Accessed: Jun. 8, 2025].
- [22] X. Zhao, P. -F. Zhang, Q. Zhao, D. Zhang, Y. Tuerxunmaiti, and H. Cao, "ASHAP algorithm-based prediction of the interlaminar shear strength degradation of G/BFRP bars embedded in concrete exposed to marine environment," *Case Stud. Constr. Mater.*, vol. 22, pp. 1-19, Jul. 2025.



Damrong Sattayawaksakul is an Assistant Professor at the Faculty of Business, Asia-Pacific International University, Thailand. He holds a PhD in Business Administration. He obtained an MBA in management, a BBA in Finance and Banking, and another BEcon in Business Economics. His research interests focus on business administration, management, marketing, and finance.



Pimpa Cheewaprakobkit is an Assistant Professor at the Faculty of Information Technology, Asia-Pacific International University, Thailand. She holds a Ph.D. in Computer Science and Information Engineering from National Central University, Taiwan. She also earned an M.S. in Computer Science and Information Systems and a B.Sc. in Computer Science. Her research interests include computer vision, object tracking, big data analysis, deep learning, object detection, and object recognition.

PAPER FORMAT (IEEE Style)

I. FORMAT

- Your paper must use a paper size corresponding to A 4 which is 210 mm (8.27 inch) Wide and 297 mm (11.69 inch)
- Your paper must be in two column format
- Articles not more than 15 pages in length, single-sided A4 paper, margins (top, bottom, left, right) are 1 inch (2.54 cm)
- Abstract and References and content set to double columns,
- English font is Times New Roman, as follows:

TABLE I
FONT SIZES FOR PAPERS

Content	Font Size	Labelling
Title (Single column)	18 (CT)	bold
Authors (Single column)	11 (CT)	bold
Authors Information (Single column)	10 (CT)	regular
Abstract	10 (LRJ)	bold
Index Terms (Keywords)	10 (LRJ)	bold
Content	10 (LRJ)	regular
Heading1	10 (CT)	regular (Capitalization)
Heading 2	10 (LJ)	regular
Table Title (Place above the Table)	8 (CT)	regular
Table content	8 (CT)	regular
Figure caption (Place below the figure)	8 (LJ)	regular
Reference Head	10 (CT)	regular (Capitalization)
Reference	8 (LJ)	regular
Author Profiles	10 (LRJ)	bold author name/ profile regular

CT=Centre Text, LJ=Left Justified, RJ=Right Justified, LRJ=Left & Right Justified

II. COMPOSITION OF THE ARTICLE

A. Article title

B. *Authors information*, Write (all) the author's name, affiliation, department, city, country and E-mail (set to Single Column) all.

C. *Abstract*, Must be under 200 words and not include subheadings or citations. Define all symbols used in the abstract. Do not delete the blank line immediately above the abstract.

D. *Index Terms*, Enter key words or phrases in alphabetical order, separated by commas.

E. Content

1) *Academic article*, should include: Introduction, Content, and Conclusion.

2) *Research article*, should include: Introduction, Literature review, Materials Methods, Results, Discussion, and Conclusion.

Clearly summarize the important findings of the paper. It should contain such as objectives, methods and major results.

F. Introduction

The Introduction section of reference text expands on the background of the work (some overlap with the Abstract is acceptable). The introduction should not include subheadings.

G. *Pictures, table, etc.*, Must be use in numerical order in the article, provided the source correctly, cannot use other people's copyright.

Chart should be colored contrastingly or in black and white.

H. Reference

1) *Cited in the main text*. Indicate the number in the [] mark at the end of the text or the name of the referring person. Let the numbers be in the same line of content as [1].

2) *Cited after the article*. Put all bibliographical reference after articles, and order according to the author's name, please refer IEEE format. The footer reference format is as follows.

III. REFERENCES

References in research articles and scholarly articles. For academic and research journals, INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET). The technology defines referrals according to the IEEE format. All references should be listed at the end of the paper using the following.

Basic format for books:

J. K. Author, "Title of chapter in the book," in *Title of His Published Book*, xth ed. City of Publisher, Country if not USA: Abbrev. of Publisher, year, ch. x, sec. x, pp. xxx-xxx.

Examples:

- [1] G. O. Young, "Synthetic structure of industrial plastics," in *Plastics*, 2nd ed., vol. 3, J. Peters, Ed. New York: McGraw-Hill, 1964, pp. 15-64.
- [2] W.-K. Chen, *Linear Networks and Systems*. Belmont, CA: Wadsworth, 1993, pp. 123-135.

Basic format for periodicals:

J. K. Author, "Name of paper," *Abbrev. Title of Periodical*, vol. x, no. x, pp. xxx-xxx, Abbrev. Month. year.

Examples:

- [3] J. U. Duncombe, "Infrared navigation—Part I: An assessment of feasibility," *IEEE Trans. Electron Devices*, vol. ED-11, no. 1, pp. 34-39, Jan. 1959.
- [4] E. P. Wigner, "Theory of traveling-wave optical laser," *Phys. Rev.*, vol. 134, pp. A635-A646, Dec. 1965.
- [5] E. H. Miller, "A note on reflector arrays," *IEEE Trans. Antennas Propagat.*, to be published.

Basic format for reports:

J. K. Author, "Title of report," Abbrev. Name of Co., City of Co., Abbrev. State, Rep. xxx, year.

Examples:

- [6] E. E. Reber, R. L. Michell, and C. J. Carter, "Oxygen absorption in the earth's atmosphere," Aerospace Corp., Los Angeles, CA, Tech. Rep. TR-0200 (4230-46)-3, Nov. 1988.
- [7] J. H. Davis and J. R. Cogdell, "Calibration program for the 16-foot antenna," Elect. Eng. Res. Lab., Univ. Texas, Austin, Tech. Memo. NGL-006-69-3, Nov. 15, 1987.

Basic format for handbooks:

Name of Manual/Handbook, x ed., Abbrev. Name of Co., City of Co., Abbrev. State, year, pp. xxx-xxx.

Examples:

- [8] *Transmission Systems for Communications*, 3rd ed., Western Electric Co., Winston-Salem, NC, 1985, pp. 44-60.
- [9] *Motorola Semiconductor Data Manual*, Motorola Semiconductor Products Inc., Phoenix, AZ, 1989.

Basic format for books (when available online):

Author. (year, month day). Title. (edition) [Type of medium]. volume (issue). Available: site/path/file

Example:

- [10] J. Jones. (1991, May 10). *Networks*. (2nd ed.) [Online]. Available: <http://www.atm.com>

Basic format for journals (when available online):

Author. (year, month). Title. *Journal*. [Type of medium]. volume (issue), pages. Available: site/path/file

Example:

- [11] R. J. Vidmar. (1992, Aug.). On the use of atmospheric plasmas as electromagnetic reflectors. *IEEE Trans. Plasma Sci.* [Online]. 21(3), pp. 876-880. Available: <http://www.halcyon.com/pub/journals/21ps03-vidmar>

Basic format for papers presented at conferences (when available online):

Author. (year, month). Title. Presented at Conference title. [Type of Medium]. Available: site/path/file

Example:

- [12] PROCESS Corp., MA. Intranets: Internet technologies deployed behind the firewall for corporate productivity. Presented at INET96 Annual Meeting. [Online]. Available: <http://home.process.com/Intranets/wp2.htm>

Basic format for reports and handbooks (when available online):

Author. (year, month). Title. Comp any . City, State or Country. [Type of Medium]. Available: site/path/file

Example:

- [13] S. L. Talleen. (1996, Apr.). The Intranet Architecture: Managing information in the new paradigm. Amdahl Corp., CA. [Online]. Available: <http://www.amdahl.com/doc/products/bsg/intra/infra/html>

Basic format for computer programs and electronic documents (when available online):

ISO recommends that capitalization follow the accepted practice for the language or script in which the information is given.

Example:

- [14] A. Harriman. (1993, June). Compendium of genealogical software. *Humanist*. [Online]. Available e-mail: HUMANIST@NYVM.ORG Message: get GENEALOGY REPORT

Basic format for patents (when available online):

Name of the invention, by inventor's name. (year, month day). Patent Number [Type of medium]. Available: site/path/file

Example:

- [15] Musical toothbrush with adjustable neck and mirror, by L.M.R. Brooks. (1992, May 19). Patent D 326 189 [Online]. Available: NEXIS Library: LEXPAT File: DESIGN

Basic format for conference proceedings (published):

J. K. Author, "Title of paper," in *Abbreviated Name of Conf.*, City of Conf., Abbrev. State (if given), year, pp. xxxxxx.

Example:

- [16] D. B. Payne and J. R. Stern, "Wavelength-switched passively coupled single-mode optical network," in *Proc. IOOC-ECOC*, 1985, pp. 585-590.

Example for papers presented at conferences (unpublished):

- [17] D. Ebehard and E. Voges, "Digital single sideband detection for interferometric sensors," presented at the 2nd Int. Conf. Optical Fiber Sensors, Stuttgart, Germany, Jan. 2-5, 1984.

Basic format for patents:

J. K. Author, "Title of patent," U.S. Patent x xxx xxx, Abbrev. Month. day, year.

Example:

- [18] G. Brandli and M. Dick, "Alternating current fed power supply," U.S. Patent 4 084 217, Nov. 4, 1978.

Basic format for theses (M.S.) and dissertations (Ph.D.):

J. K. Author, "Title of thesis," M.S. thesis, Abbrev. Dept., Abbrev. Univ., City of Univ., Abbrev. State, year.

J. K. Author, "Title of dissertation," Ph.D. dissertation, Abbrev. Dept., Abbrev. Univ., City of Univ., Abbrev. State, year.

Examples:

- [19] J. O. Williams, "Narrow-band analyzer," Ph.D. dissertation, Dept. Elect. Eng., Harvard Univ., Cambridge, MA, 1993.
- [20] N. Kawasaki, "Parametric study of thermal and chemical nonequilibrium nozzle flow," M.S. thesis, Dept. Electron. Eng., Osaka Univ., Osaka, Japan, 1993.

Basic format for the most common types of unpublished references:

J. K. Author, private communication, Abbrev. Month, year.

J. K. Author, "Title of paper," unpublished.

J. K. Author, "Title of paper," to be published.

Examples:

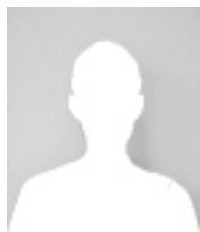
- [21] A. Harrison, private communication, May 1995.
- [22] B. Smith, "An approach to graphs of linear forms," unpublished.
- [23] A. Brahms, "Representation error for real numbers in binary computer arithmetic," IEEE Computer Group Repository, Paper R-67-85.

Basic format for standards:

Title of Standard, Standard number, date.

Examples:

- [24] IEEE Criteria for Class IE Electric Systems, IEEE Standard 308, 1969.
- [25] Letter Symbols for Quantities, ANSI Standard Y10.5-1968.

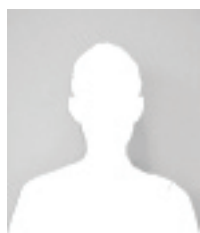


First A. Author and the other authors may include biographies at the end of regular papers. Biographies are often not included in conference related papers. The first paragraph may contain a place and/or date of birth (list place, then date).

Next, the author's educational background is listed. The degrees should be listed with type of degree in what field, which institution, city, state, and country, and year the degree was earned. The author's major field of study should be lower-cased.

The second paragraph uses the pronoun of the person (he or she) and not the author's last name. It lists military and work experience, including summer and fellowship jobs. Job titles are capitalized. The current job must have a location; previous positions may be listed without one. Information concerning previous publications may be included. Try not to list more than three books or published articles. The format for listing publishers of a book within the biography is: title of book (city, state: publisher name, year) similar to a reference. Current and previous research interests end the paragraph.

The third paragraph begins with the author's title and last name (e.g., Dr. Smith, Prof. Jones, Mr. Kajor, Ms. Hunter). List any memberships in professional societies. Finally, list any awards and work for committees and publications. If a photograph is provided, the biography will be indented around it. The photograph is placed at the top left of the biography, and should be of good quality, professional-looking, and black and white (see above example). Personal hobbies will be deleted from the biography. Following are two examples of an author's biography.

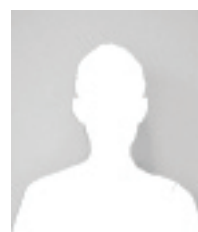


Second B. Author was born in Greenwich Village, New York City, in 1977. He received the B.S. and M.S. degrees in aerospace engineering from the University of Virginia, Charlottesville, in 2001 and the Ph.D. degree in mechanical engineering from Drexel

University, Philadelphia, PA, in 2008. From 2001 to 2004, he was a Research Assistant with the Princeton Plasma Physics Laboratory. Since 2009, he has been an

Assistant Professor with the Mechanical Engineering Department, Texas A&M University, College Station. He is the author of three books, more than 150 articles, and more than 70 inventions. His research interests include high-pressure and high-density nonthermal plasma discharge processes and applications, microscale plasma discharges, discharges in liquids, spectroscopic diagnostics, plasma propulsion, and innovation plasma applications. He is an Associate Editor of the journal *Earth, Moon, Planets*, and holds two patents.

Mr. Author was a recipient of the International Association of Geomagnetism and Aeronomy Young Scientist Award for Excellence in 2008, the IEEE Electromagnetic Compatibility Society Best Symposium Paper Award in 2011, and the American Geophysical Union Outstanding Student Paper Award in Fall 2005.



Third C. Author received the B.S. degree in mechanical engineering from National Chung Cheng University, Chiayi, Taiwan, in 2004 and the M.S. degree in mechanical engineering from National Tsing Hua University, Hsinchu, Taiwan, in 2006. He is currently

pursuing the Ph.D. degree in mechanical engineering at Texas A&M University, College Station.

From 2008 to 2009, he was a Research Assistant with the Institute of Physics, Academia Sinica, Tapei, Taiwan. His research interest includes the development of surface processing and biological/medical treatment techniques using nonthermal atmospheric pressure plasmas, fundamental study of plasma sources, and fabrication of micro- or nanostructured surfaces.

Mr. Author's awards and honors include the Frew Fellowship (Australian Academy of Science), the I. I. Rabi Prize (APS), the European Frequency and Time Forum Award, the Carl Zeiss Research Award, the William F. Meggers Award and the Adolph Lomb Medal (OSA).

Remark: More detail information, Please read Preparation of Papers for INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET), <https://ph02.tci-thaijo.org/index.php/isjet/index>



Panyapiwat Institute of Management (PIM)
85/1 Moo 2, Chaengwattana Rd,
Bang Talat, Pakkred, Nonthaburi 11120, Thailand
Tel. +66 2855 1560
<https://www.tci-thaijo.org/index.php/isjet/index>
<https://isjet.pim.ac.th>
E-mail: isjet@pim.ac.th