ISJET

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY

Volume 8 No. 2 July-December 2024

ISSN 2586-8527 (Online)

Panyapiwat Institute of Management

Indexed in the Thai-Journal Citation Index (TCI 2)

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET)

Volume 8 No. 2 July-December 2024

ISSN 2586-8527 (Online)
PANYAPIWAT INSTITUTE OF MANAGEMENT

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET)
Volume 8 No. 2 July-December 2024
ISSN 2586-8527 (Online)

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

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET)

Volume 8 No. 2 July-December 2024 ISSN 2586-8527 (Online)

Objective:
International Scientific Journal of Engineering and Technology will be dedicated to serving as a forum to share knowledge on research advances in all fields of sciences: Engineering, Technology, Innovation, Information Technology, Management Information Systems, Logistics and Transportation, Agricultural Science and Technology, Animal Science and Aquaculture, Food Science, and other areas in Sciences and Technology. Submissions are welcomed from both PIM as well as other Thai and foreign institutions.

Scope:
Engineering, Technology, Innovation, Information Technology, Management Information Systems, Logistics and Transportation, Agricultural Science and Technology, Animal Science and Aquaculture, Food Science, and other areas of Sciences and Technology.

Type of Article:
• Research article
• Academic article
• Book review
• Review article

Languages of academic works:
An article written in either English language is accepted for publication.

Reviewing Policy:
1. Any manuscript to be accepted for publication must have been reviewed and approved by at least three 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 each other.
2. The submitted manuscript must have never been published in any other periodical, and must not be in the approving process for publication by any other periodical. Also, the author must not plagiarize the work of other people.
3. The article, expression, illustrations, and tables that are published in the Journal are the sole responsibility of the author, and definitely not that of Panyapiwat Institute of Management.
4. The Editorial Board of International Scientific Journal of Engineering and Technology 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 International Scientific Journal of Engineering and Technology reserves the right to cancel the publication that has been issued a certification of publication in the Journal.
6. The Editorial Board of International Scientific Journal of Engineering and Technology reserves the right for decision making on publishing any article in the Journal.

Frequency of Publication:
Twice a year
• The first issue: January-June
• The second issue: July-December

Publication and Access Charges:
There are no charges to submit and publish all types of articles. Full articles in pdf format can be downloaded free from the journal website at https://ph02.tci-thaijo.org/index.php/isjet/index

ISJET Journal Editorial Board
The office of Research and Development
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

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET)
Volume 8 No. 2 July-December 2024
ISSN 2586-8527 (Online)

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET)
Volume 8 No. 2 July-December 2024
ISSN 2586-8527 (Online)

Peer Reviewers

Prof. Dr. Parames Chutima	Chulalongkorn University, Thailand
Assoc. Prof. Dr. Kiatfa Tangchaichit	Khon Kaen University, Thailand
Assoc. Prof. Dr. Nattapon Chantarapanich	Kasetsart University, Sriracha Campus, Thailand
Assoc. Prof. Dr. Nikorn Sirivongpaisal	Prince of Songkla University, Hat Yai Campus, Thailand
Assoc. Prof. Dr. Nivit Charoenchai	Chiang Mai University, Thailand
Assoc. Prof. Dr. Panomkhawn Riyamongkol	Naresuan University, Thailand
Assoc. Prof. Dr. Paritud Bhadhubanyong	Panyapiwat Institute of Management, Thailand
Assoc. Prof. Dr. Patomsok Wilaipon	Naresuan University, Thailand
Assoc. Prof. Dr. Ruengsak Kawtummachai	Charoen Pokphand Group, Thailand
Assoc. Prof. Dr. Sontisuk Teerachaichayu	King Mongkut's Institute of Technology Ladkrabang, Thailand
Assoc. Prof. Dr. Suchada Rianmora	Sirindhorn International Institute of Technology (SIIT), Thammasat University, Thailand
Assoc. Prof. Dr. Watcharin Po-ngaen	King Mongkut's University of Technology North Bangkok, Thailand
Asst. Prof. Dr. Adisak Joomwong	Maejo University. Chiang Mai, Thailand
Asst. Prof. Dr. Chatchawin Namman	Ubon Ratchathani University, Thailand
Asst. Prof. Dr. Chorkaew Jaturanonda	King Mongkut's University of Technology Thonburi, Thailand
Asst. Prof. Dr. Jaturaporn Rak-ngan	Naresuan University, Thailand
Asst. Prof. Dr. Kanyarat Lueangprasert	Burapha University Sakaeo Campus, Thailand
Asst. Prof. Dr. Narongdech Keeratipranon	Chulalongkorn University, Thailand
Asst. Prof. Dr. Pairoj Raothanachonkun	Burapha University, Thailand
Asst. Prof. Dr. Sompoche Pojjanapimol	Thammasat University, Thailand
Asst. Prof. Satien Janpla	Suan Sunandha Rajabhat University, Thailand
Dr. Kwankamon Dittakan	Prince of Songkla University, Thailand
Dr. Sanparith Marukatat	National Electronics and Computer Technology Center (NECTEC), Thailand
Dr. Thanawat Yochanang	King Mongkut's University of Technology North Bangkok, Thailand

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET)
Volume 8 No. 2 July-December 2024
ISSN 2586-8527 (Online)

Dear Colleagues,

Technology and Artificial Intelligence are playing an increasingly significant role in the workplace, such as the application of AI to reduce time and enhance efficiency. This issue features a diverse collection of articles that highlight the latest advancements in engineering.

The journal presents various articles on the application of AI, such as "A Transfer Learning-based Deep Convolutional Neural Network Approach for White Shrimp Abnormality Classification", which aims to develop a transfer learning-based deep convolutional neural network system capable of accurately categorizing shrimp. Another notable study is "Enhancing Warehouse Management with AI and Computer Vision: A Case Study in a Logistics Service Company", which focuses on the use of Artificial Intelligence (AI) and Computer Vision (CV) for inventory tracking and stock registration. Additionally, the article "The Development of Smart Farming System for Sea Lettuce Cultured Process" demonstrates how seaweed farming is being transformed into an intelligent system through IoT-based solutions.

Furthermore, the journal includes intriguing topics in Engineering and Technology, such as "The Morphology Associated With Harvesting Stage of Siam Red Ruby Pumelo (*Citrus grandis*)", "A Comparison between Profit and Economic Value Added Optimization to Design a Supply Chain Network: A Case Study of Food Supply Chain in Vietnam", and "A Literature Review of Steering Angle Prediction Algorithms for Autonomous Cars".

I am confident that this issue of ISJET will be of interest to a broad audience of researchers, engineers, and practitioners. I encourage you to read the articles carefully and consider submitting your research to future issues of the journal.

With kind regards,

Asst. Prof. Dr. Phannachet Na Lamphun
Associate Editor of Engineering and Technology
isjet@pim.ac.th

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND TECHNOLOGY (ISJET)
Volume 8 No. 2 July-December 2024
ISSN 2586-8527 (Online)

CONTENTS

A Comparison between Profit and Economic Value Added Optimization to Design a Supply Chain Network: A Case Study of Food Supply Chain in Vietnam

Tran Thi Uyen Linh[1], Navee Chiadamrong[2*], Nathridee Suppakitjarak[3], and Somrote Komolavanij[4]
[1,2]Logistics and Supply Chain Systems Engineering,
Sirindhorn International Institute of Technology, Thammasat University, Pathum Thani, Thailand
[3]Faculty of Commerce and Accountancy, Chulalongkorn University, Bangkok, Thailand
[4]Panyapiwat Institute of Management, Nonthaburi, Thailand
E-mail: ttulinh2000@gmail.com, navee@siit.tu.ac.th*, nathridee@acc.chula.ac.th, somrotekom@pim.ac.th

Received: April 26, 2024 / Revised: June 26, 2024 / Accepted: June 28, 2024

***Abstract*— The research field of food supply chain network design and optimization has expanded significantly. However, most recent studies have focused only on minimizing costs or maximizing profits, neglecting other important financial factors that affect the overall prosperity of the chain. This study generates two scenarios in the design of the supply chain network, comparing the maximization of profit and Economic Value Added (EVA) to assess their effectiveness in real-world situations. The comparison is based on supplier and potential distribution center selection, along with considerations of production level, production capacity, and the sizes of the plant, distribution centers, and retailers. The methodology considered in this research is based on Mixed-Integer Linear Programming (MILP) under deterministic parameters. The study provides computational results and managerial insights based on a case study of the food supply chain in Southern Vietnam. The findings indicate that the EVA maximization model offers a more precise evaluation of company wealth as compared to the profit maximization model as it can determine more suitable operating supply chain's decision variables leading to a significant decrease of 11.1% in the invested capital.**

***Index Terms*— Economic Value Added (EVA), Food Supply Chain Network Design (FSCND), Mixed Integer Linear Programming (MILP), Profit Maximization**

I. Introduction

The strategic design of the food supply chain network plays a pivotal role in addressing the fundamental needs of humanity, encompassing a diverse range of activities, including production, transportation, processing, distribution, and consumption [1]. However, real-world challenges, such as high logistics costs and limited investment funds to meet customer demand, hinder the practical implementation of the food supply chain [2].

The complexity of the food supply chain planning problem arises due to its vast scale, involving numerous interconnected echelons and variables that need careful consideration and strategic management [3]. For instance, the decision of selecting suitable suppliers or establishing distribution centers becomes increasingly intricate. In the field of food supply chain network design, challenges such as cost considerations, regulatory compliance, or transportation logistics may also necessitate adjustments to ordering quantities and facilities capacities or even the need for new centers in the case of much extra capacity being required. Burgess et al. [4] generated alternative food networks to enhance processes, meet customer needs, and elevate the overall quality of the supply chain. Gholian-Jouybari et al. [5] concentrated on designing a closed-loop agri-food supply chain network design for the soybean industry while Gholian-Jouybari et al. [6] developed policies in the agri-food supply chain coconut industry to provide fresh, healthy products to their societies.

Southern Vietnam, with its tropical climate and abundant agricultural products, holds great potential for strategically planning a food supply chain that can cater to the entire nation. This region plays a vital role as a key contributor to the economic market. According to Insights [7] on the food industry in Vietnam, the revenue surged from 64.86 billion USD in 2018 to 94.37 billion USD in 2023, as Fig. 1. The market is expected to grow annually by 7.46% (CAGR 2024-2028). To ensure adequate consumption, the Vietnamese government has prioritized the

promotion of the supply chain network. This includes a focus on effectively and sustainably managing all members while taking into account the economic value. Traditionally, the primary goal of planning in the food supply chain has been to maximize the profit or minimize total costs across the chain, aiming to design the most cost-effective strategy for the production, storage, and transportation of goods. It is essential to include the relevant costs in managing the supply chain, such as purchasing, transportation, holding, and production.

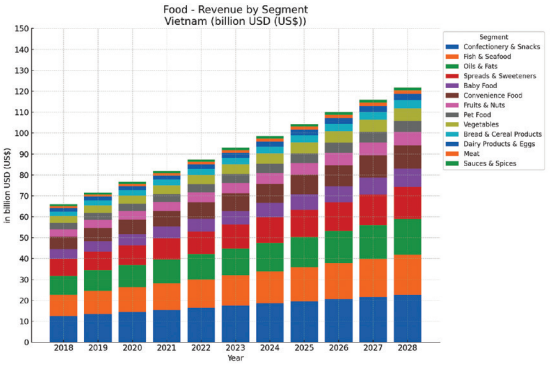


Fig. 1. Revenue of the food industry in Vietnam (2018-2028)

However, for a more practical and sustainable approach, it is important to evaluate the overall wealth of the company. In order to make decisions that involve such an investment, Economic Value Added (EVA) incorporates both operational and investment expenditures. It helps showcase the business performance based on the total costs invested, including both operating and fixed capital [3]. This financial resource allocation plays a vital role in the decision-making process of supply chain management [8]. Therefore, making the right decisions can save a business a significant amount of financial resources. For instance, instead of opening a new distribution center, a more cost-effective approach might involve allocating less amount of resources to expand the current distribution center. In some cases, the assessment and selection of potential facilities face limitations. For example, focusing solely on profit optimization may lead decision-makers to choose partners with low transportation expenses. However, a more detailed analysis could reveal that the cost savings in delivery would require more investment in the fixed assets. This is not, however, considered in the profit calculation. Therefore, this aspect deserves attention when framing a supply chain management problem and should be considered in the decision-making processes.

As a result, this study aims to suggest a shift in the perspective within the supply chain, particularly in the private food supply chain, moving from the traditional focus on maximizing profit to a more holistic approach that prioritizes economic income. Encouraging asset optimization, as highlighted by Economic Value Added (EVA), underscores how businesses can effectively use their assets to create value while simultaneously reducing capital costs [9]. The structure of this paper is set as follows. Section 2 presents the literature review. Section 3 introduces the methodology. In section 4, the mathematical models are formulated based on both scenarios (profit and EVA maximization). In section 5, a case study is presented to illustrate the methodology. Section 6 presents the results and discussion. Finally, section 7 summarizes the conclusion and outlines future studies.

II. Literature Review

A. Supply Chain Network Design and Optimization

Supply chain networks are complex structures that span the globe, encompassing a range of interconnected entities such as suppliers, production centers, distribution centers, retailers, and customers [8]. These entities engage in diverse activities, including raw material procurement, transportation, manufacturing, and product distribution to meet customer demands. The operations within these networks require careful consideration of several factors (strategic, tactical, operational), the type of product (single or multiple), and the studied periods (single or multiple). For instance, Nagurney [10] highlighted that the modeling framework developed in the study included many echelons (manufacturers, two-level distribution centers, and retailers), and the solution of the model yielded optimal product flows, capacity investments, and demand satisfaction with the minimum total costs. Kashanian and Ryan [11] proposed a sustainable supply chain network design for chemicals from biomass, which incorporated green electrochemistry to minimize annual costs with the three-echelon network (supplier, facility, and customer). Ala et al. [12] designed a blood collection and distribution network to optimize fixed and mobile facilities and supply points, considering the short-term and long-term aspects from donation places to hospitals through temporary and permanent centers. Nagurney [10] proposed a framework for the supply chain network design and redesign that minimized the total costs with two main factors determining the level of capacity in various nodes and operational flows, subject to customer satisfaction. Based on the aforementioned reviews, it was found that most discussions regarding the supply chain network design have focused mainly on maximizing the profit or minimizing the operational costs, without considering investment funds and the costs of capital. These funds could reflect the chain's capital, which constitutes authentic economic profitability.

B. Economic Value Added (EVA) as a Financial Metric for Decision-Making

EVA as present in Equation (1) is a performance metric that calculates the creation of shareholder value. It distinguishes itself from traditional financial performance metrics, such as net profit. EVA is the calculation of what profits remain after the costs of a company's capital are deducted from the operating profit [13].

$$EVA = NOPAT - (WACC \times IC) \quad (1)$$

Where:

- *NOPAT* is the net operating profit after tax.
- *WACC* is the weighted average cost of capital.
- *IC* is the invested capital.

Operating profit, obtained by subtracting Cost of Goods Sold (COGS), operating expenses, and depreciation from revenue, excludes interest and taxes. After-tax deductions, it becomes the Net Operating Profit After Tax (NOPAT), which represents the company's profit from core operations [5]. For investors, the Weighted Average Cost of Capital (WACC) is an important tool in assessing a company's potential for profitability. A lower WACC often indicates a robust business capable of securing capital from investors at a lower expense, whereas a higher WACC tends to signal riskier ventures necessitating higher returns to attract investors. Hence, within the complex framework of supply chain management, a nuanced comprehension of WACC emerges as crucial for guiding pivotal investment determinations, as proposed by Ashayeri and Lemmes [14]. Invested capital, in the context of EVA, refers to the total capital invested in a business to generate profits, usually calculated by the sum of all company-held assets.

In essence, EVA emerges as an indispensable tool, offering a nuanced and comprehensive framework for navigating the multifaceted landscape of corporate investments with precision and insight, offering benefits such as making more informed investment decisions considering the cost of capital, considering simultaneously the total expenses, and revenue, and facilitating decisions for long-term planning horizons. This EVA not only outweighs projected costs but also indicates the project's financial viability, and emphasizes parameters crucial to financial considerations such as capital investment and accounts payables/receivables. Longinidis and Georgiadis [9] presented an explanation of how physical network planning and financial formulation can be integrated to calculate the EVA. They focused on calculating the current fixed assets with a depreciation rate and considered both debt and equity as the cost of capital. Li et al. [15] simultaneously took into account three values: operating costs, capital expenditures, and revenues when proposing an effective supply chain network. They highlighted the significance of investing in sustainable resources and fixed assets when planning capital expenditures for the optimal design of a supply chain network.

C. Linear Programming Model in Supply Chain Network Design

Mixed Integer Linear Programming (MILP) is the most commonly used optimization technique for designing complex supply chain networks. Purnomo et al. [16] employed the MILP to minimize the total costs, encompassing production, traceability, transport, inventory, and emission costs within a supply chain, serving multiple customers over various periods. Kazancoglu et al. [17] proposed the MILP to investigate the optimal selection of echelons and transportation alternatives in a closed-loop supply chain network. Moretti et al. [18] proposed the MILP model to choose the strategy of advanced biofuel supply chains including the distribution of network nodes and the efficient planning of logistics activities. Kumar and Kumar [19] employed the MILP to maintain the balanced flow across all stages of the network and optimize the usage of raw materials in production while minimizing both production costs and greenhouse gas emissions.

The MILP commonly serves as an optimization model, which is typically geared towards achieving either cost or revenue objectives. However, this study endeavors to illuminate another dimension within supply chain networks by broadening the scope to encompass the economic value realized through investment. A comprehensive understanding of this issue necessitates a holistic examination of real-world supply chains, influenced by myriad factors including long-term investment capacity and size, potential facility relocations, which may be variable, and pertinent financial metrics such as taxes, cost of capital, and asset depreciation. Notably, this investigation advocates for replacing the singular profit objective with a comprehensive assessment, thereby facilitating strategic, long-term investment decisions concerning asset capacity.

According to the aforementioned literature, it was found that there could be several research gaps, as shown in Table I. To fill these gaps, this study can be contributed as follows:

1) Introduction of EVA as a key factor in the supply chain network design to determine suppliers and prospective distribution centers, production level and production capacity at the plant, and the sizes of the plant, distribution centers, and retailers, as compared to the profit maximization through Mixed Integer Linear Programming (MILP).

2) Through the comparative analysis between Economic Value Added (EVA) and the profit maximization, it can highlight the key advantages at each member in the supply chain when maximizing EVA over the profit under the long-term planning.

III. METHODOLOGY

In the context of food supply chain network design, two models have been developed under deterministic conditions under Mixed-Integer Linear Programming (MILP). The primary objective is to demonstrate the advantages of maximizing Economic Value Added (EVA) over the conventional profit maximization approach. Both models are based on identical datasets and follow the same methodological steps, as shown in Fig. 2.

1. The first step involves defining the objective either maximizing profit or Economic Value Added (EVA). A mathematical model is formulated using deterministic input parameters such as selling prices, purchasing costs, customer demand, and transportation costs. These inputs are integrated into a MILP model, which defines the objective function and includes indices, parameters, constraints, and required decision variables. The goal is to derive optimal solutions for potential suppliers, distribution centers, production levels, capacities, material flows, facility sizes, and inventory levels, aligned with the chosen objective.

2. The second step applies the developed models to a practical scenario specifically, the food supply chain in Southern Vietnam. This application serves to test the models' real-world viability and effectiveness.

3. The final step involves a thorough analysis and comparison of the results obtained from the case study. This analysis focuses on evaluating the performance of the profit maximization model against the EVA maximization model. By comparing the outcomes, stakeholders can discern the long-term economic income from EVA and the short-term profit maximization.

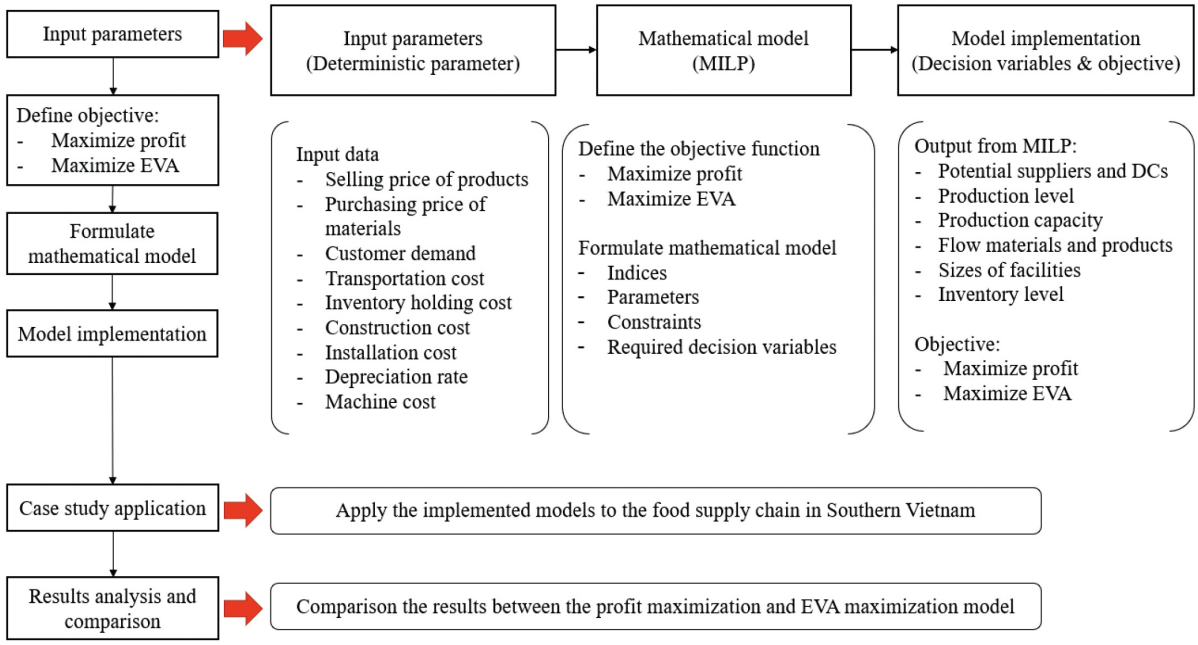


Fig. 2. Procedures of MILP model formulation of the proposed three-step approach

IV. Mathematical Formulation

All optimization models were implemented and solved using the IBM, ILOG, CPLEX, and Optimization Studio on a Windows 10 Pro 64-bit system with an Intel Core i7-8565U CPU running @ 2.0 GHz and 8.0 GB of memory. The supply chain configuration used in this study is illustrated in Fig. 3.

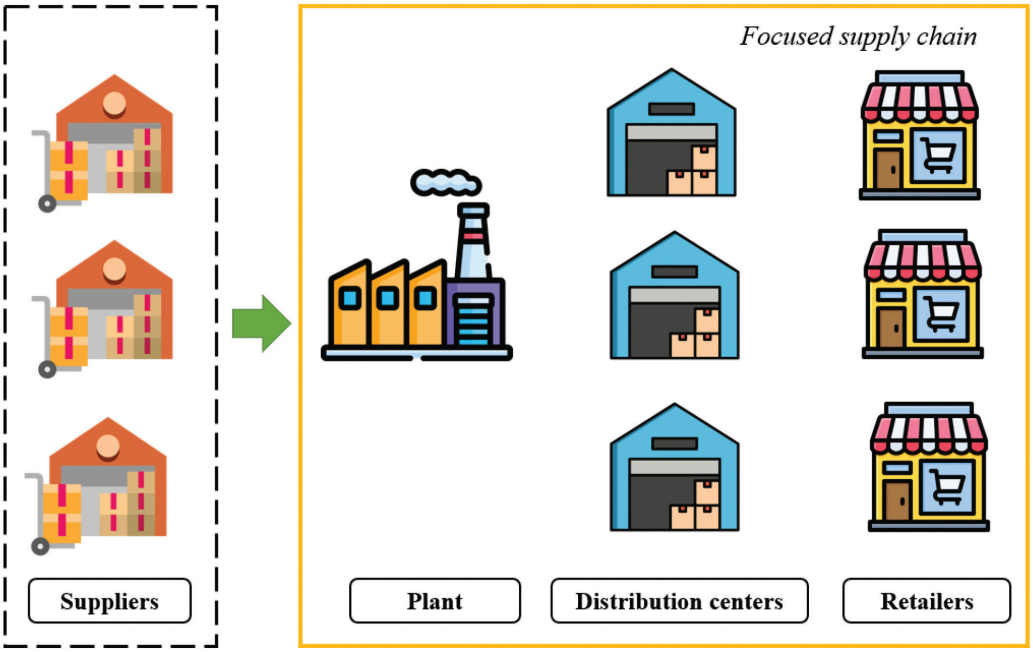


Fig. 3. The supply chain configuration

TABLE I
SUMMARY OF PREVIOUS LITERATURE REVIEWS

References	Number of Echelons	Planning Period		Type of Systems	Objectives	Model Types	Solving Tool
		S	L				
Duffuaa et al., 2024 [20]	4	-	x	SCND	Min cost	MILP	CPLEX
Moretti et al., 2021 [18]	2	-	x	SCND	Min cost	MILP	CPLEX
Li et al., 2020 [15]	4	x	-	SCND	Min cost	MILP	CPLEX
Kumar and Kumar, 2024 [19]	4	-	x	SCND	Min cost	MILP	LINGO
Khalifehzadeh et al., 2015 [21]	4	-	x	SCND	Min cost Max reliability	MILP	LINGO
Sheibani and Niroomand, 2024 [22]	4	x	x	SCND	Min cost Min CO_2 Max social effects	MINLP	CPLEX
Ji and Chiadamrong, 2019 [23]	2	x	-	SCNL	Max profit	MILP	CPLEX
Martins et al., 2017 [24]	3	-	x	SCNR	Min cost	MIP	CPLEX
Aqlan and Lam, 2016 [25]	4	-	x	SCRM	Max profit Min lead time	LP	CPLEX
De Keizer et al., 2015 [26]	3	x	x	SCND	Min cost	MILP	CPLEX
Longinidis and Georgiadis, 2011 [9]	4	-	x	SCND	Max EVA	MILP	CPLEX
Badakhshan and Ball, 2022 [8]	4	x	x	SCND	Max EVA	MILP	CPLEX
This paper	4	x	x	SCND	Max EVA	MILP	CPLEX

Abbreviations: S=Short-term Planning, L=Long-term Planning, SCND=Supply Chain Network, SCNR=Supply Chain Network Redesign, SCRM=Supply Chain Risk Management, SCNL=Supply Chain Network Planning, MINLP=Mixed Integer Non-Linear Programming, MILP=Mixed Integer Linear Programming, MIP=Mixed Integer Programming, LP=Linear Programming

A. Mathematical Notations

Indices

S	Set of the suppliers, $I \in S$
D	Set of the distribution centers, $k \in D$
R	Set of the retailers, $l \in R$
T	Set of the periods, $t \in T$

Deterministic Parameters

C_t^P	Production cost of products at the plant in period t ($/ton)
C_{it}^S	Purchasing cost of raw materials at supplier i in period t ($/ton)
C_t^{HP}	Inventory holding cost of products at the plant in period t ($/ton)
C_{kt}^{HD}	Inventory holding cost of products at distribution center k in period t ($/ton)
C_{lt}^{HR}	Inventory holding cost of products at retailer l in period t ($/ton)
C_{it}^{TR}	Transportation cost of materials from supplier i to the plant in period t ($/ton)
C_{kt}^{TR}	Transportation cost of products from the plant to distribution center k in period t ($/ton)
C_{klt}^{TR}	Transportation cost of products from distribution center k to retailer l in period t ($/ton)
C_{kt}^O	Installation cost of distribution center k in period t ($/ton)
PR_t^{mac}	Production rate per machine at the plant in period t (tons/machine)
$Cost_{mac}$	Purchasing cost of a machine at the plant ($/machine)
$Price_{lt}$	Price of products at retailer l in period t ($/ton)
$Penalty_{lt}$	Penalty cost of lost sales at retailer l in period t ($/ton)
$FAPV_t$	Construction cost at the plant in period t ($/m^2)
$FADV_{kt}$	Construction cost at distribution center k in period t ($/m^2)
$FARV_{lt}$	Construction cost at retailer l in period t ($/m^2)
dr	Depreciate rate (%)
$WACC$	Weighted average cost of capital (%)
D_{lt}	Customer demands at retailer l in period t (tons)
Q_t^{max}	Maximum capacity at supplier i to provide raw materials in period t (tons)
$size_t$	Size of the plant in period t (m^2/ton)
$size_{kt}$	Size of distribution center k in period t (m^2/ton)
$size_{lt}$	Size of retailer l in period t (m^2/ton)
FAP_t	Fixed asset value of the plant at the beginning of period t ($)
FAD_{kt}	Fixed asset value of distribution center k at the beginning of period t ($)
FAR_{lt}	Fixed asset value of retailer l at the beginning of period t ($)
$investP_t$	Investment value for the extra space and machine capacity at the plant in period t ($)
$investD_{kt}$	Investment value for the extra space of distribution center k in period t ($)
$investR_{lt}$	Investment value for the extra space of retailer l in period t ($)
Rev_t	Total revenue received from selling products at all retailers in period t ($)
$Profit_t$	Total net profit calculated by subtracting total costs from the revenue in period t ($)
TC_t^{Profit}	Summation of all costs in the profit maximization in period t ($)
TC_t^{EVA}	Summation of all costs in the EVA maximization in period t ($)
PC_t	Production cost at the plant in period t ($)
RMC_t	Raw material cost transferred from supplier i to the plant in period t ($)
HC_t	Total inventory holding cost at the plant, at all distribution centers, and at all retailers in period t ($)
TLS_t	Total penalty cost of all retailers in period t ($)
DPR_t	Total depreciation cost of all fixed assets at the end of period t ($)
IC_t	Total invested capital of the plant, all distribution centers, and all retailers in period t ($)
FDC_t	Total installation cost of all distribution centers in period t ($)
$PROFIT$	Total net income in all periods ($)
EVA	Total economic value added in all periods ($)

Binary Decision Variables

Y_{it}	1 if supplier i is established, otherwise 0 in period t
Y_{kt}	1 if distribution center k is established, otherwise 0 in period t
L_{klt}	1 if the connection from distribution center k to retailer l is established, otherwise 0 in period t

A. Mathematical Notations (Con.)

Decision Variables

PR_t	Production level at the plant in period t (tons)
M_t^{int}	Integer number of machines at the plant in period t (machines)
M_t^{dec}	Decimal number of machines at the plant in period t (machines)
Cap_t	Size of the plant in period t (tons)
Cap_{kt}	Size of distribution center k in period t (tons)
Cap_{lt}	Size of retailer l in period t (tons)
S_{lt}	Amount of shortages at retailer l in period t (tons)
Q_{it}	Amount of materials transferred from supplier i to the plant in period t (tons)
Q_{kt}	Amount of products transferred from the plant to distribution center k in period t (tons)
Q_{klt}	Amount of products transferred from distribution center k to retailer l in period t (tons)
I_t	Inventory level of products at the plant at the end of period t (tons)
I_{kt}	Inventory level of products at distribution center k at the end of period t (tons)
I_{lt}	Inventory level of products at retailer l at the end of period t (tons)

B. Mathematical Notations

Objective Function

Two scenarios are considered: One focuses on the traditional profit metric, while the other delves into the assessment of Economic Value Added (EVA). With profit maximization, its objective function as presented in Equation (2) aims to maximize the profit by satisfying customer demand, subject to the total costs including the installation cost, the purchasing raw material cost, the transportation cost, the inventory holding cost, the production cost, and the shortage cost. The calculation of the maximization of Economic Value Added (EVA), as outlined in Equation (3), is calculated by subtracting the capital charge (WACC multiplied by invested capital) from the company's profit. In essence, it is a measure of how much value a company is generating above and beyond the cost of the capital that it employs in its operations.

$$PROFIT = \sum_{t=1}^{T} (Rev_t - TC_t^{Profit}) \quad (2)$$

$$EVA = \sum_{t=1}^{T} (Profit_t - WACC \times IC_t) \quad (3)$$

where:

$$Profit_t = \sum_{t=1}^{T} (Rev_t - TC_t^{EVA}) \quad (4)$$

Constraints

1) Revenue generated from the demand satisfaction is calculated by subtracting the revenue lost due to unmet demand (shortages) from all initial demands at the retailers and then multiplying with the selling price per ton, as expressed in Equation (5).

$$Profit_t = \sum_{l=1}^{R} (D_{lt} - S_{lt}) \times price_{lt}, \forall t \quad (5)$$

2) In the context of the profit maximization, the ordinary total costs encompass all associated expenses related to production, transportation, inventory holding, material procurement, installation of distribution centers, and shortage penalties, as stated in Equation (6). Additionally, the depreciation of all fixed assets has been included within the framework of maximizing Economic Value Added (EVA), as expressed in Equation (7).

$$TC_t^{Profit} = PC_t + TRC_t + HC_t + RMC_t + FDC_t + TLS_t, \forall t \quad (6)$$

$$TC_t^{EVA} = PC_t + TRC_t + HC_t + RMC_t + FDC_t + TLS_t + DPR_t, \forall t \quad (7)$$

3) The production cost at the plant is calculated by multiplying the units of products produced at the plant by the production cost per ton, as shown in Equation (8).

$$PC_t = \sum_{l=1}^{R} PR_t \times C_t^P, \forall t \quad (8)$$

4) Transportation cost comprises the cost of transporting materials from suppliers to the plant, products from the plant to distribution centers, and products from distribution centers to retailers, multiplied by the respective transportation cost per ton, as shown in Equation (9).

$$TRC_t = \sum_{i=1}^{S} C_{it}^{TR} \times Q_{it} + C_{kt}^{TR} \times Q_{kt} + \sum_{k=1}^{D} C_{klt}^{TR} \times Q_{klt}, \forall t, k, l \quad (9)$$

5) Inventory holding cost, representing the expenses incurred in storing products at the plant, distribution centers, and retailers, is calculated by multiplying their respective inventory holding cost per ton, as specified in Equation (10).

$$HC_t = \frac{I_{t-1} + I_t}{2} \times C_t^{HP} + \sum_{k=1}^{D} C_{kt}^{HD} \times \frac{I_{k(t-1)} + I_{kt}}{2} + \sum_{l=1}^{R} C_{lt}^{HR} \times \frac{I_{l(t-1)} + I_{lt}}{2}, \forall t \quad (10)$$

6) Raw material cost transferred from suppliers to the plant is determined by multiplying the quantity of raw materials by their respective purchasing price, as shown in Equation (11).

$$RMC_t = \sum_{i=1}^{S} C_{it}^{S} \times Q_{it}, \forall t \quad (11)$$

7) The installation cost is charged in each period when a distribution center operates, as detailed in Equation (12).

$$FDC_t = \Sigma_{k=1}^{D} C_{kt}^{O} \times Y_{kt}, \forall t \quad (12)$$

8) Depreciation cost is determined by multiplying all fixed asset values by the Depreciation rate (*dr*) at the end of period *t*, as detailed in Equation (13).

$$DPR_t = (FAP_t + \Sigma_{k=1}^{D} FAD_{kt} + \Sigma_{l=1}^{R} FAR_{lt}) \times dr, \forall t \quad (13)$$

9) Fixed asset values, encompassing the existing fixed asset values plus new invested values of facilities such as the production capacity, space of potential new distribution centers, and retailers, are represented in Equations (14) - (16).

$$FAP_t = FAP_{(t-1)} \times (1-dr) + investP_t, \forall t \quad (14)$$

$$FAD_{kt} = FAD_{k(t-1)} \times (1-dr) + investD_{kt}, \forall t,k \quad (15)$$

$$FAR_{lt} = FAR_{l(t-1)} \times (1-dr) + investR_{lt}, \forall t,l \quad (16)$$

Where:

$$investP_t = [Cap_t - Cap_{(t-1)}] \times FAPV_t \times size_t + M_t^{int} \times cost_{mac}, \forall t \quad (17)$$

$$investD_t = [Cap_{kt} - Cap_{k(t-1)}] \times FADV_t \times size_{kt}, \forall t \quad (18)$$

$$investR_t = [Cap_{lt} - Cap_{l(t-1)}] \times FARV_t \times size_{lt}, \forall t \quad (19)$$

Equation (17) determines the total plant investment, encompassing production machine capacity (assumed to be $100,000 for a machine) and the space investment cost within a specific period. This equation suggests additional capacity and space when the customer demand rises, with no investment required for constant or decreasing customer demands. Equations (18) - (19) calculate the space investment cost at the distribution centers and retailers. Facility sizing is estimated using a simplified assumption of one square meter of storage per one ton of product.

10) The capital investment is determined by summing the total fixed asset values at the plant, distribution centers, and retailers at the beginning of the period, as stated in Equation (20).

$$IC_t = FAP_t + FAD_{kt} + FAR_{lt}, \forall t,k,l \quad (20)$$

11) Each supplier has the maximum capacity of 20,000 tons to supply raw materials to the plant, as shown in Equation (21).

$$Q_{it} \le Q_t^{max} \times Y_{it}, \forall t,i \quad (21)$$

12) In each period, at least one supplier and one distribution center must be established as expressed in Equations (22) - (23). Furthermore, all retailers must be linked with at least one distribution center during each period, as shown in Equation (24). The connection between each distribution center and retailers is established, as shown in Equation (25). Lastly, it is imperative to ensure the continuity of distribution center operations throughout the designated design period, as shown in Equation (26).

$$\Sigma_{i=1}^{S} Y_{it} \ge 1, \forall t \quad (22)$$

$$\Sigma_{k=1}^{D} Y_{kt} \ge 1, \forall t \quad (23)$$

$$\Sigma_{k=1}^{D} L_{klt} \ge 1, \forall t, l \quad (24)$$

$$L_{klt} \le Y_{kt}, \forall t, k, l \quad (25)$$

$$L_{kt} \ge Y_{k(t-1)}, \forall t, k \quad (26)$$

13) Inventory balance at each node in the supply chain is calculated as the sum of the products that flow into the facility, adding the remaining products from the previous period, and then subtracting the products that flow out of the facility, as shown in Equations (27) - (29).

$$I_t = I_{j(t-1)} + PR_t - \Sigma_{k=1}^{D} Q_{kt}, \forall t \quad (27)$$

$$I_{kt} = I_{k(t-1)} + Q_{kt} - \Sigma_{l=1}^{R} Q_{klt}, \forall t,k \quad (28)$$

$$I_{lt} = I_{l(t-1)} + \Sigma_{k=1}^{D} Q_{klt} - (D_{lt} - LS_{lt}), \forall t,l \quad (29)$$

14) Suppliers are assumed to supply the raw materials immediately to the plant when it receives the order from the plant, as expressed in Equation (30).

$$PR_t = \Sigma_{i=1}^{S} Q_{it}, \forall t \quad (30)$$

15) The machine capacity at the plant, representing the required number of machines, is determined by dividing the production level in each period by the production rate per machine, which is set at 2,000 tons per machine per period (year), as stated in Equation (31). However, since the number of machines must be an integer, as described in Equations (32) - (35), several constraints are imposed. Equation (32) ensures that the minimum number of machines needed in each period is at least equal to the total production level divided by the production rate per machine. To prevent underestimation, Equation (33) sets an upper limit on the number of machines required by adding one extra machine to the calculation. Additionally, Equation (34) safeguards against reducing the required number of machines, even in the event of a decrease in customer demands. These equations collectively ensure sufficient machine capacity while accommodating variations in production levels and customer demands.

$$M_t^{dec} = \frac{PR_t}{PR_t^{mac}}, \forall t \quad (31)$$

$$M_t^{int} \geq M_t^{dec}, \forall t \quad (32)$$

$$M_t^{int} \leq M_t^{dec} + 1, \forall t \quad (33)$$

$$M_t^{\text{int}} \geq M_{t-1}^{int}, \forall t \quad (34)$$

16) Equation (35) specifies that the size of the plant must not decrease as compared to the previous period. Likewise, similar constraints are applied to the sizes allocated for the distribution centers and the retailers, as depicted in Equations (36) - (37).

$$Cap_t = max\left[PR_t, Cap_{t-1}\right], \forall t \quad (35)$$

$$Cap_{kt} = max\left[Q_{kt}, Cap_{k(t-1)}\right], \forall t, k \quad (36)$$

$$Cap_{lt} = max\left[\sum_k^D Q_{klt}, Cap_{l(t-1)}\right], \forall t, l \quad (37)$$

17) The penalty of shortages incurred due to the lost sales is calculated by multiplying the number of tons shortage by the shortage penalty cost per ton, as shown in Equation (38).

$$TLS_t = \sum_{l=1}^{R} S_{lt} \times penalty_{lt}, \forall t \quad (38)$$

18) The maximum size of each distribution center is limited to 20,000 tons, as indicated in Equation (39).

$$Cap_{kt} \leq 20{,}000, \forall t, k \quad (39)$$

19) Customer Service Level (CSL) ensures that all retailers must distribute a minimum quantity of products to their customers in each period. The study guarantees at least a 90% service level at all retailers, as expressed in Equation (40).

$$\sum_{k=1}^{D} Q_{klt} \geq 0.9 * D_{lt}, \forall t, l \quad (40)$$

20) Equation (41) ensures that the production level at the plant does not exceed the total customer demands for a given period, thereby aligning the production level with the customer demands.

$$PR_t \geq \sum_{l=1}^{R} D_{lt}, \forall t \quad (41)$$

21) Equations (42) - (47) are established to ensure that the values of all decision variables are non-negative, with some constraints to be integer or binary.

$$Q_{it}, Q_{kt}, Q_{klt} \geq 0, \forall i, k, l, t \quad (42)$$

$$I_t, I_{kt}, I_{lt} \geq 0, \forall k, l, t \quad (43)$$

$$PR_t, M_t^{int} \geq 0, \forall t \quad (44)$$

$$Y_{it}, Y_{kt}, Y_{klt} \in \{0,1\}, \forall i, k, l, t \quad (45)$$

$$S_{lt} \geq 0, \forall t, l \quad (46)$$

$$Cap_t, Cap_{kt}, Cap_{lt} \geq 0, \forall t, k, l \quad (47)$$

V. Case Study

A. Problem Description

A case study is constructed to illustrate and evaluate the effectiveness of the proposed MILP decision-making model in addressing a FSCND (Food Supply Chain Network Design) problem within a small-sized food industry in the Southern Region of Vietnam. The model aims to demonstrate the advantages of utilizing EVA over profit maximization. Following sorting at the supplier level, the raw materials are then transferred to the plant for processing. Here, they undergo a series of essential steps including washing, sorting, cutting, packaging, and labeling. Having been processed, the products are dispatched to Distribution Centers (DCs) for additional quality assessments and sorting, then distributed to retailers, who display and sell them to consumers.

In this study, seven districts in southern Vietnam are considered in the supply chain network. These districts comprise one plant located in Dong Thap, three distribution centers situated in Can Tho, Tien Giang, and Ho Chi Minh, and three retailers located in Tay Ninh, Binh Duong, and Soc Trang. The investigation focuses on the planning horizon of the proposed model, from January 2019 to December 2023, comprising five years. This location provides access to an abundant amount of food-based agriculture resources. There are three qualified suppliers, who supply raw materials at varying prices based on the quality. They consistently fulfill the requirements of the plant. Additionally, there are three potential distribution centers, and three retailers with varying demand levels across different locations, as depicted in Fig. 4.

This supply chain network structure can be segmented into three distinct stages. In the initial stage, the suppliers provide raw materials to the plant for product fabrication. Subsequently, in the second stage, products are requested from the plant by the distribution centers and then dispatched to the retailers in the third stage. Operations within this framework entail making numerous decisions throughout the supply chain network in both strategic and operational planning. These decisions encompass identifying the suppliers, and the potential distribution centers, determining the production level and the production capacity at the plant, as well as managing the sizes of the plant, distribution centers, and retailers.

B. Input Data and Cost Structure

The objective of this study is to examine the benefits of maximizing the Economic Value Added

(EVA) of the FSCND (Food Supply Chain Network Design) instead of solely focusing on maximizing the profit. Major costs are influenced by the supply chain network decisions, including raw material cost, production cost at the plant, transportation cost, inventory holding cost, installation cost of all potential distribution centers, penalty cost of shortages, and depreciation cost of the plant, distribution centers, and retailer's buildings and machines in FSCND. Total revenue is generated at all retailers by selling the products to customers. It is also assumed that shortages are permitted with the penalty cost, requiring all supply chain members to balance between enlarging the sizes of facilities and risking shortages to achieve optimal profitability while considering investment capital.

All data have been scaled with a common factor and presented on an average basis. There is one plant (P) that manufactures the products from raw materials supplied by three suppliers (S1, S2, S3). These products are intended to reach three retailers (R1, R2, R3) located in different locations to meet customer demands, facilitated through a network of three potential distribution centers (DC1, DC2, DC3). The problem entails finding the optimal three echelons in the FSCND configuration illustrated in Fig. 3, in a planning horizon of 5 periods (years). The requisite data for evaluating the model are provided in Tables II-VI. They display the transportation costs associated with deliveries from suppliers to the plant, from the plant to potential distribution centers, and from potential distribution centers to retailers. Transportation costs can vary depending on the distance between facilities while maintaining the assumption that these costs remain constant throughout the planning horizon. Table V shows the purchasing cost of raw materials at the supplier, while Table VI presents the customer demands at the retailers measured in tons per period across five distinct years. It shows that the fluctuation in the customer demands occurs from year to year, and there is a significant peak upward trend in the third year, followed by a decline and subsequent stabilization in the later years.

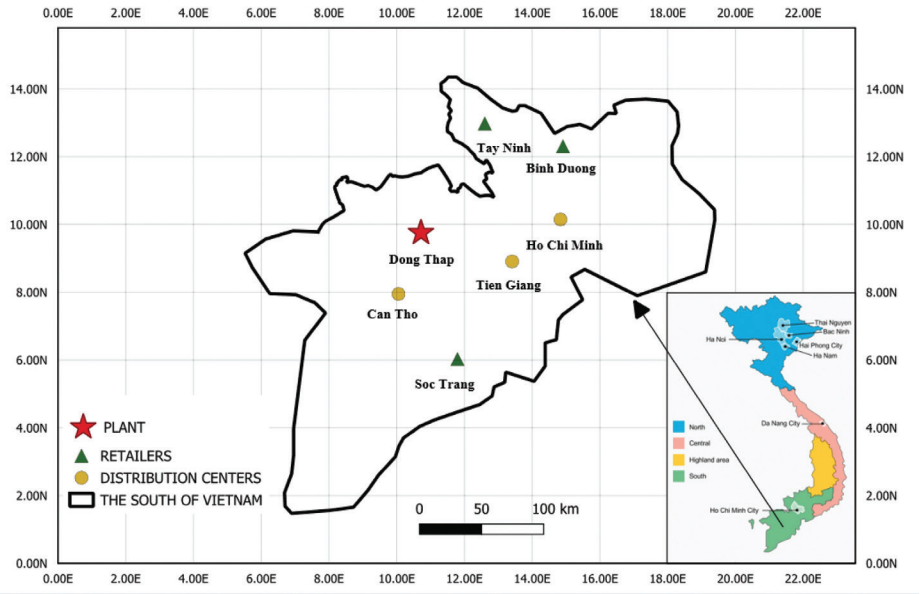


Fig. 4. The location of facilities in the southern region of Vietnam (QGIS)

TABLE II
TRANSPORTATION COST (C_t^{TR}) OF RAW MATERIALS FROM SUPPLIERS TO THE PLANT ($/TON)

Plant	Suppliers		
	S1	S2	S3
P	88.0	110.0	154.0

TABLE III
TRANSPORTATION COST (C_{kt}^{TR}) OF PRODUCTS FROM THE PLANT TO DISTRIBUTION CENTERS ($/TON)

Plant	Distribution Centers		
	DC1	DC2	DC3
P	176.0	154.0	110.0

TABLE IV
TRANSPORTATION COST OF PRODUCTS FROM DISTRIBUTION CENTERS TO RETAILERS (C_{lt}^{TR}) ($/TONS)

Distribution Centers	Retailers		
	R1	R2	R3
DC1	176.0	167.2	173.8
DC2	154.0	165.0	132.0
DC3	121.0	110.0	88.0

TABLE V
PURCHASING COST OF RAW MATERIALS (C_{it}^{S}) AT SUPPLIERS ($/TON)

Suppliers		
S1	S2	S3
330	341	352

TABLE VI
CUSTOMER DEMAND (D_{lt}) AT EACH RETAILER PER YEAR (TONS/YEAR)

Retailers	Years				
	1	2	3	4	5
R1	6,500	7,700	8,500	7,000	7,000
R2	5,300	6,300	7,500	7,500	6,600
R3	5,000	6,000	7,000	6,000	6,500

TABLE VII
OTHER PARAMETERS (%)

Other Parameters	Rate
Weighted Average Cost of Capital *(WACC)*	0.1
Depreciation Rate *(dr)*	0.1
Customer Service Level *(CSL)*	0.9

Table VII outlines important parameters expressed as percentages. The Weighted Average Cost of Capital (WACC) signifies the average rate of return expected to be paid to investors for financing assets, indicated as 10%. The depreciation rate noted as 0.1, indicates a 10% fixed asset depreciated value annually. Additionally, Customer Service Level (CSL) of 0.9 represents the target to fulfill at least 90% of the customer demands promptly and satisfactorily within the supply chain operation.

The selling price $(price_{lt})$ is set at $2,200 per ton, with a corresponding shortage penalty cost the same as the selling price ($lostsale_{lt}$). The production cost is estimated at 18% of the selling price (C_t^P = $396) per ton, while the inventory holding cost is estimated at 20% of the selling price ($C_t^{HP} = C_{kt}^{HD} = C_{lt}^{HR}$ = $440) per ton per year. The installation cost (C_{kt}^{O}) would be incurred at any distribution center upon its opening at $600,000 per year. Acknowledging the importance of plant, distribution center, and retailer sizes in the capital investment and depreciation cost determination, this study has standardized the building construction cost for each member in the chain ($FAPV_t = FAPV_{kt} = FAPV_{lt}$ = $500 per square meter) with one ton per one square meter at the plant $(size_t)$, distribution centers $(size_{kt})$, and retailers $(size_{lt})$.

VI. RESULTS AND DISCUSSIONS

The section is divided into two segments. The first segment outlines the outcomes resulting from the application of the previously described methodology to both models, within the framework of the case study conducted in the southern region of Vietnam. The second segment commences with a discourse on the disparities between profit and Economic Value Added (EVA) maximization. These differences are comprehensively illustrated in Tables VIII - XVII.

A. Profit and EVA Maximization

TABLE VIII
BINARY DECISION VARIABLES FOR SUPPLIERS SELECTION (Y_{it}) IN THE PROFIT AND EVA MAXIMIZATION

Suppliers	PROFIT			EVA		
	S1	S2	S3	S1	S2	S3
Year 1	1	0	0	1	0	0
Year 2	1	0	0	1	0	0
Year 3	1	1	0	1	0	0
Year 4	1	1	0	1	0	0
Year 5	1	1	0	1	0	0

Close=0, Open=1

TABLE IX
BINARY DECISION VARIABLES FOR POTENTIAL DISTRIBUTION CENTERS SELECTION (Y_{kt}) IN THE PROFIT AND EVA MAXIMIZATION

Distribution Centers	PROFIT			EVA		
	DC1	DC2	DC3	DC1	DC2	DC3
Year 1	0	0	1	0	0	1
Year 2	0	0	1	0	0	1
Year 3	0	1	1	0	0	1
Year 4	0	1	1	0	0	1
Year 5	0	1	1	0	0	1

Close=0, Open=1

During the five-year planning horizon, Supplier 1 (S1) emerges as the preferred choice across both models due to its combination of lower raw material price per ton and transportation expenses compared to Supplier 2 (S2) and Supplier 3 (S3), as indicated in Table VIII. For both models, each supplier's annual capacity is capped at 20,000 tons, as shown in Equation (17). Therefore, S2 will continue to supply raw materials once the plant's orders exceed S1's maximum capacity. As shown in Table VIII, S2 is selected in years 3, 4, and 5 for the case of profit maximization, but not for EVA maximization.

According to Table IX, DC3 consistently emerges as the preferred choice in both optimization models throughout the planning period. However, for the profit maximization, DC2 is chosen to open in years 3, 4, and 5. Conversely, for Economic Value Added (EVA) maximization, only DC3 is selected.

The decision to open DC2 for profit maximization reflects a focus on maximizing sales to drive profitability. It prioritizes the avoidance of shortages at the retailers by adding another distribution center to meet all demands. In contrast, for EVA maximization, the emphasis remains solely on DC3, possibly due to factors such as cost efficiency and overall fixed asset utilization in the long term, thereby ensuring

optimal economic performance throughout the whole planning horizon.

TABLE X
DECISION VARIABLES FOR PRODUCTION LEVEL (PR_t) AT THE PLANT (TONS) IN THE PROFIT AND EVA MAXIMIZATION

Years	1	2	3	4	5
PR_t (PROFIT)	16,800	20,000	23,000	20,500	20,100
PR_t (EVA)	20,000	20,000	20,000	20,000	20,000

The Production level (PR) designated for each year in the profit maximization consistently aligns with the customer demand, as evidenced in Table X. The plant consistently meets the orders placed by the distribution centers, which receive orders from the retailers. This alignment between the production levels and the customer demands ensures that the supply chain operates efficiently and effectively, ultimately contributing to the maximization of profit.

In contrast, in the pursuit of Economic Value Added (EVA) maximization, the plant adheres to a different strategy, as presented in Table X. The plant is advised to manufacture quantities equal to or exceeding the orders from the distribution centers in the initial year, ensuring an inventory buffer for subsequent periods, as shown in Table XVII, considering the fluctuation and the trend of the demand predictions. This strategy empowers the plant to manage its production better during periods of heightened demand by leveraging prepared inventory from prior periods. Consequently, it minimizes the necessity for acquiring excessive machinery or expanding the space of the plant during peak periods, which would not be fully utilized when the demand recedes. Moreover, the production levels in each period are influenced by the selection of potential DCs and the sizes of these centers, as shown in Table XV.

TABLE XI
DECISION VARIABLES FOR MACHINE CAPACITY (M_t) AT THE PLANT (MACHINES) IN THE PROFIT AND EVA MAXIMIZATION

Years	1	2	3	4	5
M_t (PROFIT)	9	10	12	12	12
M_t (EVA)	10	10	10	10	10

In Table XI, the decision variables for the machine capacity are presented under both profit and EVA maximization scenarios across the five-year planning horizon. A comparison of machine capacity reveals notable differences between the two optimization objectives. Under the profit maximization, machine capacity fluctuates, reaching its peak in the third, fourth, and fifth years at 12 machines. In contrast, the EVA maximization maintains a constant machine capacity of 10 machines throughout the planning horizon. This divergence in machine capacity allocation underscores the strategic trade-offs between short-term profitability and long-term value creation. The decision to maintain a constant machine capacity under the EVA maximization suggests a focus on machine efficiency and avoidance of unnecessary capital expenditures.

TABLE XII
DECISION VARIABLES FOR AMOUNT TRANSFERRED FROM THE PLANT TO DISTRIBUTION CENTERS (TONS) (Q_{kt}) IN THE PROFIT MAXIMIZATION

	Plant (P)				
Years	**1**	**2**	**3**	**4**	**5**
DC1	-	-	-	-	-
DC2	-	-	3,000	500	100
DC3	16,800	20,000	20,000	20,000	20,000

TABLE XIII
DECISION VARIABLES FOR SIZES (Cap_t, Cap_{kt}, Cap_{lt}) AT THE PLANT, DISTRIBUTION CENTERS, AND RETAILERS (TONS) IN THE PROFIT MAXIMIZATION

Years	1	2	3	4	5
P	16,800	20,000	23,000	23,000	23,000
DC1	-	-	-	-	-
DC2	-	-	3,000	3,000	3,000
DC3	16,800	20,000	20,000	20,000	20,000
R1	6,500	7,700	8,500	8,500	8,500
R2	5,300	6,300	7,500	7,500	7,500
R3	5,000	6,000	7,000	7,000	7,000

TABLE XIV
DECISION VARIABLES FOR AMOUNT TRANSFERRED FROM THE PLANT TO DISTRIBUTION CENTERS (TONS) (Q_{kt}) IN THE EVA MAXIMIZATION

	Plant (P)				
Years	**1**	**2**	**3**	**4**	**5**
DC1	-	-	-	-	-
DC2	-	-	-	-	-
DC3	20,000	20,000	20,000	20,000	20,000

TABLE XV
DECISION VARIABLES FOR SIZES (Cap_t, Cap_{kt}, Cap_{lt}) AT THE PLANT, DISTRIBUTION CENTERS, AND RETAILERS (TONS) IN THE EVA MAXIMIZATION

Years	1	2	3	4	5
P	20,000	20,000	20,000	20,000	20,000
DC1	0	0	0	0	0
DC2	0	0	0	0	0
DC3	20,000	20,000	20,000	20,000	20,000
R1	6,500	7,700	8,200	8,200	8,200
R2	5,300	6,300	7,500	7,500	7,500
R3	5,000	6,000	7,000	7,000	7,000

TABLE XVI
DECISION VARIABLES FOR STORAGES (S_{lt}) AT RETAILERS (TONS) IN THE PROFIT AND EVA MAXIMIZATION

Retailers	PROFIT			EVA		
	R1	R2	R3	R1	R2	R 3
Year 1	-	-	-	-	-	-
Year 2	-	-	-	-	-	-
Year 3	-	-	-	300	-	-
Year 4	-	-	-	-	-	-
Year 5	-	-	-	100	-	-

The size allocation within the supply chain network during the five years for profit maximization is illustrated in Table XIII, where the size at the Plant (P), Distribution Centers (DC1, DC2, DC3), and Retailers (R1, R2, R3) are detailed.

The capacities are adjusted to increase when the demand increases and remain the same when the demand drops. The significant observation lies in the size management of Distribution Center 2 (DC2). Despite the gradual decrease in the quantity stored in this center since its first year of operation (year 3) as indicated in Table XIII, attributed to a decline in total customer demands, it is noteworthy that the center still maintains a size of 3,000 tons.

The size allocation within the supply chain network during five years for EVA maximization is illustrated in Table XV. The size at the Plant (P), Distribution Centers (DC1, DC2, DC3), and Retailers (R1, R2, R3) are presented.

There is a notable disparity compared to the profit maximization, where DC3 consistently operates at the maximum size and the plant remains stable at a size of 20,000 tons regardless of fluctuations in total customer demands at the retailers.

The profit maximization model focuses solely on maximizing the profit, disregarding investment and depreciation costs. As demonstrated in Tables XVI-XVII, this model does not consider the necessity for holding remaining inventory or allowing shortages. Conversely, EVA maximization model indicates the potential for some shortages when the size of DC3 reaches its limitation. Adding a new distribution center might not always be beneficial, since the additional costs could outweigh the gains.

TABLE XVII
DECISION VARIABLES FOR INVENTORY HOLDING (I_t, I_{kt}, I_{lt}) (TONS) IN THE PROFIT AND EVA MAXIMIZATION

	Plant		Distribution Centers						Retailers					
	PROFIT	EVA	PROFIT			EVA			PROFIT			EVA		
	P	P	DC1	DC2	DC3	DC1	DC2	DC3	R1	R2	R3	R1	R2	R3
Year 1	-	-	-	-	-	-	-	3,200	-	-	-	-	-	-
Year 2	-	-	-	-	-	-	-	3,200	-	-	-	-	-	-
Year 3	-	-	-	-	-	-	-	500	-	-	-	-	-	-
Year 4	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Year 5	-	-	-	-	-	-	-	-	-	-	-	-	-	-

B. Comparison between the Profit and EVA Maximization

TABLE XVIII
THE COMPARATION BETWEEN EVA AND THE PROFIT MAXIMIZATION ($)

Objective	PROFIT	EVA
	$45,868,200	$19,556,832
Comparison	**EVA**	**PROFIT**
	$16,717,138	$45,468,200
Other Costs:		
Revenue (*Rev*)	$220,880,000	$220,000,000
Production Cost (*PC*)	$39,758,400	$39,600,000
Raw Material Cost (*RMC*)	$33,171,600	$33,000,000
Transportation Cost (*TRC*)	$31,017,800	$30,533,800
Installation Cost (*FDC*)	$4,800,000	$3,000,000
Inventory Holding Cost (*HC*)	$66,264,000	$67,518,000
Penalty Shortage (*TLS*)	$0	$880,000
Invested Capital (*IC*)	$145,755,310*	$129,556,840
Depreciation Cost (*DPR*)	$14,575,531*	$12,955,684

*Not applicable for the profit calculation (only for illustration)

The interpretation underscores the distinction between the profit and EVA maximization in decision-making processes concerning various variables, including suppliers and distribution centers selection, production level and machine capacity at the plant, shortages at retailers, sizes of the plant, distribution centers, and retailers. The comparison aims to highlight that EVA maximization offers more advantages than solely focusing on profit maximization. While profit maximization primarily concentrates on optimizing revenue while minimizing costs to boost profitability, EVA maximization in addition considers invested capital and depreciation, as shown in Table XVIII. This underscores the importance of considering factors beyond revenue and total costs alone, as they directly influence the objective. Ignoring these aspects may appear advantageous in the short term but could lead to erroneous decisions and increased financial losses over the long term.

These highlights underscore the advantages of employing Economic Value Added (EVA) maximization over profit maximization.

1. EVA prioritizes not just minimizing total costs, but also factoring in facility and machinery investment costs. This approach suggests opening only DC3 and operating it at full capacity at the distribution center. This is to avoid unnecessary expansion during the demand surges that would not be sustainable in the long term if the future demand drops. For example, as seen in Table XII, the profit maximization model suggests opening another Distribution Center (DC2) when the demand exceeds the maximum size of DC3. Nevertheless, the customer demand shows a gradual decline starting from year 3, resulting in DC2 not being fully utilized after year 3.

2. At the plant, the EVA maximization model recommends setting production levels that could match or exceed the customer demands to maintain some inventory buffers, as shown in Table XVII. This approach enables efficient production management during the peak demand period by utilizing inventory reserves from earlier periods. As a result, it leads to minimize the need for increased the number of machines as well as plant expansion during the high demand period, which could be not fully utilized once the future customer demands at the retailers drop.

3. It illustrates the substantial benefits of EVA maximization in the capital utilization compared to profit maximization, as shown in Table XVIII. Particularly, the EVA maximization model facilitates the lower investment capital in the fixed assets. Instead of allocating $145,755,310 under profit maximization, the EVA maximization model suggests an allocation of only $129,556,840, with a seemingly minimal difference in the achieved profitability. This highlights an importance of considering all economic values, not only operational costs in the supply chain, as overlooking them can lead to inefficient allocation of the fixed assets.

In conclusion, in terms of considering the investment and utilization of facilities within the supply chain network, EVA optimization offers valuable insights by integrating various factors and fostering connections across the chain. It recognizes the interdependence of different elements within the supply chain such as inventory, invested capital, shortage, and other costs. By doing so, EVA optimization ensures that decisions, regarding size allocation and resource utilization, are aligned with broader supply chain objectives in the long term. This holistic approach reinforces the interconnectedness of the supply chain network, preventing it from solely focusing on the immediate profit.

VII. Conclusion

In this study, we employed Mixed-Integer Linear Programming (MILP) for the optimal design of activities within a food supply chain network design. This approach has been applied to a real case study where agricultural regions of South Vietnam have been considered. The case study was based on the production of food from the raw materials available. The primary objective was to highlight the benefits of employing Economic Value Added (EVA) to determine the optimal operating conditions of the analyzed supply chain, aiming for maximizing long-term economic income, as opposed to the conventional approach of maximizing profit commonly utilized in most studies. The optimal results derived from the profit maximization model might lead to underutilized capacities of facilities, resulting in a waste of resources and missed opportunities. This could potentially cause incorrect operational assessments and have adverse effects on long-term strategic planning. Consequently, the proposed EVA emerges as a valuable decision-making tool in practice. EVA maximization impresses crucial importance on the company's wealth assessment, particularly about the allocation of invested capital. This allocation directly impacts on the production levels and capacity at the plant, the selection of suppliers and distribution centers, as well as the sizes at distribution centers and retailers, all while considering shortages.

The primary limitation of our case study lies in the assumption of deterministic data, which may lead to imprecise results when applied to real-world scenarios. Decision-makers should prioritize the acquisition and utilization of data based on true judgment regarding past resources. Furthermore, future researchers are encouraged to incorporate uncertain data into the models to better align with practical applications and improve the robustness of the analyzes.

Acknowledgement

This work was supported by the EFS scholarship awarded by Sirindhorn International Institute of Technology, Thammasat University.

References

[1] S. Pang and M. C. Chen, "Investigating the Impact of Consumer Environmental Consciousness on Food Supply Chain: The Case of Plant-Based Meat Alternatives," *Technological Forecasting and Social Change*, vol. 201, no. 4, p. 123190, Feb. 2024.

[2] S. Akhtari and T. Sowlati, "Hybrid Optimization-Simulation for Integrated Planning of Bioenergy and Biofuel Supply Chains," *Applied Energy*, vol. 259, p. 114124, Feb. 2020.

[3] P. Longinidis and M. C. Georgiadis, "Integration of Financial Statement Analysis in the Optimal Design of Supply Chain Networks under Demand Uncertainty," *International Journal of Production Economic*s, vol. 129, no. 2, pp. 262-276, Oct. 2010.

[4] P. R. Burgess, F. T. Sunmola, and S. Wertheim-Heck, "A Review of Supply Chain Quality Management Practices in Sustainable Food Networks," *Heliyon*, vol. 9, no. 11, p. e21179, Nov. 2023.
[5] F. Gholian-Jouybari, M. Hajiaghaei-Keshteli, A. Bavar et al., "A Design of a Circular Closed-Loop Agri-Food Supply Chain Network: A Case Study of the Soybean Industry," *Journal of Industrial Information Integration*, vol. 36, no. 3, p. 100530, Sep. 2023.
[6] F. Gholian-Jouybari, M. Hajiaghaei-Keshteli, N. R. Smith et al., "An In-Depth Metaheuristic Approach to Design a Sustainable Closed-Loop Agri-Food Supply Chain Network," *Applied Soft Computing*, vol. 150, no. 5, p. 111017, Nov. 2023.
[7] Statista Market Insights. (2023, Mar. 10). *Food-Vietnam*. [Online]. Available: https://www.statista.com/outlook/cmo/food/vietnam
[8] E. Badakhshan and P. Ball, "A Simulation-Optimization Approach for Integrating Physical and Financial Flows in a Supply Chain Under Economic Uncertainty," *Operations Research Perspectives*, vol. 10, no. 17, p. 100270, Feb. 2023.
[9] P. Longinidis and M. C. Georgiadis, "Managing the Trade-offs between Financial Performance and Credit Solvency in the Optimal Design of Supply Chain Networks under Economic Uncertainty," *Computers & Chemical Engineering*, vol. 48, pp. 264-279, Sep. 2012.
[10] A. Nagurney, "Optimal Supply Chain Network Design and Redesign at Minimal Total Cost and with Demand Satisfaction," *International Journal of Production Economics*, vol. 128, no. 1, pp. 200-208, Nov. 2011.
[11] M. Kashanian and S. M. Ryan, "Design of a Supply Chain Network for Chemicals from Biomass Using Green Electrochemistry," *Cleaner Logistics and Supply Chain*, vol. 10, p. 100132, Jan. 2022.
[12] A. Ala, V. Simic, N. Bacanin et al., "Blood Supply Chain Network Design with Lateral Freight: A Robust Possibilistic Optimization Model," *Engineering Applications of Artificial Intelligence*, vol. 133, p. 108053, Feb. 2024.
[13] S. Jakub, B. Viera, and K. Eva, "Economic Value Added as a Measurement Tool of Financial Performance," *Procedia Economics and Finance*, vol. 26, pp. 484-489, Jan. 2015.
[14] J. Ashayeri and L. Lemmes, "Economic Value Added of Supply Chain Demand Planning: A System Dynamics Simulation," *Robotics and Computer-Integrated Manufacturing*, vol. 22, no. 5-6, pp. 550-556, Oct. 2006.
[15] L. Li, H. Manier, and M. A. Manier, "Integrated Optimization Model for Hydrogen Supply Chain Network Design and Hydrogen Fueling Station Planning," *Computers & Chemical Engineering*, vol. 134, no. 5-6, p. 106683, Mar. 2024.
[16] M. R. A. Purnomo, I. D. Wangsa, N. Rizky et al., "A Multi-Echelon Fish Closed-Loop Supply Chain Network Problem with Carbon Emission and Traceability," *Expert Systems with Applications*, vol. 210, no. 4, p. 118416, Aug. 2022.
[17] Y. Kazancoglu, D. Yuksel, and M. D. Sezer, "A Green Dual-Channel Closed-Loop Supply Chain Network Design Model," *Journal of Cleaner Production*, vol. 332, no. 3, p. 130062, Dec. 2021.
[18] L. Moretti, M. Milani, G. G. Lozza et al., "A Detailed MILP Formulation for the Optimal Design of Advanced Biofuel Supply Chains," *Renewable Energy*, vol. 171, pp. 159-175, Jun. 2021.
[19] A. Kumar and K. Kumar, "An Uncertain Sustainable Supply Chain Network Design for Regulating Greenhouse Gas Emission and Supply Chain Cost," *Cleaner Logistics and Supply Chain*, vol. 10, no. 6, p. 100142, Jan. 2024.
[20] S. Duffuaa, M. Idris, A. Kolus et al., "A Mathematical Model for Optimal Turnaround Maintenance Planning and Scheduling for a Network of Plants in Process Industry Supply Chain," *Computers & Chemical Engineering*, vol. 180, p. 108477, Oct. 2023.
[21] S. Khalifehzadeh, M. Seifbarghy, and B. Naderi, "A Four-Echelon Supply Chain Network Design with Shortage: Mathematical Modeling and Solution Methods," *Journal of Manufacturing Systems*, vol. 35, pp. 164-175, Apr. 2015.
[22] M. Sheibani and S. Niroomand, "An Optimization Model for Sustainable Multi-Product Multi-Echelon Supply Chain Networks with U-Shaped Assembly Line Balancing Under Uncertainty," *Supply Chain Analytics*, vol. 5, no. 4, p. 100057, Dec. 2023.
[23] J. Ji and N. Chiadamrong, "Hybrid Analytical and Simulation Optimization Approach for Production and Distribution Supply Chain Planning," *Asia-Pacific Journal of Science and Technology*, vol. 24, no. 3, pp. 1-18, Jul. 2019.
[24] S. Martins, P. Amorim, G. Figueira et al., "An Optimization-Simulation Approach to the Network Redesign Problem of Pharmaceutical Wholesalers," *Computers & Industrial Engineering*, vol. 106, no. 1, pp. 315-328, Apr. 2017.
[25] F. Aqlan and S. S. Lam, "Supply Chain Optimization under Risk and Uncertainty: A Case Study for High-End Server Manufacturing," *Computers & Industrial Engineering*, vol. 93, pp. 78-87, Mar. 2016.
[26] M. De Keizer, R. Haijema, J. M. Bloemhof et al., "Hybrid Optimization and Simulation to Design a Logistics Network for Distributing Perishable Products," *Computers & Industrial Engineering*, vol. 88, pp. 26-38, Oct. 2015.

Tran Thi Uyen Linh 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. Her 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 Warwick University and Ph.D. in Manufacturing Engineering and Operational Management from the University of Nottingham, England. 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.

Nathridee Suppakitjarak is an Assistant Professor at the Faculty of Commerce and Accountancy, Chulalongkorn University, Thailand where she teaches monetary banking and computer applications in finance. She received an MBA and Ph.D. in Banking and Finance from Birmingham University, England. Her research interest includes the impact of microstructure innovations in emerging stock markets and determinants of export behavior in service firms. Her recent paper has appeared in the Service Industries Journal and the International Journal of Logistics Systems and Management.

Somrote Komolavanij received his Ph.D. in Industrial Engineering from the University of Texas at Arlington in 1995. Currently, he is an Associate Professor at Panyapiwat Institute of Management. Previously, he served as an Associate Professor at the School of Management Technology (MT), Sirindhorn International Institute of Technology, Thammasat University. His research interests include quality management, logistics and supply chain management, optimization, engineering economy, service sciences, and industrial agglomeration and development.

A Literature Review of Steering Angle Prediction Algorithms for Autonomous Cars

Shang Shi[1] and Jian Qu[2*]

[1,2]Faculty of Engineering and Technology, Panyapiwat Institute of Management, Nonthaburi, Thailand
E-mail: 6572100022@stu.pim.ac.th, jianqu@pim.ac.th[*]

Received: February 22, 2024 / Revised: March 30, 2024 / Accepted: June 25, 2024

***Abstract*—Road tracking is a critical requirement for the development of autonomous cars. It requires the car to continuously navigate within the designated driving area to avoid any deviation. The computation of steering angles is an essential aspect of achieving autonomous driving. Autonomous steering angle techniques, which are essential for enabling road tracking in autonomous cars, are comprehensively reviewed in this paper. Autonomous steering techniques, mainly involving computer vision methods and end-to-end deep learning approaches, are currently receiving considerable attention. The primary objective of this paper is to identify and reimplement state-of-the-art models in end-to-end deep learning approaches within practical scenarios. We carry out a performance evaluation of each model utilizing real-world tests using scale model cars. Furthermore, we offer perspectives on potential avenues for future research and applications. These may include adaptive modifications to dynamic road conditions, the creation of more effective real-time decision-making algorithms, or the investigation of applications in intricate traffic situations.**

***Index Terms*—End-to-End, Neural Network, Road Tracking, Steering Predicate**

I. Introduction

In the domain of road tracking, the calculation of steering angles is a crucial task in using sensor data to steer the car. Typically, LiDAR or radar sensors employ the Euclidean method for calculating steering angles, which is primarily suitable for gradual and smooth turns. Nevertheless, reliance on these sensors may lead to concerns such as system redundancy, cumulative error, and increased cost, particularly in multi-sensor fusion setups.

Therefore, calculating steering angles via camera for road-tracking is a promising route for the advancement of autonomous cars. This paper reviews the process of utilizing camera-derived external information to determine steering angles, enabling precise road tracking and vehicle control. These methods fall into two categories: Traditional computer vision techniques and end-to-end deep learning approaches.

Traditional computer vision methods can be effective but often require high computational requirements, which can impact the real-time performance of autonomous driving systems. On the other hand, the evolution of deep learning suggests that end-to-end methods show promise in overcoming these challenges. Although numerous end-to-end deep learning models have been proposed, some are based on simulations, others heavily rely on specific datasets, and some are tested solely on scale model cars. The evaluation of these models is still an open concern, which creates uncertainty regarding the direction for developing new models.

The purpose of this paper is to evaluate the performance of existing end-to-end deep learning models by replicating and rigorously assessing them in real-world scenarios. This will be achieved through testing on-scale model cars and employing consistent evaluation criteria. Our purpose in focusing on these challenges is to provide clarity on the effectiveness of these models in real-world applications and to bridge the gap between simulated or dataset performance and real-world scenarios.

II. Literature Review

Traditional vision and robotics techniques are often challenged by noise and variability in autonomous navigation. In contrast, artificial neural networks have shown exceptional performance and adaptability in noisy and variable domains.

A. End-to-End Deep Learning

Pomerleau and Dean introduces ALVINN, an autonomous navigation system driven by a neural network. The article explains the design and training methodology of ALVINN, which is a three-layer backpropagation network. The system is specifically designed for road tracking and uses road images captured by a camera and a laser range finder to navigate the car [1].

The effectiveness of ALVINN has been demonstrated through training on simulated road images. This showcased commendable performance, which was further validated on an autonomous navigation test vehicle at Carnegie Mellon University. This experiment serves as an early illustration of the ability of the ALVINN network to navigate competently on real roads in specific scenarios, and it marks a pioneering effort in the use of neural networks for the navigation of autonomous cars.

The ALVINN technique has demonstrated its effectiveness in driving scenarios with few obstacles and simplicity. This is even though it is a shallow network that uses only pixel inputs to anticipate actions. This performance highlights the potential of neural networks to support autonomous navigation. Chen et al. employed the deep convolutional neural network structure of AlexNet to correlate image features with perceptual metrics. They employed the driving game TORCS, gathering 484,815 training images and conducting model training and testing within the TORCS environment. The results demonstrate the adeptness of the model in the virtual environment of TORCS, which facilitates automated driving and provides results for visual representation [2].

It is important to note that although the model performs well in the virtual environment of TORCS, its effectiveness may be affected by differences in graphical outputs when applied to real driving video data. These variations could potentially impact the adaptability and performance of the model in real-world scenarios.

Bojarski et al. describe an end-to-end learning approach for autonomous car control. The paper outlines the training process of a convolutional neural network that translates pixel inputs from a single forward-facing camera into steering commands. The network architecture uses standard convolution-pooling layers to extract features and three fully connected layers to establish a control strategy [3].

During simulated environmental tests that covered up to 100 miles and lasted 3 hours, the network demonstrated stable driving capabilities. It operated autonomously around 90% of the time. Real-world road tests showed that the autonomous performance of the network reached 98% without any human intervention. This end-to-end approach demonstrated efficiency, requiring minimal manual driving training data to navigate conventional roads, motorways, car parks, and even unpaved roads.

However, it is important to note that the model lacks sufficient explanation of its internal knowledge representation and reasoning process. Furthermore, its ability to generalize and handle unseen data needs further validation. The performance of the model in both simulated and real-world environments has not been comprehensively evaluated, especially in complex real-world driving scenarios where it is still difficult to determine the performance of the model. These uncertainties call for more extensive research and validation efforts.

Bojarski et al. investigate the image recognition capabilities of PilotNet for automotive steering control. The paper presents a methodology for the identification of “salient objects” in images, which are crucial in influencing the steering decisions made by PilotNet. This innovative approach sheds light on the decision-making process of PilotNet [4].

The results suggest that PilotNet can effectively identify important road elements, including lane markings, road edges, and other vehicles. Notably, it can also detect more subtle details such as roadside bushes, which are difficult to capture using traditional methods. This method offers a simple yet effective way to gain insight into the learning mechanism of the end-to-end autonomous car model. It is a significant step towards interpreting the functionality of the model and building trust. However, it is important to note that the training dataset used in the paper was relatively limited, which may have restricted the range of driving scenarios covered. This limitation suggests that model learning is limited to the specifics of the provided data, which may limit performance in more diverse and extensive driving scenarios. As a result, further research and validation is needed to determine the ability of the model to generalize across different scenarios and to adapt effectively to a wider range of driving conditions.

This proposal by Rausch et al. propose using a convolutional neural network as a control strategy. This involves directly correlating monocular camera inputs to steering control outputs for comprehensive, End-to-end control. To gather marker data, they employed vehicle simulation software and conducted manual driving learning to document input images alongside corresponding steering angle outputs for network training. The paper presents a four-layer convolutional neural network structure designed for a specific purpose. Various optimization algorithms were evaluated for their training efficacy. The findings indicate that the network has robust generalization capabilities, as it demonstrated proficiency with unseen data and the ability to emulate human driving behaviors [5].

However, it is important to acknowledge that the simplicity of the model design, limited to four layers, may limit its capacity and hinder the acquisition of complex mapping relationships. Additionally, the reliance on simulation-based training without real vehicle testing data hampers the direct assessment of the real-world control effectiveness of the model. This gap raises concerns about the adaptability and performance of the model in authentic road scenarios. Further research and validation are necessary to address these uncertainties.

Do et al. present a real-time autonomous car navigation system that operates on a Raspberry Pi single-board computer, using deep convolutional neural networks. The system is designed for low cost and utilizes a Raspberry Pi, a front-facing camera, and a remote-controlled vehicle. The team collected driving data and augmented it to acquire a substantial training dataset. The proposed end-to-end deep neural network consists of a 9-layer architecture. It includes 5 convolutional layers for feature extraction and 4 fully connected layers for learning steering control tasks. This model predicts steering angles directly from raw images, enabling real-time autonomous driving functionality [6].

However, the limited computational power of the Raspberry Pi introduces significant latency in the real-time reasoning process. This delay could potentially affect the responsiveness of the system, particularly in driving scenarios that require rapid decision-making. To improve real-time performance, optimizing the model structure or transitioning to a higher-performance computing platform could be crucial. Furthermore, the hardware limitations of the Raspberry Pi may restrict the handling of more complex models or larger datasets, which could potentially impact the accuracy of the model and its ability to generalize. Addressing these computational power constraints is a critical consideration for improving system performance.

Chen et al. adopt a simpler architecture consisting of a 3-layer convolutional and a 2-layer fully connected neural network for an end-to-end mapping learning task, mapping raw images to steering angles. The comma.ai dataset was used for training and testing, and the model demonstrated proficiency in generating accurate steering control commands for vehicle navigation [7].

However, there is a concern about the potential for overfitting of the data, which could have an impact on the ability of the model to generalize across different scenarios. Overfitting occurs when the model excessively tailors itself to specific features and noise within the training data, resulting in inadequate adaptation to new and distinct situations not encountered during training. This limitation may result in suboptimal model performance in real-world applications, especially when dealing with new and unobserved scenarios.

Eraqi et al. propose a novel architecture, the Convolutional Long Short-Term Memory Recurrent Neural Network (C-LSTM), designed for End-to-end learning of autonomous car control direction. The C-LSTM framework combines a Convolutional Neural Network (CNN) for visual feature extraction with a Long Short-Term Memory network (LSTM) to capture both visual information and dynamic temporal dependencies. This design allows the model to understand both static and dynamic dependencies that are essential for driving control [8].

The approach redefines the problem of controlling direction regression as a deep classification issue. This is achieved by establishing topological relationships among discrete classification outputs, which enhances the regression objective. The innovation lies in transforming the regression problem into a classification one and refining the accuracy and resilience of the regression objective through optimized relationships between classification outputs.

The model uses a hybrid CNN and LSTM architecture to efficiently extract significant features from image data and learn temporal dynamics, enhancing its ability to predict and control driving directions. This approach is expected to improve the performance and robustness of control systems used in autonomous cars.

Jhung et al. [9] introduce an end-to-end steering controller that uses Convolutional Neural Networks (CNN) for autonomous cars. They demonstrate that this approach enhances driving performance compared to traditional CNN-based methods. The paper explains the use of a neural network called DAVE-2SKY, which was pre-trained using camera-acquired vehicle images. Subsequently, the network is enabled to infer steering angles for lateral control in autonomous cars through augmented closed-loop training. The paper utilizes the PreScan simulator and the Caffe deep learning framework for training within a Software-In-the-Loop (SIL) simulation environment. To validate the proposed end-to-end controller, experiments involve implementing the autonomous car on a DRIVE™ PX2 computer, assessing system performance through simulations and real road tests. The results suggest that the CNN-based end-to-end controller can provide robust steering control, even under partially observable road conditions. This indicates the feasibility of an autonomous vehicle controlled entirely by such a steering controller. However, the paper lacks sufficient discussion on the scalability of the method and its practicality in large-scale real-world scenarios. The article does not extensively address crucial aspects such as computational efficiency and real-time performance in authentic road conditions, which are pivotal for deploying the method on real roads. Conducting more comprehensive experiments and research could provide insights into the applicability of the method and its performance in real-world environments.

Sharma, Tewolde, and kwon implemented an end-to-end learning approach for lateral position control in automated vehicle driving a critical aspect of autonomous driving. The model was tested on the e-road training track and demonstrated the ability to traverse a complete circle, indicating that it had learned human driving patterns. In evaluations on an unseen single-lane track, the model maintained lane position for approximately 89.02% of the time,

demonstrating strong generalization. When navigating the multi-lane CG track 2, the model completed the loop with only two-lane changes, achieving 96.62% autonomy [10].

The main conclusion of the paper is that the model closely mirrors the human driving model and is highly adaptable in different environments. However, it is important to note that the model only addresses lateral control and not longitudinal control. Despite its exceptional performance in simulation environments, its safety and performance in real driving conditions remain unconfirmed. Therefore, it is imperative to conduct additional field tests and validations to ensure the model's robustness and safety in various real-world environments.

Yang et al. employ a common CNN+LSTM structure within an end-to-end multitasking framework for vehicle control prediction. The model architecture is enhanced by utilizing multimodal multitask learning to improve prediction accuracy. This involves integrating visual features captured by the forward-looking camera and feedback vehicle speed inputs from the initial 10-time steps as inputs to the model. The model simultaneously employs multiple fully connected layers to predict steering wheel angle and vehicle speed values for the following moment [11]. This approach combines visual information with vehicle speed data and uses a multi-task learning strategy to improve predictions of vehicle behavior. Although the structure of the model is not particularly innovative, its ability to combine different data sources and handle multiple prediction tasks is promising. This approach has the potential to enhance prediction performance for vehicle control, demonstrating improved robustness and accuracy in real-world applications.

Smolyakov utilized the CarND Udacity simulator to collect data and simulate vehicle driving, assembling a dataset consisting of left, center and right camera images along with steering angle data. The researcher explored models with limited parameters by introducing two convolutional neural network structures, each with a modest number of parameters suitable for embedded system applications. The experimental findings demonstrate that neural networks with only 26,000 parameters can achieve a commendable 78.5% accuracy in predicting steering angles [12].

This research showcases innovative methodology by using simulators to quickly generate training data and explore the potential of compact models for steering angle prediction. However, to ensure the prediction accuracy and real-world applicability of the model in autonomous driving scenarios, it is necessary to further validate its performance using genuine road conditions. Simulated data may not capture the full range of variations and complexities encountered on real roads, which could better illustrate the robustness and practicality of the model. Bechtel et al. developed DeepPicar, an autonomous driving testbed that uses a Raspberry Pi as the computing platform and employs the same CNN model as the NVIDIA DAVE-2 project for real-time control. However, the study does not include comparative performance tests against high-performance GPU platforms, which limits the depth of its comparative insights [13].

The paper discusses the challenges of implementing end-to-end deep learning for real-time applications in autonomous driving and offers valuable conclusions. However, it highlights the need for further optimization of models and algorithms, especially in enhancing efficiency and real-time performance when computational resources are limited, such as with low-cost computing platforms like the Raspberry Pi. Optimization techniques could include model compression, quantization, or tailored adaptations for embedded platforms to improve real-time and overall performance.

DeepPicar is a promising low-cost autonomous driving testbed, and this research provides valuable insights into real-time deep learning applications in autonomous driving. However, there is still room for further improvement and optimization to enhance its effectiveness and performance.

Sharma et al. applied two separate models for steering and speed control, aiming to improve performance by running them in parallel. The steering model had eight convolutional layers, while the speed model had five. To allow for parallel computation of model predictions, they used a multithreading mechanism [14].

Using the open-source TORCS simulation environment, both models demonstrated 100% autopilot capability on the e-road and CG-track 2, the roads where training data was gathered. This highlights the effectiveness of the training. However, the models showed subpar results on other roads, indicating limitations in their generalization ability. Improving the generalization of the models may require optimizing the network structure and collecting more diverse experimental data.

The researcher discussed the challenges of implementing multi-threading and its potential impact on real-time performance. They noted that thread switching and data transfer could cause delays that affect system responsiveness. Additionally, the researcher pointed out that the independence of steering and speed control tasks may limit GPU resource utilization.

Wang et al. proposed a hybrid architecture that combines CNN and LSTM networks to improve steering angle and speed predictions. By incorporating spatial-temporal and spatial information from image

sequences and speed feedback inputs, this approach resulted in more accurate predictions compared to the CNN single-task network [15].

The study also compared traditional LSTM networks with the proposed state-passing LSTM in terms of prediction time and effectiveness. The results indicate that integrating the auxiliary task with the state-passing LSTM significantly reduces time overhead while maintaining a high level of prediction accuracy.

The paper conducted online tests in both the GTAV simulation environment and real-road scenarios and observed a decrease in intervention frequency from 6 to 2 instances. This implies an improvement in the quality of actual control, suggesting that the model has the potential to enhance automatic control and reduce human intervention in real-world scenarios.

The research combined CNN and LSTM networks using improved models and training methods, resulting in a significant improvement in the accuracy of steering angle and speed predictions. The validation of these improvements was carried out through simulated environments and real-road tests, indicating the potential application of the methodology in automated driving.

Shuyang et al. conducted a study to explore the prediction of steering wheel angle in autonomous cars using two deep learning models. The first model combined a 3D convolutional layer with residual concatenation and an LSTM network to capture spatial and temporal relationships within visual features. This model extracted visual features using 3D convolution and captured sequential relationships through an LSTM network. During model training, transfer learning was employed by keeping the parameters of the initial 45 layers of ResNet50 fixed and feeding their output through a fully connected layer to predict the steering wheel angle [16].

The evaluation of the Udacity dataset resulted in RMSE values of 0.1123 and 0.0709 for the two models on the test set, respectively. Additionally, the models achieved 10th and 4th positions in the Udacity Challenge.

This paper utilizes various deep learning architectures, specifically combining 3D convolutional layers with LSTM networks, to improve the accuracy of steering wheel angle predictions by capturing spatial and temporal relationships. The models also demonstrate promising performance on both the Udacity dataset and the Challenge, indicating potential applicability in autonomous driving scenarios.

Lee et al. proposed a novel method for capturing spatiotemporal elements in images by combining CNN and LSTM networks. The method takes into account the influence of time-series data on driving judgments. The authors collected extensive real driving data using a driving simulator, which significantly enriched the training dataset of the network. As a result, the proposed method accurately predicts steering wheel angles, even on challenging S-curves, demonstrating its efficacy in real driving scenarios [17].

However, although simulated environments are used for data collection and training, there are still discrepancies between simulated and real driving environments. This difference may limit the model's ability to adapt to the diverse variations and intricacies encountered on actual roads. Therefore, further validation is required to assess the performance and generalizability of the model in real driving environments.

Although the model excels in simulated environments, additional data acquisition and fine-tuning may be necessary to enhance its generalizability and adaptability for real-world operations.

Kumar et al. propose a fully end-to-end deep learning model for steering wheel angle estimation. The model uses direct image inputs from the forward-looking camera to predict corresponding steering angles without intermediate feature extraction. The authors employ Dropout regularization within the CNN model to counter overfitting, alongside meticulous hyper-parameter tuning. These efforts have resulted in an accuracy of 98.6%, which is higher than previous benchmarks in the literature.

The innovation of the paper lies in its comprehensive end-to-end methodology, which eliminates the need for intermediate feature extraction or processing steps in steering angle estimation. Furthermore, the use of Dropout regularization and hyper-parameter tuning significantly improves model performance. The achieved accuracy highlights the potential application of this approach in advancing autonomous driving technology.

Mishra et al. [17] utilized a Convolutional Neural Network (CNN) to estimate steering angles, harnessing the power of deep learning and CNNs to enable automated learning of driving skills. The CNN model learned and predicted steering commands directly from raw pixel data captured by a single front-facing camera, utilizing the power of deep learning and CNNs to enable automated learning of driving skills. Data was collected using the Udacity autonomous car simulator, which provided a virtual environment to simulate diverse driving scenarios and gather data for training and validating the model's performance.

Sokipriala and Jonah utilised pre-trained deep convolutional neural networks to predict steering wheel control angles in autonomous cars. The paper employed transfer learning methods and utilised the VGG16 model to extract image features from a comprehensive image classification dataset such as ImageNet. This technique allows for resource sharing

and reduces the number of training parameters, thereby enhancing model performance [18].

The research presents a regression problem for predicting the steering wheel control angle. Mean Square Error (MSE) is used as the loss function for supervised learning. To adapt this approach for autonomous driving, the pre-trained model is further fine-tuned, resulting in more precise predictions of the steering wheel control angle. This method refines the model to better suit the specific requirements of autonomous driving.

Showrov et al. conducted an investigation involving collecting vehicle front images paired with corresponding steering angle data. They compared the performance of deep learning models such as VGG16, ResNet-152, DenseNet-201, and Nvidia. The study revealed that the Nvidia model outperformed the other pre-trained models, showcasing a Mean Square Error (MSE) value of 0.3521 [19].

The Nvidia model performed well in the trial. However, the paper does not mention any performance evaluation or validation in an autonomous driving system used in the real world. Therefore, there is a gap in validating the feasibility and effectiveness of this method in real-world applications. To implement this model in practical driving scenarios, further field tests and validation are imperative to ensure its reliability and accuracy within authentic autonomous driving systems.

Podbucki et al. presented a method for automated trolley driving using a Jetson Nano microcomputer and a ResNet18 convolutional neural network. The experimental outcomes showed that this method enabled stable cart movement along a predefined track, with improved operation as the experiments progressed. The implementation used open-source frameworks such as PyTorch and TensorRT for real-time inference, meeting the real-time demands of cart autopilot [20].

However, the paper lacks crucial algorithm evaluation metrics, such as quantitative measurements like success rate and tracking error. These metrics are pivotal in evaluating algorithm performance, enabling the assessment of accuracy and reliability in ensuring stable travel along a predefined track. The absence of these metrics hinders a comprehensive evaluation and comparison of the performance of the algorithm with other methodologies. Therefore, a comprehensive evaluation of the usefulness and effectiveness of the model requires that its performance metrics be thoroughly assessed and reported.

Khidhir et al. conducted a comparative analysis of three deep learning models for end-to-end autonomous driving. The study highlights the advantages of ResNet18 and DenseNet121 in steering angle prediction and driving performance. The results indicate that ResNet18 is more accurate, while DenseNet121 is more consistent across diverse driving tracks [22].

However, the dataset used in the study was relatively limited, consisting of only 681 images. The validation process and presentation of results may need further refinement, and a larger dataset coupled with a comprehensive validation methodology could better illustrate the performance and stability of the model. Expanding the dataset and refining validation techniques would contribute to a more robust demonstration of the model's capabilities.

Hassan et al. developed a lightweight Convolutional Neural Network (CNN) that predicts vehicle steering angles using raw image inputs. The model was trained and evaluated using data from the CARLA simulator. The experimental results showed that this lightweight model achieved comparable steering angle prediction performance to Nvidia's PilotNet, despite having only a quarter of the parameters. This highlights the ability of lightweight models to deliver satisfactory steering angle predictions with a significantly smaller parameter size, which could be particularly beneficial in resource-constrained environments or situations demanding compact model sizes [23].

Ding et al. introduced a deep learning model that integrates the CBAM attention mechanism into the ResNet18 structure, enhancing its ability to concentrate on specific objects and extract features effectively. This model accomplishes road tracking and obstacle avoidance tasks. The authors improved the model's performance through hyperparameter optimization and thoughtful parameter combinations. Experimental findings demonstrate impressive scores. The study reports that the model achieved 98% accuracy in a training setting and 72% in a mildly noisy environment, significantly higher than the 32% benchmark set by existing methods [24].

However, the study does not discuss potential challenges and limitations that may arise in practical applications. Addressing these aspects could provide valuable insights into the adaptability and performance of the model under real-world conditions. Understanding the constraints and potential obstacles in practical implementation is crucial when assessing the model's viability beyond controlled environments.

Zihao et al. developed a self-sufficient smart car system that uses a single-camera sensor and the Jetson Nano as its central processing unit. They introduced a novel neural network structure called MT-ResNet26 to achieve autonomous driving. This structure leverages ResNet residual blocks to enhance lower-layer feature extraction while integrating Batch Normalization (BN) layers for optimized training. This architecture aims to avoid the data demands posed by excessively deep or wide networks, prioritizing generalization ability and robustness [25].

However, the effectiveness of the model was assessed solely using qualitative scoring systems, lacking quantitative metrics for comprehensive analysis of the results. Incorporating quantitative measures can allow for a more extensive evaluation, providing precise insights into how the model performs, where strengths lie, and where potential improvements exist.

Ding et al. propose an Advanced Driver Assistance System (ADS) to execute five autonomous driving tasks using a single-camera sensor. The system comprises two neural network models: an end-to-end model named Efficientnet_b0-SA-RE for overall functionality and a safety assistance model, Efficientnet_b0-CA, designed for specific safety measures. These models collaborate to achieve tasks related to route tracking, turn sign recognition, lane changing, obstacle stopping, and traffic light recognition [26].

However, the proposal lacks an in-depth analysis comparing different model fusion schemes. Such an analysis would be valuable in determining the most effective approach for integrating the two models and optimizing their collaborative performance. Evaluating and contrasting alternative fusion methods could improve our understanding of how different techniques impact the overall efficiency, accuracy, and adaptability of the system across various driving scenarios.

Ding et al. implemented road tracking and traffic sign recognition and designed three real simulated environments with three sets of experiments. Their model ResNet34_B32 is capable of road tracking in both trained and untrained environments [27].

Li et al. achieved multi-task autonomous driving by adjusting the model's network structure. However, they found that the trained neural network model performed poorly in untrained scenarios after implementing multi-task autonomous driving. Therefore, Li proposed improving the model's transfer efficiency in new scenarios through transfer learning [28].

Shi et al. proposed the RDNet18-CA model, which not only achieves road tracking but also completes multiple tasks. Additionally, they introduced a new loss function LiSHTL-S to enhance the model's performance in untrained scenarios [29].

III. Methodology

This section mainly introduces the design of the evaluation method for the model, which is divided into the construction of the model car, road tracking design, data collection, and new evaluation methods. The experimental design aims to build a Jetbot model car to collect the dataset. Subsequently, each test model is used to train the dataset to obtain the model. Then, these models are deployed onto the Jetbot model car and placed on the track for testing. Evaluation data is collected, and assessment criteria are applied for calculation. Finally, data analysis is conducted to assess the performance of each model.

A. The Jetbot Platform

This paper aims to leverage the components of the open-source JetBot scale model car provided by Nvidia, including the front camera, Jetson Nano motherboard, JetBot driver board, and DC motor. As shown in Fig. 1. The Jetson Nano is chosen as the embedded computing platform due to its quad-core Cortex-A57 processor, 128-core Maxwell GPU, and 4GB LPDDR memory, which offer substantial AI computing power and facilitate the execution of various AI algorithms. This paper evaluates depth models that will be deployed on the Jetson Nano for real-time inference, demonstrating the capability of the system to perform complex AI tasks.

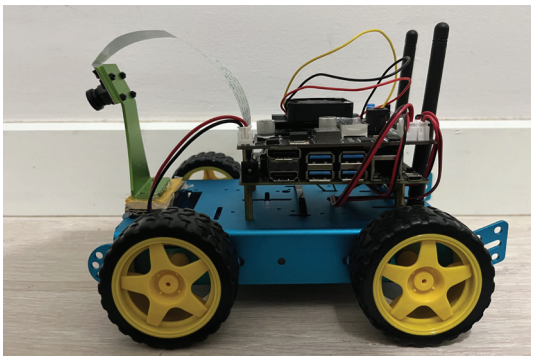


Fig. 1. The scale model car

B. Data Collection and Parameterization

To assess the performance of each model in real-world scenarios, we designed the track illustrated in Fig. 2. The track includes straight lines, curves, T-junctions, intersections, and various obstacles and traffic signs. We collected tracking data along the outer track while using the inner track as an unknown scenario to evaluate the adaptability of the model to unseen environments. For detailed information on data collection, please refer to Table I.

TABLE I
COMPOSITION OF THE DATASET

Dataset	Numbers
Road Tracking	600
Turn Right/Left	500
Barrier Avoidance Cars	900
Total	**2000**

Images of the track are captured through the camera of a model car, where each image contains X and Y coordinate labels that represent the location of the vehicle. Our goal is to train a neural network model to accurately predict the values of the X and Y

coordinates from these images. This prediction can provide us with information about the position of the vehicle on the track.

Barrier Avoidance Cars (BAC) are vehicles designed to be used on tracks to provide obstacle avoidance support for self-driving cars. These vehicles are typically placed on specific test tracks to simulate various obstacles or challenges encountered during real-world road travel. As the self-driving vehicle travels on the test track, BACs are placed in its path to act as potential obstacles. The self-driving vehicle uses its camera system to detect the BACs and successfully avoids these obstacles by performing appropriate steering maneuvers.

Before training the data, we need to preprocess the data to improve the generalization ability and performance of the model. First, we can process the image using a random horizontal flip technique where the flip probability is set to 0.5, which helps the model to learn more features from different angles. Next, we can apply color jitter to the image, where the jitter magnitude for each value is set to 0.3, which helps to increase the diversity of the data and makes the model more robust to lighting and color changes. We then resized the image to 224 × 224 pixels, the input size for many standard convolutional neural network models. We then converted the image to a tensor (tensor) format to be processed in the neural network. Finally, we normalize the image by scaling the pixel values to a specific range to ensure stability and convergence of the model training.

To test the performance of the model, we separated the test track from the track for collecting data. We designed the track shown in Fig. 3 as the test track. This test track was carefully designed to introduce scenarios not previously encountered by the model to assess the robustness and generalization ability of the model more fully.

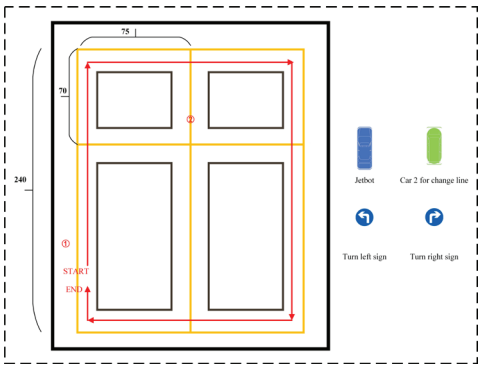


Fig. 2. Map track

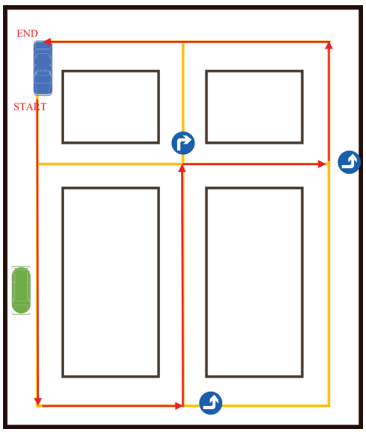


Fig. 3. Test track

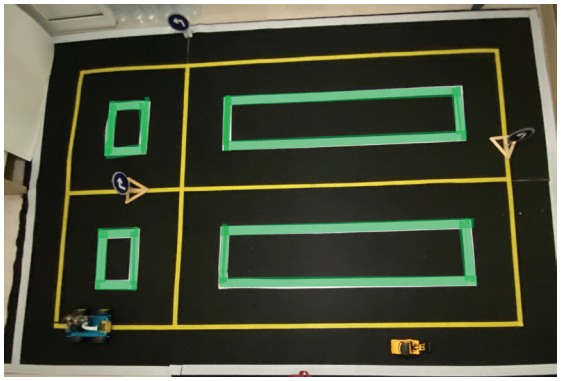


Fig. 4. Actual test track

In this experiment, the data were trained through the colab platform, and the experimental parameters were set to the epoch of 80, batch size of 8, Adam as an optimizer, and learning rate of 0.0001.

C. *Criteria for Assessment*

To comprehensively evaluate each model's performance on actual roads, we introduce a novel evaluation metric known as the Road Integrity Comprehensive Score (RICS). RICS combines two critical factors: Road Distance Completion (RDC) and Road Sign Completion (RSC). Road Distance Completion is the ratio between the distance the vehicle covers and the total route distance. Road Sign Completion is the ratio between accurately detected road traffic signs and the total number of signs.

$$\text{RICS} = \frac{1}{\text{N}_c}\sum_{i=1}^{N_c}\left(RDC_i \times RSC_i\right) \quad (1)$$

N_c : Number of Test Lap
i : -th route
RDC_i: The percentage of completion of the route
RSC_i : The infraction penalty for a route

This method provides a more detailed and comprehensive evaluation approach by considering both the coverage distance of the vehicle and the accurate identification of traffic signs. It not only accounts for the route coverage of the car but also assesses the accuracy of the model in identifying crucial traffic information. Therefore, the Road Integrity Comprehensive Score offers a more precise measure of the real-world performance and reliability of each model in actual road environments.

IV. Result

Models with fewer layers, like VGG and AlexNet, have higher sizes, as Table II demonstrates. When these models are put on embedded devices, latency concerns may arise. Although SqueezeNet has a smaller model size, its corresponding loss values are higher, resulting in suboptimal performance on the model car. When comparing ResNet18, ResNet34, and ResNet50, an layer increase directly impacts model size, while the change in loss values remains marginal. The MT-26 model is an extension of ResNet18 to 26 layers.

TABLE II
Results of Model Training

Model	Epoch	Train loss	Test Loss	Model Size(MB)
Alexnet	10	0.231967	0.224831	217
Vgg16	1	1.876685	0.200343	512
Squeezenet	37	0.228815	0.221599	2.77
Mobilenet	57	0.003410	0.003572	8.73
Densenet201	46	0.002949	0.001997	70.30
Resnet18	52	0.002777	0.001751	42.70
Resnet34	60	0.002566	0.002466	81.30
Resnet50	62	0.003016	0.001323	89.90
MT-26	79	0.003338	0.001247	90.60
Efficientnet _b0_sa	73	0.001838	0.001106	15.90
Resnet-CBAM	75	0.001598	0.001516	43.00

Despite having the largest model size, it has the lowest loss value among the ResNet series. On the other hand, ResNet-CBAM is an enhancement of ResNet18 with an attention mechanism that prioritizes significant image features during training. This yields superior performance without significantly increasing model size, with only a 0.3MB increase.

The real-world performance scores of several models are shown in Table III. It is observed that shallow networks such as AlexNet, VGG, and SqueezeNet face difficulties in achieving real-time corner predictions on embedded systems, which hinders their ability to control the model car for road tracking and sign recognition. On the other hand, MobileNet, a lightweight network at 8.73MB, achieves an RICS score of 0.81, ranking second among all tested models and showcasing commendable practical applicability. Despite its larger model size compared to MobileNet, MT-26 is ranked first. It achieves real-time execution on embedded systems with an RICS score of 1, effectively controlling the model car for road tracking and sign recognition. Other models achieve an average RICS score of 0.57, indicating their operability on model cars.

TABLE III
Tracking Test Results

Model	RICS	RDC	RSC
Alexnet	0.08	0.31	0.25
Vgg16	0.08	0.31	0.25
Squeezenet	0.00	0.31	0.00
Mobilenet	0.81	0.81	1.00
Densenet201	0.61	0.81	0.75
Resnet18	0.54	0.72	0.75
Resnet34	0.54	0.72	0.75
Resnet50	0.08	0.31	0.25
MT-26	1.00	1.00	1.00
Resnet-CBAM	0.54	0.72	0.75
Efficientnet _b0_ca	0.61	0.81	0.75
Ideal	1.00	1.00	1.00

Finally, we combine Tables II and III to analyze the performance of each model in this task. Table II shows that AlexNet, VGG16, and SqueezeNet achieved their optimal models at epoch 10, epoch 1, and epoch 31, respectively. However, their loss values are relatively high compared to other models. Although they can be deployed and run on Jetson Nano, through Table III, we found their final RICS scores to be low, all at 0.08, indicating that they failed to learn the dataset information thoroughly, and their models are too large, making running on Jetson Nano challenging.

In contrast, MobileNet and MT-26 achieved their optimal models at epoch 57 and epoch 79, respectively, in Table II. Their loss values have significantly decreased compared to AlexNet, VGG16, and SqueezeNet. By observing Table III, we can see that the final RICS scores for MobileNet and MT-26 are 0.81 and 1, respectively, indicating that they are better suited for this task. This is because their loss values have decreased significantly during the training process, and the final RICS scores indicate that they have learned the dataset features better. Additionally, their models are relatively lighter, making them more suitable for running in resource-constrained environments, such as Jetson Nano.

V. Conclusion

This paper presents an overview of steering angle prediction algorithms for autonomous driving. Our study involved reproducing and testing existing models in real environments using model cars. We found that models with fewer layers, such as AlexNet, VGG16, and SqueezeNet, could not fully comprehend the information in the training data. As a result, they could not perform autonomous driving tasks in embedded systems. ResNet50 achieved an RICS of only 0.08 in the ResNet series, while ResNet18 and ResNet34 scored 0.54. ResNet-CBAM and EfficientNet_B0_CA exhibited similar performance to the ResNet series. MobileNet, a lightweight network, attained an RICS of 0.81 in the embedded system scale model car, ranking second among all the tested models. The top-performing model was MT-26, achieving an RICS of 1 and ranking first despite having a more significant model size than MobileNet.

Acknowledgment

The first author experimented and drafted the manuscript. The last author guided and advised the experiment and co-drafted the manuscript. The first and last authors each contributed 50% equally to this work. The last author is the corresponding author.

The first author received scholarship support from CPALL for conducting this research in PIM.

References

[1] D. A. Pomerleau, "Alvinn: An Autonomous Land Vehicle in a Neural Network," in *Proc. NIPS*, 1988, pp. 305-313.

[2] C. Chen, A. Seff, A. Kornhauser et al., "Deep Driving: Learning Affordance for Direct Perception in Autonomous Driving," in *Proc. ICCV*, 2015, pp. 2722-2730.

[3] M. Bojarski, D. D. Testa, D. Dworakowski et al. (2016, Apr. 25). *End to End Learning for Self-Driving Cars*. [Online]. Available: https://arxiv.org/abs/1604.07316

[4] M. Bojarski, D. D. Testa, D. Dworakowski et al. (2017, Apr. 25). *Explaining How a Deep Neural Network Trained with End-to-End Larning Steers a Car*. [Online]. Available: https://arxiv.org/abs/1604.07316

[5] V. Rausch, A. Hansen, E. Solowjow et al., "Learning a Deep Neural Net Policy for End-to-End Control of Autonomous Vehicles," in *Proc. ACC*, 2017, pp. 4914-4919.

[6] T. D. Do, M. T. Duong, Q. V. Dang et al., "Real-Time Self-Driving Car Navigation Using Deep Neural Network," in *Proc. GTSD*, 2018, pp. 7-12.

[7] Z. Chen and J. Huang, "End-to-End Learning for Lane Keeping of Self-Driving Cars," in *Proc. IV*, 2017, pp. 1856-1860.

[8] H. M. Eraqi, M. N. Moustafa, and J. Honer. (2017, Oct. 10). *End-to-End Deep Learning for Steering Autonomous Vehicles Considering Temporal Dependencies*. [Online]. Available: https://arxiv.org/abs/1710.03804

[9] J. Jhung, I. Bae, J. Moon et al., "End-to-End Steering Controller with CNN-Based Closed-Loop Feedback for Autonomous Vehicles," in *Proc. IVS*, 2018, pp. 617-622.

[10] S. Sharma, G. Tewolde, and J. Kwon, "Behavioral Cloning for Lateral Motion Control of Autonomous Vehicles Using Deep Learning," in *Proc. EIT*, 2018, pp. 0228-0233.

[11] Z. Yang, Y. Zhang, J. Yu et al., "End-to-End Multi-Modal Multi-Task Vehicle Control for Self-Driving Cars with Visual Perceptions," in *Proc. ICPR*, pp. 2289-2294.

[12] M. V. Smolyakov, A. I. Frolov, V. N. Volkov et al., "Self-Driving Car Steering Angle Prediction Based on Deep Neural Network an Example of CarND Udacity Simulator," in *Proc. AICT*, 2018, pp. 1-5.

[13] M. G. Bechfel, E. McEllhiney, M. Kim et al. "Deep Picar: A Low-Cost Deep Neveal Network-Based Autonomous on Embedded and Real-Time Compution Systems and Applications (RTCSA), 2018, pp. 11-21.

[14] D. Wang, J. Wen, Y. Wang et al., "End-to-End Self-Driving Using Deep Neural Networks with Multi-auxiliary Tasks," *Automot. Innov.*, vol. 2, no. 2, pp. 127-136, May. 2019.

[15] S. Du, H. Guo, and A. Simpson. (2019, Dec. 11). *Self-Driving Car Steering Angle Prediction Based on Image Recognition*. [Online]. Available: https://arxiv.org/abs/1912.05440

[16] M. J. Lee and Y. G. Ha, "Autonomous Driving Control Using End-to-End Deep Learning," in *Proc. BigComp*, 2020, pp. 470-473.

[17] A. Kumar and S. Palaniswamy, "Steering Angle Estimation for Self-Driving Car using Deep Learning," *MLMHA*, vol. 1203, pp. 196-207, Apr. 2020.

[18] A. K. Mishra, "Deep Learning for Steering Angle Prediction in Autonomous Vehicles," *MSEA*, vol. 70, no. 2, pp. 1584-1590, Dec. 2021.

[19] J. Sokipriala, "Prediction of Steering Angle for Autonomous Vehicles Using Pre-Trained Neural Network," *EJETR*, vol. 6, no. 5, pp. 171-176, Aug. 2021.

[20] M. I. H. Showrov, M. R. Islam, M. A. Amin et al., "Comparative Analysis of Steering Angle Prediction for Automated Object Using Deep Neural Network," in *Proc. ICRITO*, 2021, pp. 1-7.

[21] K. Podbucki and T. Marciniak, "Aspects of Autonomous Drive Control Using NVIDIA Jetson Nano Microcomputer," in *Proc. CCSIS*, 2022, pp. 117-120.

[22] Y. G. Khidhir and A. H. Morad, "Comparative Transfer Learning Models for End-to-End Self-Driving Car," *AKEJ*, vol. 18, no. 4, pp. 45-59, Apr. 2022.

[23] I. U. Hassan, H. Zia, H. S. Fatima et al., "A Lightweight Convolutional Neural Network to Predict Steering Angle for Autonomous Driving Using CARLA Simulator," *MSE*, vol. 2022, no. 9, pp. 1-11, Aug. 2022.

[24] S. Ding and J. Qu, "Smart Car with Road Tracking and Obstacle Avoidance Based on Resnet18-CBAM," in *Proc. ICBIR*, 2022, pp. 582-585.

[25] Z. Nie and J. Qu, "Multi-Task Autonomous Driving Based on Improved Convolutional Neural Network and ST Loss in MTS and MOD Modes," *CAST*, vol. 23, no. 3, pp. 10-36, Nov. 2023.

[26] S. Ding and J. Qu, "Research on Multi-Tasking Smart Cars Based on Autonomous Driving Systems," *SNCS*, vol. 4, no. 3, p. 292, Mar. 2023.

[27] S. Ding and J. Qu, "Automatic Driving for Road Tracking and Traffic Sign Recognition," *STA*, vol. 27, no. 4, pp. 343-362, Dec. 2022.

[28] Y. Li and J. Qu. (2024, Apr. 12). *A Novel Neural Network Architecture and Cross-Model Transfer Learning for Multi-Task Autonomous Driving*. [Online]. Available: https://doi.org/10.1108/DTA-08-2022-0307

[29] J. Qu and S. Shi, "Multi-Task in Autonomous Driving through RDNet18-CA with LiSHTL-S Loss Function," *ECTI-CIT*, vol. 18, no. 2, pp. 158-173, Apr. 2024.

Shang Shi is studying for the Master of Engineering Technology, Faculty of Engineering and Technology, Panyapiwat Institute of Management, Thailand. He received a B.B.A from Nanjing Tech University Pujiang Institute, China, in 2022. His research interests are Research direction is artificial intelligence, image processing, and autonomous driving.

Jian Qu is an Assistant Professor at the Faculty of Engineering and Technology, Panyapiwat Institute of Management. He received Ph.D. with Outstanding Performance award from 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, Thammast University, Thailand, in 2010. He has been serving as a house committee for the Thai SUPERAI project since 2020. His research interests are natural language processing, intelligent algorithms, machine learning, machine translation, information retrieval, image processing, and autonomous driving.

A Transfer Learning-based Deep Convolutional Neural Network Approach for White Shrimp Abnormality Classification

Korawit Orkphol[1], Kathawach Satianpakiranakorn[2*], Jidapa Chaihuadjaroen[3], and Tamnuwat Valeeprakhon[4]

[1,2,3,4]Department of Computer Engineering, Faculty of Engineering at Sriracha, Kasetsart University Sriracha Campus, Chonburi, Thailand.

Email: korawit@eng.src.ku.ac.th, kathawach@eng.src.ku.ac.th*, jidapa@eng.src.ku.ac.th, tamnuwat@eng.src.ku.ac.th

Received: August 28, 2024 / Revised: October 10, 2024 / Accepted: October 20, 2024

***Abstract*—Shrimp transportation frequently leads to product damage, necessitating a sorting system to identify and remove compromised shrimp prior to processing. This research aims to develop a transfer learning-based deep convolutional neural network system capable of accurately categorizing shrimp into seven classes: Complete body, crunched head, head loss, head loss with remaining chin, cut tail, torn in half, and total crunched. A dataset comprising 405 color shrimp images, each with dimensions of 1,920 x 1,080 pixels, was augmented using geometric transformations to expand the dataset to 6,480 images. These augmented images were then employed to train four state-of-the-art transfer learning-based models (NasNetLarge, InceptionResNetV2, EfficientNetV2L, ConvNeXtXLarge) from Keras Applications. These models also were subsequently compared to a baseline CNN.**

Results demonstrate that the ConvNeXtXLarge model outperformed the others, achieving the highest accuracy (95%), precision (0.96%), recall (0.95%), and F1-score (0.95%), underscoring its superiority in shrimp damage classification. An analysis of misclassifications revealed potential confusion between certain damage classes, suggesting areas for future refinement to enhance the model's ability to differentiate between similar types of damage.

***Index Terms*—Anomalies Who Le White Shrimp, Deep Learning, Image Classification, Transfer Learning**

I. Introduction

Vannamei shrimp is one of the important export products of Thailand and a major economic commodity. Thailand has the potential to process shrimp into various products to meet market demand. Thailand's main export markets included the United States, Japan, the People's Republic of China, and the Republic of Korea. According to the Thailand Foreign Agricultural Trade Statistics 2023 by the Agricultural Information Center, the Office of Agricultural Economics, Ministry of Agriculture and Cooperatives. In 2023, the income from shrimp products accounted for 2.52% of the value of agricultural export products, bringing in the eighth-highest income of agricultural products and overall products. For fresh, chilled, or frozen shrimp, the export volume was 15,273 metric tons, with an export value of 3,794 million THB, accounting for 58.52% of all shrimp exports [1].

To maintain the freshness of white shrimp before entering the processing process, all steps from catching shrimp to transporting shrimp from farmers to the production line must be done rapidly. The process will start with farmers catching shrimp at nighttime to prevent damage from heat, using a net to scoop shrimp out of the pond. Then, the size sorting staff from the shrimp farm or the factory will sort the sizes. After sorting, the shrimp will be placed in baskets, as shown in Fig. 1, and transported to the factory by a refrigerated truck. However, during transportation, shrimp will be packed in baskets and stacked in layers to allow for large quantities of goods on each transportation trip. In this step, some shrimp on top of the baskets that are stacked on top of each other may be damaged. In addition, shrimp at the bottom of the basket may be crushed by the weight of the shrimp above, which may cause damage. When it arrives at the factory, the shrimp will be sorted by taking the undamaged shrimp to the production line. Damaged shrimp will be sorted into grades for processed snacks and animal feed grades. However, even though the shrimp in the baskets are fresh, some of the shrimp that were damaged during transportation, such as shrimp heads that have fallen off or bodies that have broken, will be damaged. In this step, factory workers will have to take the shrimp out of the baskets and use manual labor to select the damaged shrimps and

send the undamaged shrimps to the processing line. This step will require all manual labor.

Fig. 1. Shrimp containers for transportation

The visual characteristics of shrimp, such as color, texture, size, and shape, are key factors that influence consumer purchase decisions. They are also important indicators for classifying products according to different quality levels. Visual inspection is the most common method for quality control of shrimp before entering the production line. It is a common practice in the seafood industry. However, visual inspection is prone to errors [2], [3]. Repetitive work can intensely decrease human efficiency. In addition, there are problems of rising wages and labor shortages. Therefore, there is a demand to use automated systems instead of human labor.

The most effective technique for solving these issues in the food business is operational automation. To automate all processes; Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL) technologies are applied [4]. Machine learning and Artificial Intelligence (AI) can be used for the transformation of food safety and quality data management [5]. Machine learning and deep learning are applied to automate detection in many shrimp products. For peeled shrimp products, Valeeprakhon [6] proposed a Deep Convolutional Neural Network based on VGG-16 for the classification abnormalities of peeled shrimp. Yu [7] proposed YOLO-4 for classifying peeled and shell-on shrimp detection. For de-heading, back cutting or butterfly cutting and removing the vein, Thanasarn [3] proposed automated recognition of deveined shrimps with linear discriminant analysis and Support Vector Machine (SVM) based on grayscale image parameters. To measure the freshness of shrimp, Zhang [8] proposed deep learning detection of shrimp freshness via smartphone camera picture which is an easy and low-cost method to detect shrimp freshness. Wang [9] proposed a deep learning-based freshness assessment method for chilled red shrimp using a shrimp net, which is a lightweight module. The method was based on a Convolutional Neural Network (CNN) architecture that was trained on a data set of images of frozen red shrimp with different levels of freshness. Prema and Visumathi [10] proposed shrimp Freshness detection based on CNN compared with Hybrid CNN and SVM with DCGAN which is proposed by Radford et al. and Yeh et al for data augmentation and found that Hybrid CNN and SVM with GAN is better than classic CNN model. Most researches on fresh shrimp are related to the shrimp products industry by using image classification with Machine Learning (ML) which is focused on size, freshness, and quality inspection of shrimp products after passing the production line. No research has been found to distinguish shrimp with incomplete conditions before entering the production line.

The use of ML is to select disqualified shrimps before entering the production line. If we can distinguish the types of shrimp that are rejected by shape, for example, shrimp that is severely damaged, such as broken bodies or mushy bodies, should be classified as feed grade. Shrimp with head loss, head loss with only the shrimp chin, or crunch head shrimp can be sorted as shrimp grades higher than feed grade. This will increase the value of shrimp more than all rejected shrimp that were sorted as feed grade. In this research, four top-1 accuracy models with over 80% accuracy from Keras Applications [11], namely NasNetLarge [12], InceptionResNetV2 [13], EfficientNetV2L [14], and ConvNeXtXLarge [15], were selected for testing on a dataset with CNN as the baseline. The objective was to classify shrimps into seven categories: complete body shrimp, crunch head shrimp, shrimp with head loss, head loss shrimp with the remaining chin, cut tail shrimp, torn in half shrimp (upper half and lower half), and total crunched shrimp, as shown in Table I. The experimental results revealed that ConvNeXtXLarge achieved the highest accuracy of 95%.

TABLE I
CLASSES OF SHRIMP DATASETS

Class Label	Type	Status	Example
Complete	Complete body shrimp	Normal	
Busted Head	Crunch head shrimp	Abnormal	
HeadOff	Shrimp with head loss	Abnormal	
Chin	Head loss shrimp with remaining chin	Abnormal	
TailOff	Cut tail shrimp	Abnormal	
Half	Torn-in-a-half shrimp	Abnormal	 
Crushed	Total crunched shrimp	Abnormal	

II. METHODOLOGY

In this research, 405 images of normal and disqualified shrimps, with dimensions of 1,920 x 1,080 pixels, were used. The images were divided into 7 categories:

1. Complete body shrimp (80 images)
2. Crunched head shrimp (56 images)
3. Shrimp with head loss (54 images)
4. Head loss shrimp with remaining chin (56 images)
5. Cut tail shrimp (43 images)
6. Torn in half shrimp (45 images)
7. Total crunched shrimp (71 images)

The images were then subjected to data pre-processing before being used to train a transfer learning model. The pre-processing steps were as follows:

A. Data Preprocessing and Geometric Transformation Augmentation

1) Geometric Transformation Augmentation

Due to the limited number of input images, geometric transformation augmentation was employed. This research explores two geometric transformation techniques, image reflection, and image rotation, to generate a new set of images. In the first step, the original image is rotated by predefined angles: 0 degrees, 45 degrees, 90 degrees, 135 degrees, 180 degrees, 225 degrees, 270 degrees, and 315 degrees. Rotation is a common type of geometric transformation used for image data augmentation [16].

However, the input image is rectangular as shown in Fig. 2 (I_1), which makes it impossible to rotate the image. Therefore, image preprocessing is required before the augmentation process. The preprocessing steps are as follows:

1. Convert the input image to a binary image (I_2).
2. Remove noise from the image using opening and closing operations (I_3).
3. Find the center of the image by contouring the objects in the image, selecting the largest object, cropping the largest object to a square, and finding the center of that square (I_4).
4. Once the center of the image is found, expand the green background to make the image size 1500 x1500 pixels.
5. Flip the image (I_5).
6. The original image and the flipped image are fed into the geometric transformation augmentation.

Due to the limited number of input images, geometric transformation augmentation was employed. This research explores two geometric transformation techniques, image reflection and image rotation, to generate a new set of images. In the first step, the original image is rotated by predefined angles: 0 degrees, 45 degrees, 90 degrees, 135 degrees, 180 degrees, 225 degrees, 270 degrees, and 315 degrees. Rotation is a common type of geometric transformation used for image data augmentation [16]. The rotation equation is given in (1). The and represent the new positions of each pixel after the rotation process, the x and y pairs of coordinates denote the raw image, and represents the rotation angle.

$$\begin{bmatrix} f_x \\ f_y \end{bmatrix} = \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} \quad (1)$$

This step produces a total of eight images. Subsequently, the original image is reflected across the y-axis and rotated by the same set of predefined angles as shown in Fig. 2 (I_6-I_8). This second step generates an additional eight images, resulting in a total of sixteen images, as shown in Fig. 3.

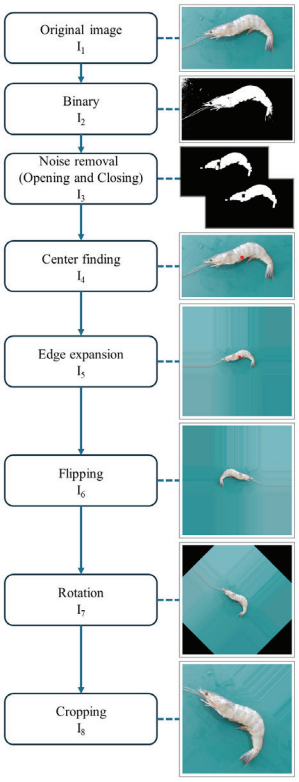
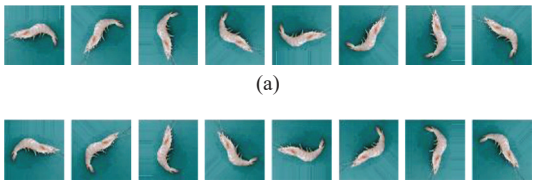


Fig. 2. Overview of the preprocessing method



(a)

(b)

Fig. 3. (a) Shows rotations of the original image at 0°, 45°, 90°, 135°, 180°, 225°, 270°, and 315°. (b) Shows the reflected image with rotations at 0°, 45°, 90°, 135°, 180°, 225°, 270°, and 315°.

2) Dataset

This dataset consists of 6,480 color images of shrimp, categorized into seven classes:

1. Complete body shrimp 1,280 images.
2. Crunch head shrimp 896 images.
3. Shrimp with head loss 864 images.
4. Head loss shrimp with remaining chin 896 images.
5. Cut tail shrimp 688 images.
6. Torn in half shrimp 720 images.
7. Total crunched shrimp 1,136 images.

All images' dimensions of 1,920 x 1,080 pixels were resized to 224 x 224 pixels to match the input requirements of the model and normalized to a range of 0-1. The dataset was randomly shuffled and divided into training (70%) and testing (30%) sets.

B. Classical CNNs and Transfer Learning Techniques

Convolutional Neural Networks (CNNs) are a category of deep neural networks specifically engineered for the processing of structured grid data, such as images[17], [18]. These networks are comprised of multiple layers, including convolutional layers, pooling layers, and fully connected layers [19].

CNNs have established themselves as a foundational framework in the domains of deep learning and computer vision due to several compelling factors: their computational efficiency, capability to effectively process image data, inherent translation invariance, hierarchical feature learning, demonstrated effectiveness, scalability, and robust community support [20], [21].

Transfer learning is a machine learning technique whereby a model developed for a specific task is repurposed as the initial framework for a model addressing a subsequent task [22], [23]. This methodology is particularly advantageous in scenarios where the subsequent task is constrained by limited data. By leveraging transfer learning, the development of high-performing models can be achieved with significantly reduced time and computational resources, thus underscoring its utility across various domains of machine learning [24].

1) Choose the family of Transfer Learning Techniques

In this research, models were selected for comparison from the Keras Applications library [11], which provides performance benchmarks for 38 models. The Top-1 Accuracy metric, which indicates a model's performance on the ImageNet validation dataset, was focused on. Four models with Top-1 Accuracy exceeding 80% and representing the best-performing model within their respective families were identified NasNetLarge, InceptionResNetV2, EfficientNetV2L, and ConvNeXtXLarge.

2) Model of Transfer Learning Techniques

The four Convolutional Neural Network (CNN) models, NasNetLarge, InceptionResNetV2, EfficientNetV2L, and ConvNeXtXLarge, were selected for comparison in this research. All models utilized transfer learning and exhibited Top-1 Accuracy exceeding 80%. Their key strengths and limitations are as follows:

NasNetLarge, developed by Google AI, employs Neural Architecture Search (NAS) to automatically discover optimal model architectures [12]. This results in high image classification accuracy but also requires significant computational resources.

InceptionResNetV2, developed by Google Research, combines Inception and Residual Network (ResNet) architectures [13]. It utilizes Inception Modules with multi-scale filters and Residual Connections to enhance learning and mitigate the vanishing gradient problem. This model is known for its accuracy and speed, but its complex architecture poses challenges for modification.

EfficientNetV2L, also developed by Google Research, is an advancement upon EfficientNet,

designed for high computational efficiency and accurate image classification [14]. It employs Progressive Learning, beginning with smaller image sizes and resolutions and gradually increasing them for faster and more efficient training. Fused-MBConv modules are used to reduce the model's parameter count and complexity, while scaling techniques allow for adaptation to available resources.

ConvNeXtXLarge, developed by Meta AI Research, is inspired by the Swin Transformer but utilizes a CNN structure for simplicity and high processing efficiency [15]. It employs a Macro Design, dividing images into patches for individual processing, and Inverted Bottlenecks, similar to ResNet but with reversed Convolutional and Depthwise Convolution layers for reduced parameter count and complexity. Large Kernel Sizes are used to enhance feature extraction, and Layer Normalization replaces Batch Normalization for improved training stability.

C. ConvNeXtXLarge Architecture Definition

The ConvNeXtXLarge architecture is shown in Fig. 4. The architecture begins by taking a 224x224 pixel input image with three color channels (RGB). This input image is then processed by a stem layer, which performs initial feature extraction and downsampling. In the stem layer, a convolutional layer with a stride of 4 is typically used, resulting in a significant reduction in spatial resolution while increasing the depth of the extracted features. The output of the stem layer is a feature map with a spatial dimension of 56x56 and 256 channels.

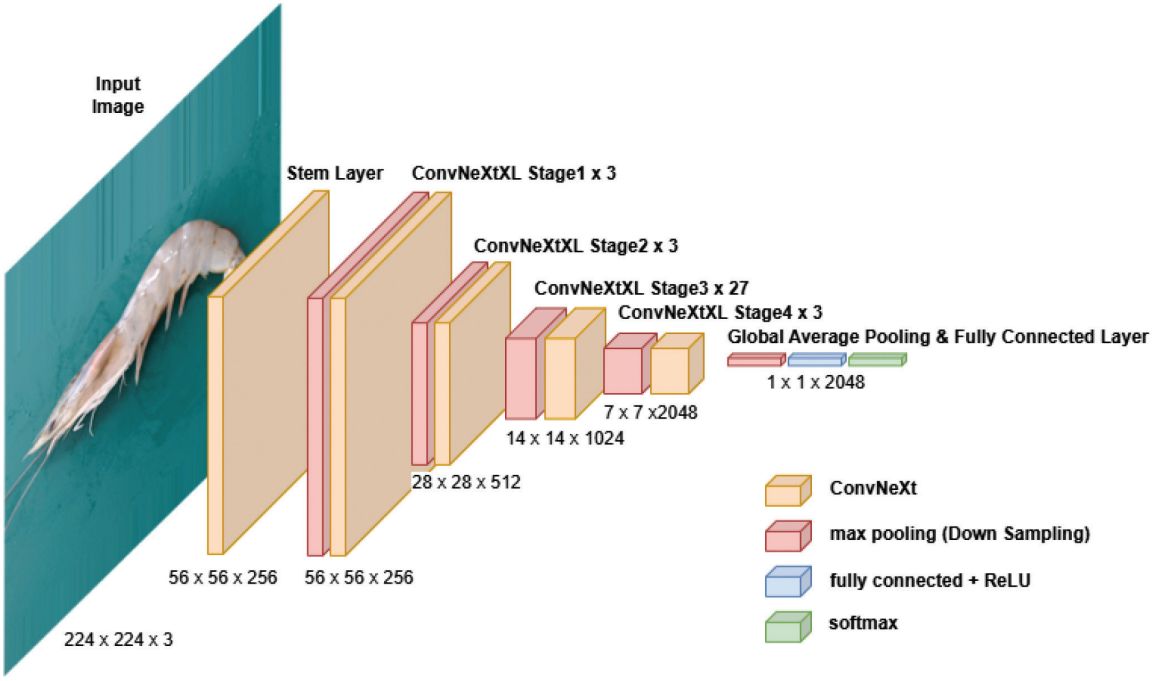


Fig. 4. ConvNeXtXLarge Architecture Diagram

Subsequent to the stem layer, the architecture features three consecutive ConvNeXtXL stages. Each stage is composed of multiple ConvNeXtXL blocks, which serve as the fundamental building blocks of the network. These blocks consist of three key components: Depthwise convolutions, layer normalizations, and inverted bottleneck layers. The Depthwise convolutions are applied to each input channel individually, offering computational efficiency. Following the Depthwise convolutions, layer normalizations are used to normalize the activations, improving the stability and convergence of the model during training. The inverted bottleneck layers then expand the number of channels before the Depthwise convolutions, allowing the model to learn more complex and expressive features.

As the feature map progresses through the ConvNeXtXL stages, its spatial dimensions are progressively downsampled while the number of channels increases. Specifically, the output of Stage 1 has a spatial dimension of 56x56 and 256 channels, Stage 2 produces a 28x28 feature map with 512 channels, and Stage 3 yields a 14x14 feature map with 1024 channels. Stage 4 further downsamples the spatial dimension to 7x7, while maintaining the 2048 channels.

The final component of the architecture is the classification head, which processes the 7x7x2048 output from Stage 4 to generate class probabilities. Global average pooling is applied to reduce each channel to a single value, resulting in a 1x1x2048

vector. This vector is then fed into fully connected layers, which learn to map the extracted features to the desired output classes. A Rectified Linear Unit (ReLU) activation function is applied to introduce non-linearity and improve the model's ability to learn complex relationships. Finally, a softmax layer is used to convert the raw scores into probabilities for each class, indicating the model's confidence in its predictions for the input image.

D. Experiments

This study investigated the application of five image classification models for shrimp image classification with two different scenarios to investigate the impact of data augmentation on a dataset. The study uses an original dataset of 405 images and compares it to an augmented dataset of 6480 images. Four pre-trained models were employed for transfer learning: InceptionResNetV2, NASNetLarge, EfficientNetV2L, and ConvNeXtXLarge were chosen from the Keras Applications library based on their high Top-1 accuracy, exceeding 80%. These models were pre-trained on the ImageNet dataset [25]. To preserve the learned features, all layers within these models were frozen. New trainable layers were added atop the pre-trained models (details in Table II) to adapt them to the shrimp classification task. The final layer was a fully connected dense layer with seven neurons and a softmax activation function, which generated a seven-class probability distribution for the shrimp images.

TABLE II
ConvNeXtXLarge LAYER DETAILS

Layer no.	Layer Type	Output Shape	Param #
0-296	ConVNeXtXLarge Layers		
297	Dense_5 (Dense)	(None,512)	1049088
298	Dense_6 (Dense)	(None,256)	131328
299	Dense_7 (Dense)	(None,128)	32896
300	Dense_8 (Dense)	(None,64)	8256
301	Dense_9 (Dense)	(None,7)	455

Total Params: 349,384,327
Trainable Params: 1,226,119
Non-trainable Params: 348,158,208

Each pre-trained model required specific input preprocessing. InceptionResNetV2, NASNetLarge, and EfficientNetV2L necessitated scaling input pixels to a range between -1 and 1. ConvNeXtXLarge, however, accepted the original input pixel range of 0 to 255. The classical CNN model, on the other hand, required scaling between 0 and 1. A batch size of 32 was used to train all models. The default training epoch was set to 30, with early stopping implemented using patience of 3. The learning rate was set to 0.001 and the optimizer was set to Adam. This signifies that training would halt and restore the best parameter weights as the final model if the validation loss failed to improve for three consecutive epochs. Each model was trained ten times, with the results reported as average precision, recall, F1-score, and accuracy. Moreover, average training time, average inference time, and memory usage were reported to compare the computational complexity of each model.

The experiments were conducted on a Windows 11 Home system equipped with an AMD Ryzen 7 5800 8-Core Processor at 3.40 GHz, 32.0 GB of installed RAM, and an Nvidia GeForce RTX 3070 graphics card. Python was the programming language utilized, and TensorFlow version 2.10.1 facilitated the training on the Nvidia GPU.

III. RESULT AND DISCUSSION

The table presents a comparison of the transfer learning models and the classical CNN model on the shrimp image classification task with and without augmentation. The results reported are the average training time, inference time, memory usage, precision, recall, F1-score, and accuracy across ten training repetitions for the seven classes. The overall results show that the accuracy of the augmented dataset is higher than the non-augmented dataset. The augmentation technique plays a crucial role in increasing model performance.

For the performance comparison, the transfer learning model, ConvNeXtXLarge, achieved superior performance compared to the other transfer learning models and the classical CNN model, particularly in F1-score and accuracy, reaching a value of 95%. This observation could be attributed to the size and complexity of ConvNeXtXLarge. As the largest model among those available in Keras applications, it has 350.1 million trainable parameters and a size of 1310 MB. This increased capacity might have contributed to its superior performance not only on the ImageNet validation dataset during pre-training but also on our specific shrimp abnormality classification validation dataset.

ConvNeXtXLarge demonstrated the highest accuracy among the evaluated models and exhibited a relatively efficient training time, ranking second only to NASNetLarge. Notably, ConvNeXtXLarge achieved a 15% accuracy improvement over NASNetLarge. However, it also presented the highest inference time and memory consumption among all the transfer learning models considered. This is attributed to its complex architecture, comprising 297 layers.

The training curves for loss and accuracy, as illustrated in Fig. 5 and Fig. 6 respectively, provide insights into the training dynamics of the ConvNeXtXLarge model. Early stopping was triggered at epoch 7 due to the inability of the model

to improve validation loss for three consecutive epochs. Consequently, the training process reverted to the weights associated with epoch 4, which achieved a validation accuracy of approximately 0.96.

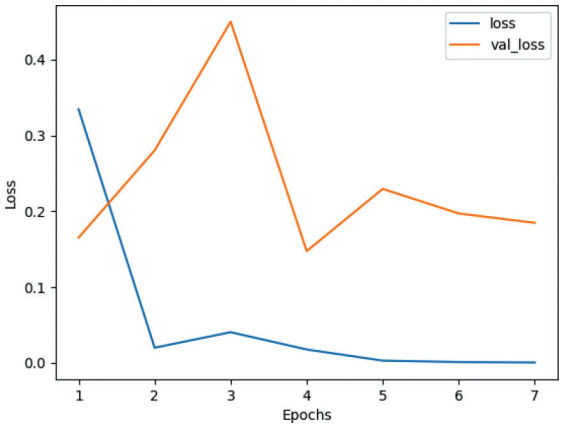


Fig. 5. Loss and validation loss when training ConvNeXtXLarge Model

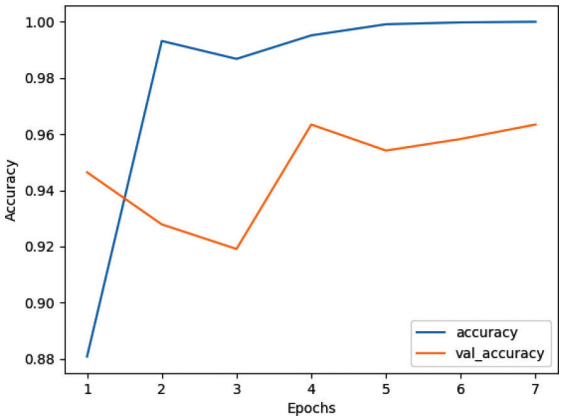


Fig. 6. Accuracy and validation accuracy when training ConvNeXtXLarge

Fig. 7 depicts the confusion matrix as a heatmap, visualizing the relationship between true labels and predicted labels. The dominant presence of values along the diagonal signifies a high degree of accurate predictions. However, a minor number of misclassifications are observed, with the BustedHead class exhibiting the lowest F1-score (0.93) as detailed in Table IV. This class appears to be visually similar to the Chin and Complete classes, potentially leading to confusion during the model's learning process. Similar misclassifications are observed for the TailOff and Half classes, both achieving an F1-score of 0.94 (second lowest).

The superior performance observed in certain models, notably ConvNeXtXLarge, can be attributed to their advanced architectures. ConvNeXtXLarge benefits from the ConvNeXt architecture, incorporating Depthwise convolution, inverted bottlenecks, and attention mechanisms, which facilitate efficient feature extraction. As depicted in Table III, ConvNeXtXLarge yields the highest accuracy, but also exhibits the highest inference time and memory consumption.

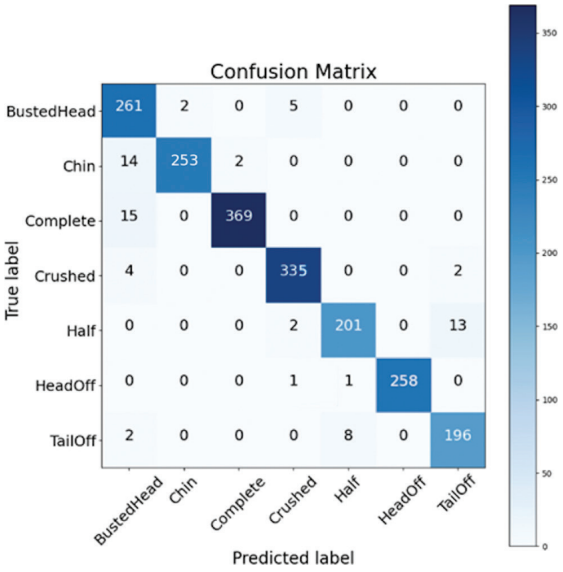


Fig. 7. Confusion matrix of the ConvNeXtXLarge Model

EfficientNetV2L leverages the EfficientNet architecture, prioritizing a balance of model size, processing speed, and accuracy through techniques such as compound scaling and progressive learning. Table III reveals that this model achieves the second-highest accuracy while maintaining lower inference time and memory usage compared to the top-performing model.

In contrast, InceptionResNetV2 and NASNetLarge, while employing complex architectures with inception modules and residual connections, exhibit comparatively lower performance. In Table III shows these two models demonstrate lower inference time and memory usage; however, exhibit comparatively lower performance.

TABLE III
THE COMPARISON AVERAGE OF TRAINING TIME, INFERENCE TIME, MEMORY USAGE, PRECISION, RECALL, F1- SCORE AND ACCURACY OF AUGMENTATION (AUG) AND NO AUGMENTATION (No AUG)

Model	Avg. Training Time (mins)	Avg. Inference Time (ms per Sample)	Memory Usage (GiB)	Avg. Precision	Avg. Recall	Avg. F1-score	Avg. Accuracy
Classical CNN (AUG)	21.42	26.79	3.44	0.79	0.77	0.77	0.77
Classical CNN (No AUG)	1.27	28.73		0.75	0.72	0.72	0.72
InceptionResNetV2 (AUG)	14.6	22.15	3.75	0.84	0.84	0.84	0.84
InceptionResNetV2 (No AUG)	2.09	31.58		0.818	0.81	0.807	0.81
NASNetLarge (AUG)	11.47	22.48	3.99	0.81	0.80	0.80	0.80
NASNetLarge (No AUG)	1.83	32.77		0.79	0.78	0.77	0.78
EfficientNetV2L (AUG)	19.10	23.49	4.28	0.88	0.88	0.88	0.88
EfficientNetV2L (No AUG)	2.53	35.48		0.83	0.82	0.82	0.82
ConvNeXtXLarge (AUG)	12.83	24.16	4.91	0.96	0.95	0.95	0.95
ConvNeXtXLarge (NoAUG)	2.35	37.29		0.94	0.93	0.93	0.93

TABLE IV
PRECISION, RECALL, F1-SCORE OF SEVEN CLASSES

Class Label	Precision	Recall	F1-score	Supports
BustedHead	0.88	0.97	0.93	268
Chin	0.99	0.94	0.97	269
Complete	0.99	0.96	0.98	384
Crushed	0.98	0.98	0.98	341
Half	0.96	0.93	0.94	216
HeadOff	1.00	0.99	1.00	260
TailOff	0.93	0.95	0.94	206
Accuracy	-	-	0.96	1944
Macro avg	0.96	0.96	0.96	1944
Weighted avg	0.97	0.96	0.96	1944

The goal of this model selection process is to identify the optimal model for deployment on an embedded system connected to a camera for shrimp sorting on a conveyor belt. Resource constraints of the system play a crucial role in this selection.

For systems with ample resources, ConvNeXtXLarge is recommended due to its superior performance. However, when resources such as processing power and memory are limited, a smaller model with acceptable accuracy and inference time, like EfficientNetV2L, becomes a more suitable choice.

In extremely resource-constrained scenarios, InceptionResNetV2 or NASNetLarge may be considered. While these models offer reduced inference time and memory usage, this efficiency comes at the cost of lower accuracy. Therefore, the final model selection requires a careful trade-off between performance and resource utilization based on the specific limitations of the embedded system.

Future improvements could be achieved by fine-tuning the transfer learning models. This involves freezing certain layers of the pre-trained models and retraining them with an adjusted learning rate. Additionally, the shrimp dataset could be further refined to ensure class balance, mitigating potential bias and enhancing model performance.

IV. CONCLUSION

Shrimp transportation often results in damage to the product. Consequently, it is crucial to implement a shrimp sorting system prior to processing to remove damaged shrimp from the production line. Damaged shrimp are categorized into six grades and sold at varying prices, thereby enhancing the value of these compromised products.

This research employed two scenarios: a shrimp dataset without augmentation and an augmented shrimp dataset. Geometric transformation augmentation was used to increase the dataset size and serve as input to four top-1 accuracy models with over 80% accuracy from Keras Applications: NasNetLarge, InceptionResNetV2, EfficientNetV2L, and ConvNeXtXLarge. These models were evaluated against a baseline CNN model on a dataset with seven

shrimp classes: complete body shrimp, crushed head shrimp, shrimp with head loss, head loss shrimp with remaining chin, cut tail shrimp, torn-in-half shrimp (upper half and lower half), and totally crushed shrimp.

Experimental results revealed two primary findings. Firstly, the augmented shrimp dataset achieved superior performance compared to the non-augmented dataset. Secondly, in the performance comparison, the transfer learning model ConvNeXtXLarge emerged as the superior model achieving the highest accuracy (95%), precision (0.96%), recall (0.95%), and F1-score (0.95%). Analysis of misclassifications indicated that certain classes share similar characteristics, potentially causing model confusion. However, further accuracy improvement is possible through fine-tuning. Future research directions include developing an automated shrimp sorting machine on a conveyor belt utilizing this model.

Acknowledgment

During the preparation of this work, the authors used ChatGPT, Google Gemini, and Microsoft Copilot to check and correct grammatical errors and paraphrase passages during the manuscript writing process. After using these tools/services, the authors reviewed and edited the content as needed and took full responsibility for the content of the publication.

To thank Mr. Patcharapol Boonsin, owner of a shrimp pond in Chachoengsao Province, who provided information about shrimp farming and about shipping shrimp to the processing factory.

References

[1] Centre for Agricultural Information, Office of Agricultural Economics. (2024, Aug. 22). *Thailand Foreign Agricultural Trade Statistics 2023*. [Online]. Available: https://www.oae.go.th/assets/portals/1/files/jounal/2567/tradestat2566.pdf

[2] N. Masunee, S. Chaiprapat, and K. Waiyagan, "Development of an Image Processing System in Splendid Squid Quality Classification," in *Proc. ICDIP*, 2013, pp. 252-259.

[3] N. Thanasarn, S. Chaiprapat, K. Waiyakan et al., "Automated Discrimination of Deveined Shrimps Based on Grayscale Image Parameters," *J. Food Process Eng.*, vol. 42, no. 4, p. e13041, Mar. 2019.

[4] P. H. Andersen, *Sustainable Operations Management (SOM) Strategy and Management: An Introduction to Part I.* Melbourne, AU: Palgrave Macmillan, 2019, pp. 15-25.

[5] R. Kler, G. Elkady, K. Rane et al., "Machine Learning and Artificial Intelligence in the Food Industry: A Sustainable Approach," *Journal of Food Quality*, vol. 2022, no. 1, pp. 1-9, May 2022.

[6] T. Valeeprakhon, K. Orkphol, and P. Chaihuadjaroen, "Deep Convolutional Neural Networks Based on VGG-16 Transfer Learning for Abnormalities Peeled Shrimp Classification," *International Scientific Journal of Engineering and Technology (ISJET)*, vol. 6, no. 2, pp. 13-23, Dec. 2022.

[7] H. Yu, X. Liu, H. Qin et al., "Automatic Detection of Peeled Shrimp Based on Image Enhancement and Convolutional Neural Networks," in *Proc. The 8th International Conference on Computing and Artificial Intelligence*, 2022, pp. 439-450.

[8] Y. Zhang, C. Wei, Y. Zhong et al., "Deep Learning Detection of Shrimp Freshness Via Smartphone Pictures," *Journal of Food Measurement and Characterization* , vol. 16, no. 5, pp. 3868-3876, Jun. 2022.

[9] K. Wang, C. Zhang, R. Wang et al., "Quality Non-Destructive Diagnosis of Red Shrimp Based on Image Processing," *Journal of Food Engineering*, vol. 357, p. 111648, Jul. 2023.

[10] K. Prema and J. Visumathi, "An Improved Non-Destructive Shrimp Freshness Detection Method Based on Hybrid CNN and SVM with GAN Augmentation," in *Proc. 2022 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)*, 2022, pp. 1-7.

[11] Keras Team. (2024, August 22). *Keras Applications*. [Online]. Available: https://keras.io/api/applications/

[12] B. Zoph, V. Vasudevan, J. Shlens et al., "Learning Transferable Architectures for Scalable Image Recognition," in *Proc. The IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8697-8710.

[13] C. Szegedy, S. Ioffe, V. Vanhoucke et al., "Inception-v4, Inception-Resnet and the Impact of Residual Connections on Learning," in *Proc. The AAAI Conference on Artificial Intelligence*, 2017. pp. 4278-4284.

[14] M. Tan and Q. Le, "Efficientnetv2: Smaller Models and Faster Training," in *Proc. International Conference on Machine Learning, PMLR, 2021*, pp. 10096-10106.

[15] Z. Liu, H. Mao, C. Y. Wu et al., "A Convnet for the 2020s," in *Proc. The IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11976-11986.

[16] N. E. Khalifa, M. Loey, and S. Mirjalili, "A Comprehensive Survey of Recent Trends in Deep Learning for Digital Images Augmentation," *Artif. Intell. Rev.*, vol. 55, no. 3, pp. 2351-2377, Sep. 2022.

[17] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, no. 7553, pp. 436-444, May. 2015.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet Classification with Deep Convolutional Neural Networks," *Adv. Neural Inf. Process. Syst*., vol. 25, 2012, pp. 84-90.

[19] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, Massachusetts: MIT Press, 2016, pp. 1-800.

[20] W. Rawat and Z. Wang, "Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review," *Neural Comput.*, vol. 29, no. 9, pp. 2352-2449, Jun. 2017.

[21] J. Gu, Z Wang, J. Kuen et al., "Recent Advances in Convolutional Neural Networks," *Pattern Recognition*, vol. 77, pp. 354-377, May. 2018, doi: https://doi.org/10.1016/j.patcog.2017.10.013

[22] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345-1359, Oct. 2009.

[23] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A Survey of Transfer Learning," *Journal of Big Data*, vol. 3, no. 1, p. 9, Dec. 2016, doi: 10.1186/s40537-016-0043-6

[24] C. Tan, F. Sun, T. Kong et al., "A Survey on Deep Transfer Learning," in *Proc. 27th International Conference on Artificial Neural Networks, Rhodes, Greece*, 2018, pp. 270-279.

[25] J. Deng, W. Dong, R. Socher et al., "Imagenet: A Large-Scale Hierarchical Image Database," in *Proc. 2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248-255.

Korawit Orkphol is a lecturer at the Department of Computer Engineering, Faculty of Engineering at Sriracha, Kasetsart University, Sriracha campus, Thailand. He graduated Ph.D. in Computer Science and Technology from Harbin Engineering University, P.R. Harbin, China. His research field including of Artificial Intelligence, Machine Learning, and Natural Language Processing.

Kathawach Satianpakiranakorn is a lecturer at the Department of Computer Engineering, Faculty of Engineering at Sriracha, Kasetsart University, Sriracha campus, Thailand. He graduated M.Eng. in Computer Applied Technology from Harbin Engineering University, P.R. Harbin, China. His research field including of Artificial Intelligence, Computer Visions, and Digital Image Processing.

Jidapa Chaihuadjaroen is an Assistant Professor at the Department of Computer Engineering, Faculty of Engineering at Sriracha, Kasetsart University, Sriracha campus, Thailand. She graduated M.S in Computer Science from King Mongkut's Institute of Technology Ladkrabang, Thailand. Her research field including of Computer Vision and Image Processing.

Tamnuwat Valeeprakhon is an Assistant Professor at the Department of Computer Engineering, Faculty of Engineering at Sriracha, Kasetsart University, Sriracha campus, Thailand. He graduated M.Eng in Computer Engineering from Khon Kaen University, Thailand. He was granted a scholarship from the National Science and Technology Development Agency (NSTDA) for studying at the Kochi University of Technology, Japan. His research interests including of Computer Vision, Image Processing, and Artificial Intelligence.

Enhancing Warehouse Management with AI and Computer Vision: A Case Study in a Logistics Service Company

Nalinya Utamapongchai[1*], Sirawich Ngernsalung[2], Raveekiat Singhaphandu[3], and Warut Pannakkon[4*]

[1,2,3,4]School of Manufacturing Systems and Mechanical Engineering, Sirindhorn International Institute of Technology, Thammasat University, Pathum Thani, Thailand

[3]School of Knowledge Science, Japan Advanced Institute of Science and Technology, Ishikawa, Japan

E-mail: nalinya.utm@gmail.com*, sirawich.nge@dome.tu.ac.th, raveekiat@gmail.com, warut@siit.tu.ac.th

Received: June 21, 2024 / Revised: September 6, 2024 / Accepted: September 17, 2024

***Abstract*— In the evolving landscape of warehouse management in Industry 4.0, this paper explores the convergence of Artificial Intelligence (AI) and Computer Vision (CV) for inventory tracking and stock registration. Conducted in collaboration between SIIT and KNS, a logistics service company specializing in warehousing, the study introduces a framework that optimizes image capture conditions through real-time analysis of gyroscope values, distinguishing mobile phone movement from stationary states. Additionally, an object detection model using the YOLOv8 algorithm achieves 83% accuracy in label detection and 75% in box detection within a curated dataset. The research highlights the successful development of the phone motion detection model and Optical Character Recognition (OCR) integration. This framework promises to advance warehouse management systems by addressing current limitations with a comprehensive, efficient, and user-friendly solution.**

***Index Terms*— Digital Transformation, Artificial Intelligence, Computer Vision, Inventory Tracking**

I. Introduction

In the present time, it is a transformative era, where the world is rapidly progressing towards a new age characterized by innovative and technological revolution [1]. This age is also defined by the convergence of numerous advancements, essential among them being the development of Industry 4.0 and the extraordinary growth of Artificial Intelligence (AI) [1], [2].

The rapid growth of AI does not only enable more applications in various fields, but it also enhances the capabilities of most of its features, including computer vision technology [1], [2]. Computer vision is referred to as a field of AI that enables computers and systems to derive meaningful information from digital images, videos, and other visual inputs [3], [4]. Before the AI era, computer vision primarily relied on handcrafted algorithms, which were time-consuming and low in capacity and accuracy. However, as deep learning becomes predominant, it has now become the core part of computer vision [3], [4]. This significant change allows the algorithms to handle more complex and diverse scenes with precise results [1], [3].

As a result, modern computer vision technology is utilized in numerous fields. Plus, with the development of big data and the Internet of Things (IoT), a powerful, well-balanced integrated management system is established [2], [3], [5]. Some of the fields gaining the most benefit from this system are warehousing and inventory management [2], [6].

However, most warehouse management technologies offer automation, visualisation, and inventory tracking, suitable for large-scale systems. Challenges include inflexibility to system changes, high initial costs, and complex, immobile control programs [2], [6], [7]. Barcode scanners, the primary outgoing product check tool, have limitatio ns, scanning one product at a time and causing operator fatigue [6], [8]. A significant gap exists in tracking stocks moving through the system without proper registration, hindering analysis of stock flow duration in the warehouse.

Thus, with the engagement of AI along with Optical Character Recognition (OCR) technology in computer vision, this study aims to achieve the following objectives: (1) Develop a phone motion detection model that optimizes image capture conditions through real-time analysis of gyroscope values; (2) Apply the knowledge of computer vision and OCR to develop box and label detection models for a stock registration/record system; (3) Evaluate the performance of the developed models. The rest of the paper is organized as follows: Part II - Related Work, including literature review and relevant theory, Part III - Methodology, including project framework and performance evaluation, Part IV - Case Study, including the procedures, Part V - Result and Discussion, and Part VI - Conclusion and Future Work.

II. Related Work

In the domain of Artificial Intelligence (AI), Computer Vision (CV) plays a pivotal role by enabling machines to comprehend visual information akin to human vision [4]. This extends from image and video analysis to intricate tasks such as object recognition, tracking, and scene understanding [7], [9], [10]. The integration of CV algorithms and Optical Character Recognition (OCR) techniques augments its capabilities, allowing localization of regions of interest and text recognition [11], [12].

In the context of Industry 4.0, a transformative shift towards a connected, automated, and data-driven approach is witnessed, with key components including the Internet of Things (IoT), Big Data and Analytics, AI, and Machine Learning (ML) [4], [5], [10]. The emergence of smart Warehouse Management Systems (WMS) exemplifies the interconnected use of AI, CV, and IoT to optimize warehouse processes, aiming for improved accuracy and efficiency [2], [6]. This holistic approach spans stock planning, product placement, order picking, transport, and tracking within the warehouse workflow, presenting opportunities for optimization in each process.

In the retail sector, CV's rapid growth has led to a focus on product image recognition systems [9]. This framework encompasses image capture, pre-processing, feature extraction, feature classification, and recognition output, offering a systematic approach to image-based recognition.

The emphasis on high-quality input images is critical for the success of CV applications, underlining the importance of meticulous image management from the initial capture phase [13]. Simultaneously, the investigation into the utility of built-in tri-axial accelerometer and gyroscope sensors in smartphones has been conducted across various scenarios involving human motion [14], [15]. This exploration extends to diverse activities, including the recognition of hand gestures, and extends further by incorporating additional wrist-worn sensor attachments [16]. The seamless integration of these technologies contributes to a comprehensive understanding of motion, enabling enhanced applications in image recognition and other CV-related tasks.

Additionally, the YOLO (You Only Look Once) algorithm is scrutinized for real-time object detection, assessing its characteristics and performance. Various versions of YOLO have been developed, incorporating innovative ideas and techniques, offering researchers new directions for addressing challenges in object detection [17]-[19].

In conclusion, the comprehensive review of related work demonstrates the dynamic evolution of intelligent systems, highlighting the integration of AI, CV, and IoT in Industry 4.0. The utilities of gyroscopes and accelerometers in smartphones have the potential to be applied as key facilitators in the image acquisition stage, effectively reducing barriers and skill requirements for operators. These findings collectively contribute to the ongoing exploration of advanced technologies for enhanced image recognition and object detection in the research presented in this paper.

III. Methodology

This part consists of two main sections: A discussion of the generalized concept of the proposed framework and a performance evaluation. Fig. 1 illustrates the framework of the proposed system for double-checking outbound processes.

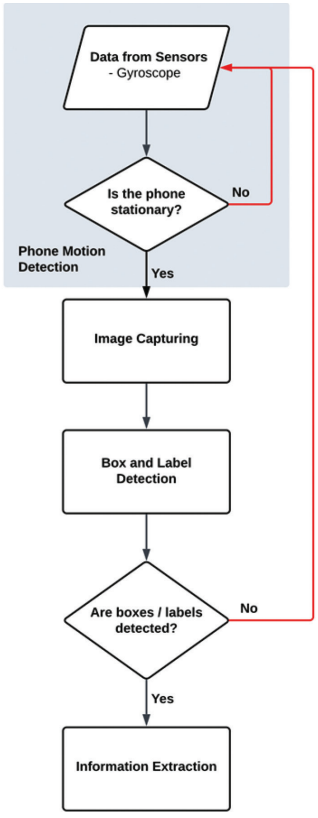


Fig. 1. Framework of the proposed system

A. Proposed Framework

The proposed methodology starts with a foundational hypothesis suggesting that images captured under stationary conditions with a mobile phone inherently have good quality, leading to successful OCR outcomes. To validate this hypothesis, the model was developed using Android Studio with the capability to identify whether the phone is stationary or in motion, which is crucial in determining image quality for OCR purposes. After that, the captured images undergo box and label detection models scoping down the area of interest before passing on to the information extraction step by the OCR process as illustrated in Fig. 2.

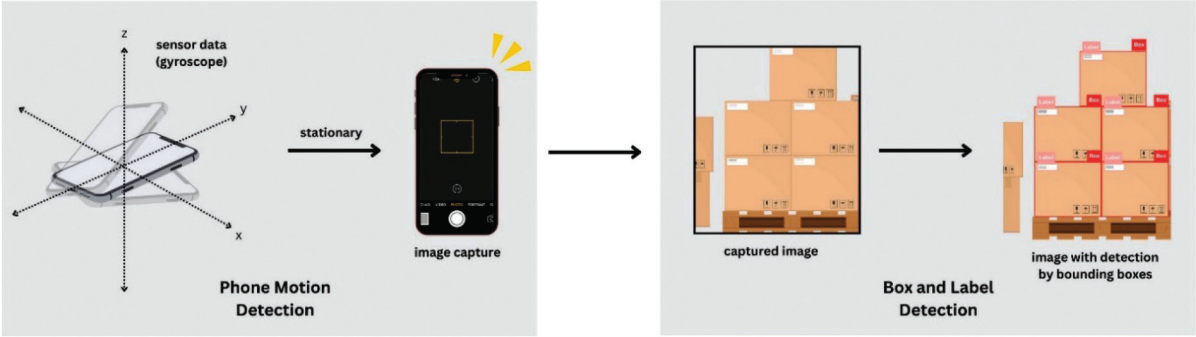


Fig. 2. System function

1) Phone Motion Detection

In practical terms, the application employs a comparative analysis of gyroscope values against predetermined threshold values. If the gyroscope values fall within the defined limits across all axes, the phone is considered stationary and ready for image capture. Conversely, if one or more gyroscope values surpass the established thresholds, the application identifies the phone as in motion.

In contrast to the gyroscope's comparative analysis, the application will not utilize accelerometer sensor values for motion detection. This decision is based on the consistent influence of gravitational force (g) on the phone, yielding sensor values around g (9.81 m/s^2) or negative g (-9.81 m/s^2), particularly influenced by the device's orientation.

During the phase of phone movement, the application should consistently exhibit a "NOT READY" status. It is anticipated that the readiness status will shift to "READY" when the phone comes to a stop and will persist in this status during stationary phases.

2) Box and Label Detection

The data collection phase began with the use of a mobile phone camera to capture images of boxes and labels from various angles and distances. This approach aimed to provide a diverse representation of real-world scenarios, allowing the model to extract relevant features and patterns essential for robust performance across different situations.

After the completion of the data collection phase, Roboflow played a pivotal role in streamlining the workflow. Its annotation tools facilitated the precise annotation of collected images with bounding boxes for both boxes and labels, ensuring accurate object localization. Moreover, Roboflow's augmentation techniques, including shear, blur, and noise, expanded the dataset and enhanced its diversity.

Leveraging Roboflow for image pre-processing, YOLOv8 effectively processed the augmented data to accurately detect and classify objects within the images. Capitalizing on YOLOv8's deep neural network architecture, the model demonstrated the capability to simultaneously detect multiple objects, including boxes and labels.

B. Performance Evaluation

The evaluations are required for two main processes of the framework, phone motion detection and box and label detection.

The evaluation of phone motion detection focuses on the application's ability to dynamically adapt its status indicators in response to real-time phone movements, ensuring an accurate depiction of readiness during stationary periods and non-readiness during mobile phases. This is achieved through the intricate processing of sensor data and threshold comparisons, which distinguish between stationary and moving states. With the important role of threshold value resulting in the sensitivity of motion detection, the value adjustment allows the application to suitably perform in the simulated scanning task, replicating the real-world use.

The evaluation of the box and label detection models involves a comprehensive examination of their performance in accurately identifying and categorizing boxes and labels within a predefined dataset.

The performance metrics are considered, with a specific emphasis on scrutinizing a confusion matrix. This matrix serves as a pivotal analytical tool, offering a nuanced understanding of the model's classification errors and providing valuable insights into specific challenges and areas for improvement, particularly in the domain of multi-label classification. The essential

elements within the confusion matrix, encompassing True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN), provide insights into the model's strengths and weaknesses, especially in box and label classification scenarios, highlighting specific areas and allowing for targeted enhancements to the model's predictive capabilities.

IV. Case Study

In our case study, the proposed framework is applied to a real-world application to develop a system implemented in two main sections: phone motion detection and box and label detection.

A. Phone Motion Detection

In adherence to the methodology outlined earlier, our application conducts a real-time comparative analysis of gyroscope sensor values against a predefined threshold to discern the stationary or moving status of the mobile phone. During the tuning process, through trial-and-error experiments, this threshold is set at 0.2, taking into account specific characteristics of the mobile device and factors related to human interaction. This calibrated value strikes a balance, ensuring the phone remains stable enough for high-quality image capture while remaining manageable for users to stay within the specified limits.

As the application currently exists in a prototype stage, the image-capturing process is not activated immediately. Once the system confirms the phone's stationary state, the user interface will prominently display the status "READY" accompanied by a reassuring green box. This visual cue signals that the phone is primed for image capture. Conversely, if the gyroscope values exceed the threshold, the application communicates the status "NOT READY" along with a conspicuous red square, signaling a temporary pause in the image-capturing process. This streamlined feedback mechanism aligns with the application's intent to enhance user experience and maintain control over imaging conditions.

B. Box and Label Detection

1) Data Collection

In the objective of developing a robust object detection and classification model applicable in practical scenarios, a comprehensive dataset was collected from KNS Logistics Service Company Limited, a company offering extensive services in warehouse management, storage, and distribution of products. The dataset comprises images and videos capturing diverse dimensions and characteristics of boxes and labels within the warehouse and dock stations. A mobile phone camera was utilized during the data collection process to replicate the anticipated implementation method.

Each product was captured from multiple angles, including various distances to encompass a range of perspectives. The decision to incorporate both long-distance and close-up images was to ensure the clarity of information captured for both boxes and labels. The intentional inclusion of multiple perspectives enhances the model's adaptability to real-world scenarios, where products may be encountered at different distances. Sample images captured are illustrated in Fig. 3 to Fig. 5.

Fig. 3. Sample of the captured image of boxes

Fig. 4. Sample of the captured image of labels

Fig. 5. Sample of captured image

To enrich the dataset further, videos were recorded, showcasing individual labels or groups of labels continuously. This deliberate approach aimed to capture temporal variations and nuances in label presentation, contributing to a more comprehensive understanding of the visual context. In total, 140 images and video snapshots were initially prepared, and through special augmentation techniques provided by Roboflow, the dataset was expanded to a final count of 340 samples.

2) Model Development

Since the automatic image-capturing algorithm has been waiting for forthcoming development, the manually captured images have been used as the imitated datasets to ensure the performance of the detection model. The collected images were processed and prepared through Roboflow before being passed onto the YOLOv8 model for training by steps as follows:

• *Image Annotation*: Roboflow's annotation tools were utilized with 140 uploaded images containing objects to be detected and nulled images with bounding boxes, ensuring accurate annotations for boxes and labels.

• *Train/Test Split*: To contribute to the accuracy and reliability of the model, the dataset was randomly split into training, validating, and testing sets with the set of ratios (i.e., 70%:20%:10%).

• *Pre-Processing*: All images in the dataset underwent image transformations with the aim of reducing training time and enhancing performance. This was achieved through the application of the auto-orient and resize functions.

• *Augmentation*: New variations and the increase of training images were performed by augmentation application. The additional images for the training set of the model are generated from the existing in augmented versions with shear, blur, and noise effects, resulting in 340 total images.

• *Model Training*: The annotated and augmented dataset underwent conversion to a format compatible with YOLOv8. Subsequently, it was subjected to the training process, aiming to minimize the disparity between predicted and actual outputs.

V. Result

The mobile application prototype was rigorously evaluated for its performance, particularly focusing on its ability to correctly identify the phone's motion status during the experiments. The application demonstrated excellent performance, with no false results recorded throughout the testing phase. Specifically, as shown in Fig. 6, when the mobile phone was in motion, the application reliably indicated a 'NOT READY' status. Conversely, when the phone was stationary, it swiftly transitioned to the 'READY' status, maintaining this state effectively without requiring excessive stabilization efforts.

Fig. 6. The mobile application's User Interface (UI) displaying the current status

The box and label detection models were tested using a series of input images, with the results visualized in figures (Fig. 7 to Fig. 9). The detection models achieved a 75% accuracy rate in identifying boxes and an 83% success rate in detecting labels, as demonstrated by the confusion matrix in Fig. 10. However, the confusion matrix also highlighted areas of concern: the box detection model showed a 43% false positive rate, and the label detection model showed a 57% false positive rate. A possible cause of these false positives is that the training image sets contain boxes and labels at various distances, which may have caused the model to lack a three-dimensional understanding, leading it to mistakenly identify non-relevant objects as the object of interest. Additionally, some boxes have multiple colors and patterns, which can further confuse the model when distinguishing the object of interest.

Fig. 7. Result of multi-label detection for boxes

Fig. 8. Result of multi-label detection for labels

Fig. 9. Result of multi-label detection for boxes and labels

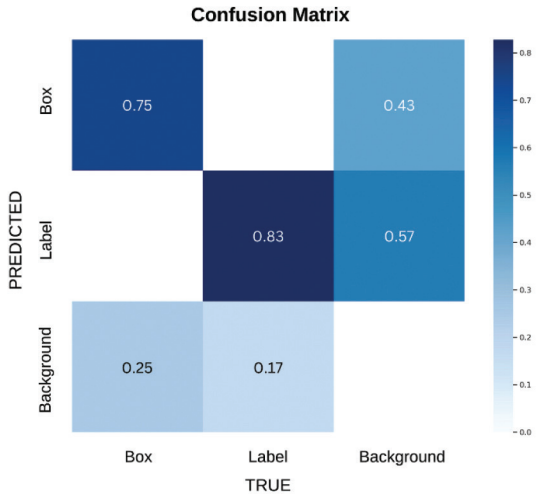


Fig. 10. Confusion matrix of multi-label detection

An example of a false positive case is shown in Fig. 11 and Fig. 12. Fig. 11 illustrates the human annotation (ground truth), while Fig. 12 displays the model's prediction. It appears that the model mistook the white rectangle on the top left corner as the label of interest, highlighting a specific scenario where the model's limitations are evident.

Fig. 11. Example of human annotation (ground truth)

Fig. 12. Example of false positive prediction

VI. Discussion

The sensitivity of the motion detection feature in the mobile application was found to be finely tuned, balancing the need to distinguish between minor, natural hand movements and significant motion. The selected threshold values were set to accommodate small shakes or sways that naturally occur due to hand fatigue, ensuring that these do not falsely trigger motion detection. This careful calibration minimizes errors and enhances the accuracy of the application's ability to detect when the phone is truly stationary versus in motion. Despite the success of the current motion detection capabilities, the application currently only displays the status ('READY' or 'NOT READY') without further action. Future developments should focus on integrating this motion detection with an automatic image-capturing process that activates when the device is in the 'READY' state, enabling continuous image capture and real-time data processing.

The results suggest that the box and label detection models may be experiencing overfitting, as indicated by the high false positive rates. Overfitting occurs when the model learns specific features from the training data that do not generalize well to new, unseen data, potentially mistaking noise for actual patterns. On the other hand, the false negative rates—25% for boxes and 17% for labels—point to underfitting, where the model fails to capture important details, resulting in missed detections. A critical observation is that the current dataset is more suited to label detection due to the close-up nature of the images, which limits the visibility of entire boxes. To improve the robustness and generalizability of the models, expanding the dataset to include a more diverse range of images is necessary. This expansion will likely reduce both overfitting and underfitting, enhancing the overall performance of the detection models.

VII. Conclusion and Future Work

The culmination of this study has resulted in the successful development of two integral components: The phone motion detection and the box and label detection models. The phone motion detection model adeptly utilizes real-time gyroscope values to discern the dynamic state of the phone, effectively distinguishing between periods of movement and stationary conditions. The performance evaluation, conducted through a simulated scanning process, underscores the application's excellence by consistently and accurately indicating the mobile phone's status.

In parallel, the box and label detection models are powered by a meticulously curated dataset comprising 140 images and videos of actual products collected during a company visit. Augmented and trained using the YOLOv8 model through Rob flow, the model demonstrates commendable performance metrics.

Notably, label detection achieves an impressive 83% accuracy, considering the limited training samples, while box detection lags with 75% accuracy. This disparity can be attributed to the nature of the collected samples, which favoured label detection due to their proximity to the boxes, making the latter challenging to discern clearly.

While both components have met expectations and demonstrated exceptional individual performance, acknowledging the need for revision and enhancement is crucial. Further refinement is essential before amalgamating the mobile application prototype and the object detection and classification model into a comprehensive and polished version of the application. This research lays a solid foundation for future iterations, emphasizing the continuous pursuit of excellence in mobile application development and object detection technology.

In considering future work, several key recommendations emerge to enhance the overall functionality and performance of both the mobile application and the object detection and classification model. For the phone motion detection model, an imperative focus lies in developing a streamlined mechanism to access the mobile phone's camera. This feature is crucial to enable the application to initiate image capture during periods when the status is deemed ready. Additionally, there is a clear opportunity to refine and elevate the User Interface (UI) to ensure a simpler, more professional appearance, incorporating all essential information seamlessly for user clarity and engagement.

Furthermore, an experiment on the gyroscope's threshold values is warranted to ascertain the optimal settings, striking a balance between minimizing false results and ensuring a user-friendly configuration.

For the box and label detection models, it is recommended to conduct a prospective visit to the company for the collection of more samples, specifically long-distance shots, to enhance the performance of box detection. Continual training iterations for the model should be pursued to ensure sustained improvement in accuracy and overall performance. Additionally, integrating OCR into the model is a promising avenue to extract serial number information from detected labels, contributing valuable data for subsequent processing. The consideration of using PPOCR should be processed in this further step as its ease of use, lightweight design, and minimal computational requirements. It supports multiple languages, handles text in various orientations and backgrounds, and is backed by good documentation and community support, simplifying integration and troubleshooting. Incorporating phone motion detection with the box and label detection models, future efforts should concentrate on establishing a consistent mechanism to pass captured images to the model for further processing. Implementing a robust storage solution,

such as a database, will be essential for efficiently storing the extracted information. Furthermore, efforts can be directed toward linking the extracted information with the sales order database to facilitate a comprehensive double-check process, thereby enhancing the overall accuracy and reliability of the system.

Acknowledgments

The authors wish to express their sincere appreciation to KNS Logistics Service Company Limited for graciously providing invaluable information and industry insights that greatly supported this study.

References

[1] R. S. Peres, X. Jia, J. Lee et al., "Industrial Artificial Intelligence in Industry 4.0-Systematic Review, Challenges and Outlook," *IEEE Access*, vol. 8, pp. 220121-220139, Dec. 2020.

[2] Ö. Albayrak Ünal, B. Erkayman, and B. Usanmaz, "Applications of Artificial Intelligence in Inventory Management: A Systematic Review of the Literature," *Arch. Compute. Methods Eng.*, vol. 30, no. 4, pp. 2605-2625, Feb. 2023.

[3] M. D. Capua, A. Ciaramella, and A. De Prisco, "Machine Learning and Computer Vision for the Automation of Processes in Advanced Logistics: The Integrated Logistic Platform (ILP) 4.0," *Procedia Comput. Sci, 217*(9), vol. 217, no. 9, pp. 326-338. doi.org/10.1016/j.procs.2022.12.228

[4] M. Javaid, A. Haleem, R. P. Singh et al., "Exploring Impact and Features of Machine Vision for Progressive Industry 4.0 Culture," *Sensors Int*, vol. 3, no. 5, p. 1000132, Nov. 2021. p. 100132. doi.org/10.1016/j.sintl.2021.100132

[5] Y. Song, F. R. Yu, L. Zhou et al., "Applications of the Internet of Things (IoT) in Smart Logistics: A Comprehensive Survey," *IEEE Internet Things Journal*, vol. 8, no. 6, pp. 4250-4274, Mar. 2021. doi.org/10.1109/jiot.2020.3034385

[6] E. Zunic, S. Delalic, K. Hodzic et al., "Smart Warehouse Management System Concept with Implementation," in *Proc. 2018 14th Symposium on Neural Networks and Applications (NEUREL)*, 2018, pp. 1-5. doi.org/10.1109/neurel.2018.8587004

[7] A. Vukicevic, M. Mladineo, N. Banduka et al., "A Smart Warehouse 4.0 Approach for the Pallet Management Using Machine Vision and Internet of Things (IOT): A Real Industrial Case Study," *Adv. Prod. Eng. Manag*, vol. 16, no. 3, pp. 297-306, Sep. 2021. doi.org/10.14743/apem2021.3.401-

[8] R. Wudhikarn, P. Charoenkwan, and K. Malang, "Deep Learning in Barcode Recognition: A Systematic Literature Review," *IEEE Access*, vol. 10, pp. 8049-8072, Jan. 2022. doi.org/10.1109/access.2022.3143033

[9] Y. Wei, S. Tran, S. Xu et al., "Deep Learning for Retail Product Recognition: Challenges and Techniques," *Comput. Intell. Neurosci*, vol. 2020, pp. 1-23, Nov. 2020.

[10] G. Monteiro, L. Camelo, G. Aauino et al., "A Comprehensive Framework for Industrial Sticker Information Recognition Using Advanced OCR and Object Detection Techniques," *Appl. Sci.*, vol. 13, no. 12, p. 7320, May. 2023. doi.org/10.3390/app13127320

[11] S. Long, X. He, and C. Yao, "Scene Text Detection and Recognition: The Deep Learning Era," *Int. J. Comput. Vis.*, vol. 129, no. 1, pp. 161-184, Aug. 2020. doi.org/10.1007/s11263-020-01369-0

[12] X. Chen, L. Jin, Y. Zhu et al., "Text Recognition in the Wild," *ACM Comput. Surv.*, vol. 54, no. 2, pp. 1-35, Dec. 2021. doi.org/10.1145/3440756

[13] X. Yang, F. Sang, T. Wang et al., "Research on the Influence of Camera Velocity on Image Blur and a Method to Improve Object Detection Precision," in *Proc. The 21 International Conference on Cyber-Physical Social Intelligence* (ICCSI), 2021, pp. 1-6. doi.org/10.1109/iccsi53130.2021.9736224

[14] I. M. Pires, N. M. García, E. Zdravevski et al., "Activities of Daily Living with Motion: A Dataset with Accelerometer, Magnetometer and Gyroscope Data from Mobile Devices," *Data Brief*, vol. 33, p. 106628, Dec. 2020. doi.org/10.1016/j.dib.2020.106628

[15] A. Jain and V. Kanhangad, "Human Activity Classification in Smartphones using Accelerometer and Gyroscope Sensors," *IEEE Sensors Journal*, vol. 18, no. 3, pp. 1169-1177, 2018. doi.org/10.1109/jsen.2017.2782492

[16] M. Shoaib, S. Bosch, Ö. D. İncel et al., "Complex Human Activity Recognition Using Smartphone and Wrist-Worn Motion Sensors," *Sensors*, vol. 16, no. 4, p. 426, Feb. 2016. doi.org/10.3390/s16040426

[17] P. Jiang, D. Ergu, F. Liu et al., "A Review of Yolo Algorithm Developments," *Procedia Comput. Sci.*, vol. 199, pp. 1066-1073, Feb. 2022.

[18] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object Detection Using YOLO: Challenges, Architectural Successors, Datasets and Applications," *Multimedia Tools Appl.*, vol. 82, no. 6, pp. 9243-9275, Aug. 2022. doi.org/10.1007/s11042-022-13644-y

[19] C. H. Kang and S. Y. Kim, "Real-Time Object Detection and Segmentation Technology: An Analysis of the YOLO Algorithm," *JMST Advances*, vol. 5, no. 2-3, pp. 69-76, Sep. 2023. doi.org/10.1007/s42791-023-00049-7

Nalinya Utamapongchai is currently a fourth-year student, pursuing a Bachelor of Engineering in Industrial Engineering and Logistics Systems at Sirindhorn International Institute of Technology (SIIT), Thammasat University, Thailand. Her research interests encompass supply chain management, digital transformation, and data analysis.

Sirawich Ngernsalung is currently a fourth-year student in the Bachelor of Engineering program in Industrial Engineering and Logistics Systems at Sirindhorn International Institute of Technology (SIIT), Thammasat University, Thailand. Additionally, he is enrolled as an external student in the master's program of Logistics and Supply Chain Systems Engineering at SIIT. His research interests include lean automation, digital transformation, and data analytics.

Raveekiat Singhaphandu received a B.S. degree in computer science from the Sirindhorn International Institute of Technology (SIIT), Thammasat University, Thailand, in 2014, and an M.S. degree in informatics from the Technical University of Munich, Germany, in 2017. He is currently pursuing a Ph.D. degree with SIIT, Thammasat University, and the Japan Advanced Institute of Science and Technology, Japan. His research interests include machine learning, computer vision, and applied software engineering in industrial engineering applications, Science and Technology, Japan. His research interests include machine learning, computer vision, and applied software engineering in industrial engineering applications.

Warut Pannakkong is an Associate Professor of the School of Manufacturing Systems and Mechanical Engineering (MSME), Sirindhorn International Institute of Technology (SIIT), Thammasat University, Thailand. He received his B.Engr. in Industrial Engineering and M. Eng. in Logistics and Supply Chain Systems Engineering both from SIIT in 2010 and 2014, respectively. He earned his Ph.D. in Knowledge Science from Japan Advanced Institute of Science and Technology (JAIST) in 2017. His research interests include artificial intelligence for industry, time series forecasting, data mining, machine learning, computer vision in industrial applications, discrete event system simulation and optimization, production planning, and logistics and supply chain management.

The Development of Smart Farming System for Sea Lettuce Cultured Process

Kamolwan Wongwut[1*], Chitraporn Chaisermvong[2], and Daungkamol Angamnuaysiri[3]
[1,2,3]Faculty of Engineering and Industrial Technology, Phetchaburi Rajabhat University, Phetchaburi, Thailand
E-mail: kamolwan.won@mail.pbru.ac.th *, chitraporn.cha@mail.pbru.ac.th, daungkamol.ang@mail.pbru.ac.th

Received: June 21, 2024 / Revised: August 20, 2024 / Accepted: August 28, 2024

***Abstract*—Smart farming represents an advanced approach that integrates information and communication technology into machinery, equipment, and sensors for high-tech farm management. Central to this development is the Internet of Things (IoT), which enables remote connectivity and control. This paper proposes transforming seaweed farming into an intelligent system with IoT-based water oxygen control and environmental monitoring. Utilizing Node-RED, an IoT gateway facilitates device connectivity and provides real-time graphical data through a Node-RED Dashboard. Experiments demonstrate that the oxygen control system, managed by two valves, operates effectively through a web application and Node-RED. The system maintains measurement accuracy with a percentage error of less than 5% for water temperature, air temperature, and humidity. The Node-RED Dashboard offers real-time data on valve control status, water temperature, air temperature, humidity, and oxygen valve switches, transmitted wirelessly based on IoT principles.**

***Index Terms*—Environmental Monitoring, Internet of Things, Node-RED Dashboard, Smart Farming, Water Oxygen Control**

I. Introduction

The Phetchaburi Province area is a model area for strategies and strategic plans for seaweed development. It offers unique opportunities to create careers and significantly increase income for farmers, particularly in the cultivation of grape seaweed in the Ban Laem District. The project is not only supported by the Fisheries Department but also serves as a prototype province in strategy and a strategic plan for developing seaweed. It is not just poised to become one of the model provinces under the "Phetchaburi Model" guidelines. However, it is also on the verge of becoming a prototype model for driving the seaweed project. With its excellent potential, Phetchaburi Province is a beacon of inspiration for the future of seaweed development. The field survey of seaweed cultivation data in Phetchaburi province found that seaweed cultivation is both in soil ponds. Cement ponds and fiberglass tanks affect algae growth, including season, light, temperature, nutrients in seawater, water turbidity, the amount of oxygen in the water, pH, and salinity of the water. The temperature of the water used in raising seaweed is in the range of 25 to 33 degrees Celsius. The air temperature in the rising area Algae is in the range of 25 to 34 degrees Celsius. The salinity of the water is in the range of 21.0 to 33.0 parts per thousand. The pH value has changed from 7.9 to 8.4, and the oxygen value in the water (DO) ranges from 2.73 to 8.25 milligrams per liter. Algae growth rates are higher in April to July. The concentration of nutrients in the seawater used to grow algae is relatively high, including ammonia, nitrite, nitrate, and orthophosphate. Algae can take up these nutrients and use them for growth. In addition, from May to July, there will be a relatively high level of oxygen in the water (DO), resulting in a higher growth rate of algae during this period because algae need to use oxygen to breathe to grow.

Therefore, the researcher has a crucial proposal to develop an intelligent farm system prototype for the sea lettuce cultured process in the closed system. This proposal is vital as it aims to control factors that affect growth, enabling the cultivation of seaweed every season and increasing productivity. The widespread practice of growing sea lettuce algae in cement ponds or plastic tanks is a testament to its convenience and durability. It can be easily cultivated and raised in various ways, such as allowing it to float freely in the water or raising it with aquatic animals. Factors that affect the growth of sea lettuce seaweed include the number of nutrients, light intensity, water salinity, and temperature [1]. The above information supports the researcher's innovative idea of using the agricultural Internet of Things (IoT) has brought new changes to agricultural production. It not only increases agricultural output but can also effectively improve the quality of agricultural products, reduce labor costs, increase farmers' income, and truly realize agricultural modernization and intelligence [2], [3].

The Internet of Things is a network of devices for communicating machine to machine based on wired and wireless Internet. IoT in agriculture is a revolutionary technology that can be applied to agricultural production year-round [4]. The growth of the global population of coupled with a decline in natural resources, farmland, and the increase in unpredictable environmental conditions. These problems are motivators that are driving the agricultural industry to transition to smart agriculture with the application of the Internet of Things [5]. An IoT technology set applied to the acquisition of agricultural data using open-source solutions such as FIWARE and LoRaWAN, which allow extensive customization and integration with advanced weather forecasting, Machine Learning, and real-time dashboard services [6]. This new technology allows devices to connect remotely, paving the way for smart farming [7]-[10]. The development of IoT technologies has played a significant role throughout the farming sector [11], [12], mainly through its communication infrastructure. This includes connecting smart objects, remote data acquisition, using vehicles and sensors through mobile devices and the Internet, cloud-based intelligent analysis [13], [14], interfacing, decision formation, and the automation of agricultural operations [15], [16].

The proposed intelligent farming system for the sea lettuce cultured process is a transformative solution. It is designed to elevate the process into a more efficient and sustainable system. Comprising a control system for adding oxygen to the water and a control system for various environmental factors, it operates on the principle of sensor devices measuring various values and sending the readings to the microcontroller. The system then collects and processes this information, sending the data to Node-RED via the Internet. The results are displayed in real-time with Node-RED Dashboard in Graphic format, heralding a shift from traditional to modern agriculture. This system not only creates competitive opportunities and emphasizes self-reliance but also promotes the application of digital technology, benefiting the economy, society, and community and contributing to the sustainability of the agricultural sector.

II. Materials and Methods

This paper aims to develop the seaweed farming process into an intelligent farm system with water oxygen control and environmental monitoring using Internet of Things technology. The IoT gateway is built with Node-RED to connect devices through APIs (Application Programming Interface) with Node-RED Dashboard in graphical format. Adopting Internet of Things technology to help manage and develop the sea lettuce cultured process into an intelligent farming system. It consists of 3 parts: 1) The valve controls the oxygen supply to the water, 2) The environmental factor measurement system, and 3) The data display system. In designing the system, the system's functionality can connect multiple sets of environmental measurement devices.

Moreover, uses data communication methods according to the IEEE 802.11 wireless communication standard to transmit measurement data to a small data processing and storage device with Raspberry Pi and create a data display system on a web browser. Whose working principle is that the sensor device measures various values and sends the readings to the microcontroller. The system will receive information from sensor devices. Connected via microcontroller to collect information and process information received from sensors. Then, send the data to Node-RED via the internet network with the MQTT protocol and record it in an online database. The MQTT uses the publish/subscribe model and design for devices with low data transmission speeds or low bandwidth; most of the time, the IoT devices look like that. The purpose of MQTT is to make our systems send and receive data more efficiently, including making the device use less energy. In the IOT system, we want to send real-time data and want our devices to use only a little energy unnecessarily. MQTT has a broker (server) and clients (publisher/subscriber). We will call sending data in MQTT publish. The data will be sent to which Topic and receiving information is called subscribe, which means receiving information but will only receive information from the Topic. A broker is an intermediary that will receive all information from Clients (Publishers), no matter what the Topics are, and then manage to send it. Information Clients (Subscribers) who have subscribed to the Topic received can find a Global Broker or a Cloud MQTT Broker on many websites. It can be created within the network using Mosquito Broker, which can be installed on the Raspberry Pi. The MQTT Operation is shown in Fig. 1. Display results on the information system in real-time with the Node-RED Dashboard in graphic form, as shown in Fig. 2.

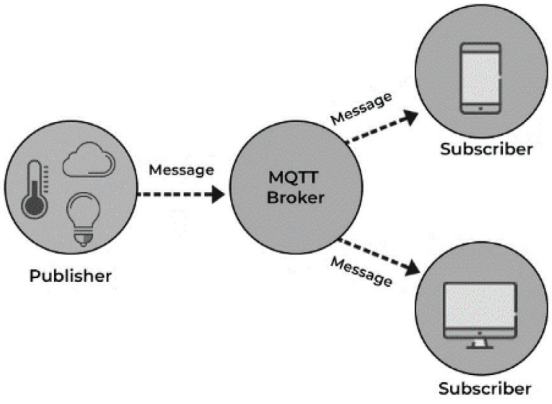


Fig. 1. The MQTT operation

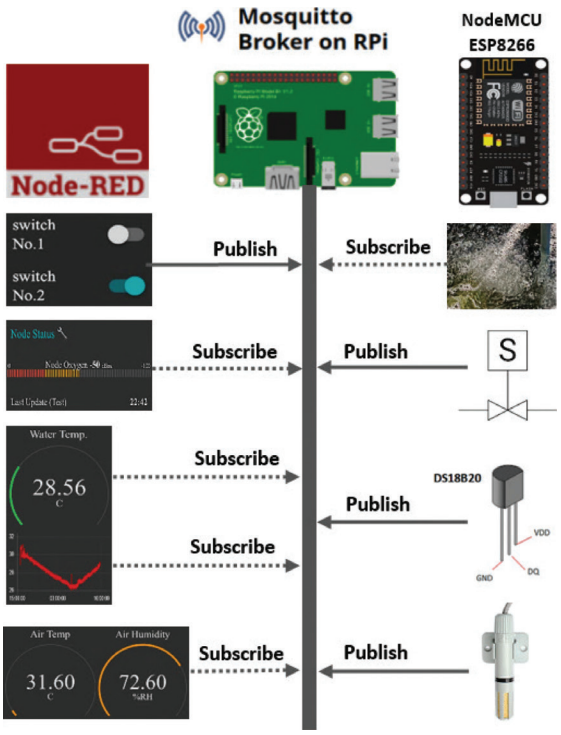


Fig. 2. The design of the proposed embedded system for a smart farm with a Node-RED dashboard

A. The Valve Controls the Oxygen Supply to the Water for Sea Lettuce Cultured Process

The valve control system, a pivotal element of our cutting-edge farm, is designed for optimal efficiency. It comprises a valve control kit that precisely regulates the water oxygen levels. This valve can be conveniently controlled online via a web server using a device such as a solenoid valve or a magnetic controller. The solenoid valve, a YCWS3 type with an orifice of 2.5 mm and a pressure range of 0 to 0.7 Mpa operates on a DC 12 V power supply, as depicted in Fig. 3 to Fig. 5.

Fig. 3. The solenoid valve controls system

Fig. 4. The design of the valve controls the oxygen supply to the water for sea lettuce cultured process

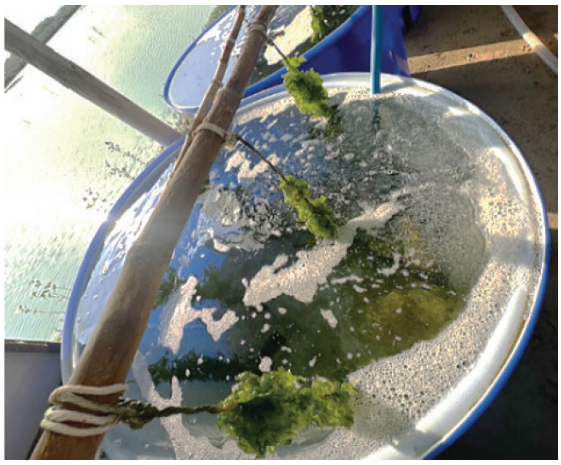


Fig. 5. Addition of oxygen to the water for the sea lettuce cultured process

B. The Environmental Factor Measurement System

The environmental factor measurement system is shown in Fig. 6. There is a principle of considering equipment selection from factors related to algae growth and the appropriate usage characteristics for the actual use environment.

Fig. 6. The design of measuring environmental factors system

The equipment used in this study includes:

1. NodeMCU V2 is an IoT experimental board with ESP8266 chips that provide convenient Wi-Fi and wireless Internet access. The ESP8266 chips control devices and receive data over wireless networks efficiently.

2. Temperature Sensor is DS18B02 Model; Probe waterproof sealed with stainless steel; Input voltage DC 3.0 to 5.5 V; Temperature range from -55°C to 125°C; temperature accuracy ±0.5°C from -10°C to 85°C.

3. Temperature and Humidity Sensor is AM2305 Model; Input voltage DC 3.5 to 5.5 V; temperature range from -40°C to 125°C; humidity range from 0 to 99.9%RH; Temperature accuracy ±0.3°C; Humidity accuracy ±2%RH; temperature resolution 0.1°C; humidity resolution 0.1%RH.

Environmental factors measuring instruments the data from the above sensors must be transmitted to the processing equipment to write a measurement result interpretation program and compare it to a standard measurement instrument. In this study, the defined error was ±1°C for the water temperature measurement, the error was ±1°C for the air temperature measurement, and the humidity measurement error was ±3%RH.

C. The Data Display System

Currently, communication on a computer network has selected a tiny Raspberry PI as the central processing device of the system. Using Linux as the network operating system, as well as using all open-source programs. In this study, our system incorporates an IoT gateway with Node-RED for device connection via APIs (Application Programming Interface). This gateway is a standout feature of our system, excelling in real-time data processing. It receives data from sensor devices, connects them through microcontrollers, processes the data from the sensor, then sends it to Node-RED over the internet using the MQTT protocol and saves it to an online database. The Gateway IoT with Node-RED, as shown in Fig. 7.

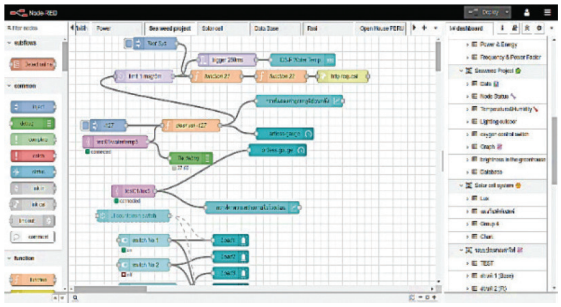


Fig. 7. The Gateway IoT with Node-RED

III. Results and Discussion

This section presents the results obtained from the proposed embedded system in the sea lettuce process for smart farms, including (1) The water oxygen control system, (2) The monitoring environmental factors system, and (3) The data display system

A. The Results of the Oxygen Control System

Fig. 8 shows that the oxygen control system can be operated using two valves, switch No. 1 and No. 2, controlled through a web application, Node-RED, using IoT technology and internet network connectivity. Communication is supported through the MQTT protocol, so the control valve device can be connected to a server or web service that provides web application services. Users can access web browsers on devices that require control, such as computers, smartphones, etc. The user can control the operation of oxygen supply valves, such as on/off and adjust fill levels, through web applications quickly and easily without direct access to the device.

The Web application control can provide several benefits: (1) Users can quickly and easily control the oxygen supply in water through a web browser without technical knowledge. (2) Remote access allows users to control their systems wherever they have an Internet connection, not necessarily near the system they want to control. (3) Users can check the real-time status of their oxygen-in-water filling system without having to be near the system. Therefore, using web applications is a convenient and effective way to control the oxygen-in-water filling system in IoT systems without direct access to equipment in potentially risky or inconvenient locations.

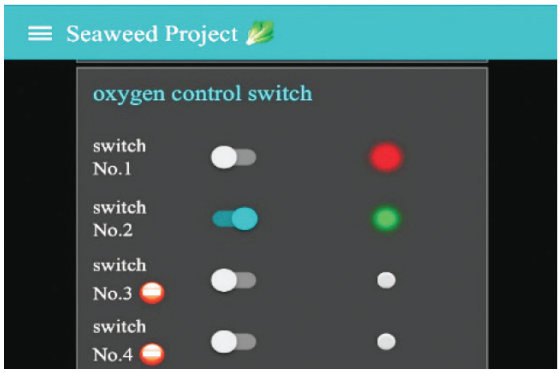


Fig. 8. The switch for the valve controls oxygen

B. The Results of the Environmental Factor Measurement System

The environmental factor measurement system, driven by IoT devices, is a testament to precision. It meticulously measures water temperature, air temperature, and humidity in the greenhouse's seaweed culture process. The defined error for water temperature was ±1°C, for air temperature was ±1°C, and for humidity was ±3%RH. This level of accuracy instills confidence in the system's performance, ensuring reliable and consistent results.

Table I measured water temperature measurements compared to the AZ8371 Instrument Salinity/Temp

Meter; Temperature range from 0°C to 50.0°C; Temperature accuracy ±0.5°C; temperature resolution 0.1°C; Normalization temperature fixed at 25°C. An error range, which represents the difference between the measured value and the true value, is found to be between 0.1-0.4 °C. This means that our measurements are consistently within this range of the true value, indicating a high level of precision.

TABLE I
COMPARISON OF WATER TEMPERATURE MEASUREMENT

Day Time	Experiment (°C)	Measurement (°C)	Error (°C)
15/02/2024 07:30 a.m.	26.3	26.1	0.2
18/02/2024 10:05 a.m.	26.8	26.4	0.4
21/02/2024 1:20 p.m.	28.8	28.5	0.3
24/02/2024 11:00 a.m.	27.4	27.1	0.3
27/02/2024 09:15 a.m.	26.2	26.1	0.1
1/03/2024 12:30 p.m.	28.1	27.8	0.3
4/03/2024 3:40 p.m.	29.5	29.3	0.2
7/03/2024 08:45 a.m.	26.7	26.5	0.2
10/03/2024 10:15 a.m.	26.7	26.4	0.3
13/03/2024 2:50 p.m.	28.6	28.2	0.4
16/03/2024 1:00 p.m.	28.1	28.0	0.1
19/03/2024 1:30 p.m.	28.9	28.5	0.4
22/03/2024 11:00 a.m.	26.5	26.3	0.2
25/03/2024 12:45 p.m.	28.6	28.5	0.1

Table II and Table III compare the temperature and humidity measurements in the greenhouse with the standard AS808 Hygrometer. The temperature range was -40°C to 70°C, and the humidity ranged from 20%RH to 90%RH. The temperature resolution was 0.1°C (0.1°F), and the humidity resolution was 1%RH. The temperature accuracy was ±1°C (1.8°F), and the humidity accuracy was ±5%RH. The error range was between 0.1-0.6 °C and 0.9 - 2.8-% RH. These results reaffirm the effectiveness and accuracy of the IoT devices developed in this study, providing the audience with a sense of reassurance about the system's capabilities.

TABLE II
COMPARISON OF AIR TEMPERATURE MEASUREMENT

Day Time	Experiment (°C)	Measurement (°C)	Error (°C)
15/02/2024 07:30 a.m.	25.3	24.8	0.5
18/02/2024 10:05 a.m.	28.7	28.4	0.3
21/02/2024 1:20 p.m.	32.1	31.7	0.4
24/02/2024 11:00 a.m.	29.8	29.5	0.3
27/02/2024 09:15 a.m.	27.4	26.8	0.6
1/03/2024 12:30 p.m.	31.2	30.8	0.4
4/03/2024 3:40 p.m.	32.3	32.1	0.2
7/03/2024 08:45 a.m.	26.8	26.2	0.6
10/03/2024 10:15 a.m.	28.5	28.2	0.3
13/03/2024 14:50 p.m.	32.4	32.2	0.2
16/03/2024 13:00 p.m.	31.1	31.0	0.1
19/03/2024 13:30 p.m.	32.5	31.9	0.4
22/03/2024 11:00 a.m.	30.8	30.5	0.3
25/03/2024 12:45 p.m.	31.6	31.2	0.4

TABLE III
THE DISPLAY OF THE REAL-TIME DATA ON THE NODE-RED DASHBOARD

Day Time	Experiment (%RH)	Measurement (%RH)	Error (%RH)
15/02/2024 07:30 a.m.	87.3	85.1	2.2
18/02/2024 10:05 a.m.	80.5	78.5	2.0
21/02/2024 1:20 p.m.	65.8	64.3	1.5
24/02/2024 11:00 a.m.	75.4	74.5	0.9
27/02/2024 09:15 a.m.	82.3	81.2	1.1
1/03/2024 12:30 p.m.	72.6	70.5	2.1
4/03/2024 3:40 p.m.	62.7	60.9	1.8
7/03/2024 08:45 a.m.	84.8	82.5	2.3
10/03/2024 10:15 a.m.	79.5	78.2	1.3
13/03/2024 2:50 p.m.	63.5	60.8	2.7
16/03/2024 1:00 p.m.	68.9	66.1	2.8
19/03/2024 1:30 p.m.	64.5	63.5	1.0
22/03/2024 11:00 a.m.	78.3	74.8	3.5
25/03/2024 12:45 p.m.	70.6	68.9	1.7

C. The Display of the Real-Time Data on the Node-RED Dashboard

The software ensures precise data storage and display with the cloud server on Raspberry Pi, powered by Node-RED. The real-time measurement data, with its high level of accuracy, is transmitted via a wireless system of devices, all based on the IoT concept, which involves interconnected devices exchanging data. This reliable system allows for the display of real-time data on the Node-RED dashboard, providing an exact view of node status including valve control oxygen, water temperature, air temperature, humidity, lighting, and the switch of the valves control oxygen, as depicted in Fig. 9.

Fig. 9. The display of the real-time data on Node-RED

D. The Comparison of the Percentage Error

In the experiment, fault tolerance is configured as not exceeding 5 percent for the water temperature, the air temperature, and the humidity measurement. The calculation of the error was done between a measurement from a highly calibrated instrument and the experiment system, which is a replica of the real-time data display system. The results were displayed and compared to determine the percentage difference, as in equation (1).

$$\text{Error} = \left| \frac{\text{Measuring}_{Real} - \text{Measuring}_{\text{Exp}}}{\text{Measuring}_{Real}} \right| \times 100 \qquad (1)$$

TABLE IV
COMPARISON OF PERCENTAGE ERROR

Day Time	$TEMP_W$ (%)	$TEMP_A$ (%)	Humidity (%)
15/02/2024 07:30 a.m.	0.77	2.02	2.59
18/02/2024 10:05 a.m.	1.52	1.06	2.55
21/02/2024 1:20 p.m.	1.05	1.26	2.33
24/02/2024 11:00 a.m.	1.11	1.02	1.21
27/02/2024 09:15 a.m.	0.38	2.24	1.35
1/03/2024 12:30 p.m.	1.08	1.30	2.98
4/03/2024 3:40 p.m.	0.68	0.62	2.96
7/03/2024 08:45 a.m.	0.75	2.29	2.79
10/03/2024 10:15 a.m.	1.14	1.06	1.66
13/03/2024 2:50 p.m.	1.42	0.62	4.44
16/03/2024 1:00 p.m.	0.36	0.32	4.24
19/03/2024 1:30 p.m.	1.40	1.88	1.57
22/03/2024 11:00 a.m.	0.76	0.98	4.68
25/03/2024 12:45 p.m.	0.35	1.28	2.47

From the experiment, the Table IV Comparison of percentage error, it was found that the water temperature measurement on 18/02/2024 at 10:05 a.m. had the maximum error value, equal to 1.52%. The date 25/03/2024 at 12:45 p.m. had a minor error value equal to 0.35%, measuring the air temperature in the greenhouse on 07/03/2024 at 08:45 a.m. has a maximum error equal to 2.29%, and on 16/03/2024 at 1:00 p.m. has the most negligible error value is equal to 0.32%, the humidity measurement in the greenhouse on 22/03/2024 at 11:00 a.m. has the maximum error value equal to 4.68% and the date 24/02/2024 at 11:00 a.m. has the most negligible error value is equal to 1.21%.

IV. CONCLUSION

This paper delves into the tools and equipment used in applications of wireless sensors in IoT agriculture, highlighting the anticipated challenges faced when merging technology with conventional farming activities. The environmental measurements obtained from this system boast an impressive accuracy of at most 5%. Data transmission between the measuring node and the central processor via a wireless network is not just efficient, but remarkably easy, allowing for seamless measurement and recording. The system can be viewed backward when connected to the Internet, providing a comprehensive overview at a low cost. The embedded systems measure the environment for smart farms, and this design can measure and report various environmental values, which is still needed. Further development of other environmental sensors to provide flexibility in use as appropriate. However, the system has been designed to support data communication according to international standards, which can connect this system to other systems, instilling confidence in the accuracy and efficiency of IoT devices in environmental monitoring.

ACKNOWLEDGEMENT

This work was supported by (1) Phetchaburi Rajabhat University (PBRU), (2) Thailand Science Research and Innovation (TSRI), and (3) National Science Research and Innovation Fund (NSRF) (Fundamental Fund: Fiscal year 2024, Title: Developing a Prototype Smart Farm System for Cultivating Seaweed using Digital Technology).

REFERENCES

[1] S. Worasing, "Effect of Salinity Level to Growth Rate of Sea Lettuce (Ulva Rigida C. Agardh, 1823)," Dept. of Fisheries. Trat, TH, Rep. No. 35/2008, Aug. 2008.

[2] J. Xu, B. Gu, and G. Tian, "Review of Agricultural IoT Technology," *Artificial Intelligence in Agriculture*, vol. 6, no. 1, pp. 10-22, Jan. 2022.

[3] A. Anand, N. K. Trivedi, V. Gautam et al., "Applications of Internet of Things (IoT) in Agriculture: The Need and Implementation," in *Proc. International Conference Advancement in Data Science, E-learning and Information Systems (ICADEIS)*, 2022, pp. 1-5.

[4] W. S. Kim, W. S. Lee, and Y. J. Kim, "A Review of the Applications of the Internet of Things (IoT) for Agricultural Automation," *J. Biosyst. Eng.*, vol. 45, no. 4, pp. 385-400, 2020.

[5] M. R. M. Kassim, "IoT Applications in Smart Agriculture: Issues and Challenges," *in Proc. IEEE Conference on Open Systems (ICOS)*, 2020, pp. 19-24.

[6] A. Heideker, D. Ottolini, I. Zyrianoff et al., "IoT-Based Measurement for Smart Agriculture," in *Proc. IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor)*, 2020, pp. 68-72.

[7] G. S. Nagaraja, A. B. Soppimath, T. Soumya et al., "IoT Based Smart Agriculture Management System," in *Proc. 4th International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*, 2019, pp. 1-5.

[8] S. P. Jaiswal, V. S. Bhadoria, A. Agrawal et al., "Internet of Things (IoT) for Smart Agriculture and Farming in Developing Nations," *International Journal of Scientific & Technology Research*, vol. 8, no. 12, pp. 1049-1056, Dec. 2019.

[9] I. M. Marcu, G. Suciu, C. M. Balaceanu et al., "IoT Based System for Smart Agriculture," in *Proc. 11th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, 2019, pp. 1-4.

[10] M. Ayaz, M. Ammad-Uddin, Z. Sharif et al., "Internet-of-Things (IoT)-Based Smart Agriculture: Toward Making the Fields Talk," *IEEE Access*, vol. 7, pp. 129551-129583, Aug. 2019.

[11] J. Pitakphongmetha, W. Suntiamorntut, and S. Charoenpanyasak, "Internet of things for Aquaculture in Smart Crab Farming," *Journal of Physics: Conference Series*, vol. 1834, no. 4, Nov. 2021.

[12] V. Dankan Gowda, M. S. Prabhu, M. Ramesha et al., "Smart Agriculture and Smart Farming using IoT Technology," *Journal of Physics: Conference Series*, vol. 2089, no. 1, pp. 1-9, Nov. 2021.

[13] S. Namani and B. Gonen, "Smart Agriculture Based on IoT and Cloud Computing," in *Proc. 3rd International Conference on Information and Computer Technologies (ICICT)*, 2020, pp. 553-556.

[14] E. Elbasi, N. Mostafa, Z. AlArnaout et al., "Artificial Intelligence Technology in the Agricultural Sector: A Systematic Literature Review," *IEEE Access*, vol. 11, pp. 171-202, Jan. 2023.

[15] A. Rehman, T. Saba, M. Kashif et al., "A Revisit of Internet of Things Technologies for Monitoring and Control Strategies in Smart Agriculture," *Agronomy*, vol. 12, pp. 127, Jan. 2022.

[16] B. M. Zerihun, T. O. Olwal, and M. R. Hassen, "Design and Analysis of IoT-Based Modern Agriculture Monitoring System for Real-Time Data Collection," *Computer Vision and Machine Learning in Agriculture*, vol. 2, pp. 73-82, Jan. 2022.

Kamolwan Wongwut (Member, IEEE, Thailand Section) received a B.Sc. degree in electrical engineering from Naresuan University, Phitsanulok, Thailand, in 2011 and a M.Sc. degree in electrical engineering from Chiang Mai University, Chiang Mai, Thailand, in 2016. She is currently pursuing a doctorate degree in the faculty of Electrical Engineering at King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand. She was a lecturer from 2017 to the present at the Department of Electrical Engineering, Faculty of Engineering and Industrial Technology, Phetchaburi Rajabhat University, Phetchaburi, Thailand. Her research interests are in power system optimization, power system planning, and application of neural networks in power systems, and renewable energy.

Chitraporn Chiasermvong received the B.F.A. degree in Interior Design in 2012 from Rajamangala University of Technology Rattanakosin, Bangkok, Thailand. In 2018 she received the M.Arch. degree in Interior Architecture from King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand. She was a lecturer from 2017 to the present at the Faculty of Interior Architecture, Phetchaburi Rajabhat University, Phetchaburi, Thailand. Her research interests are in Architecture, Interior Design augmented reality, virtual Reality, and mixed reality.

Daungkamol Angamnuaysiri received the B.S.I.Ed. degree in Engineering Education (Telecommunications Engineering) in 2015 from King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand. In 2016, she received the M.SC.LEd. degree in Electrical Communications Engineering from King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand. She was a lecturer from 2017 to present at the Faculty of Engineering and Industrial Technology, Phetchaburi Rajabhat University, Phetchaburi, Thailand. Her research interests are microcontrollers, artificial intelligence, augmented reality, virtual reality, and mixed reality.

The Morphology Associated with Harvesting Stages of Siam Red Ruby Pumelo (*Citrus grandis*)

Nopparat Tatmala[1*], Samak Kaewsuksaeng[2], Chairat Burana[3], and Kornlawat Tantivit[4]
[1]Department of Crop Production, Faculty of Agricultural Technology,
Rajamangala University of Technology Thanyaburi, Pathum Thani, Thailand
[2]Department of Plant Science, Faculty of Technology and Community Development,
Thaksin University, Phatthalung Campus, Phatthalung, Thailand
[3,4]Faculty of Innovative Agriculture and Management, Panyapiwat Institute of
Management, Nonthaburi, Thailand
E-mail: nopparat_t@rmutt.ac.th*, samak@tsu.ac.th, chairatbur@pim.ac.th, kornlawattan@pim.ac.th

Received: November 14, 2024 / Revised: December: 15, 2024 / Accepted: December 17, 2024

Abstract— **This study aims to evaluate fruit morphological characteristics and potential production of 'Siam Red Ruby' pumelo from Pak Panang district, Nakhon Si Thammarat Province compared between normal and senescence fruit peel. A three-year-old tree was selected to evaluate fruit morphological characteristics and a harvesting date or time at 90, 120, 150, 180, and 210 Days After Fruit Set (DAFS). The result showed anatomical association to the senescence of 'Siam Red Ruby' pummelo using the peel of normal and senescence zones using compound microscopy. The 'Siam Red Ruby' pummelo was shown a green color inside a peel-90 days after fruit setting and in a peel showed a trichome or hair covering all the fruit (100%) then decreased during fruit development until 210 days after fruit setting showed the hair cover 20% in all fruit compared with others stage. During the maturity stage, the pulp was yellow-colored from 90 to 120 days after the fruit set. From 150 days after the fruit set, the pulp turned red gradually. In addition, the structure of the peel was changed during senescence. The first day of the peel or normal zone showed high firmness and peel green color. In the senescence zone, the peel changed to a yellow color. In addition, Titratable Acidity (TA) decreased from 90 to 210 days after fruit setting and related with Total Soluble Solid (TSS) increased from mature fruit (90 days after fruit setting) to ripening fruit (210 days after fruit setting). In reflective sheet treatment, clouds induce the red color in the pulp of 'Siam Red Ruby' *pumelo.***

Index Terms— Siam Red Ruby Pumelo, Senescence, Morphology

I. Introduction

'Siam Red Ruby' pumelo (*Citrus grandis*) is plantation the Pakpanang district, Nakhon Si Thammarat province, Thailand. It is a Geographical Indication (GI) product of Thailand and is a popular pumelo cultivar in the premium fruit marketplace. The external appearance of the 'Siam Red Ruby' pumelo fruit is characterized by a dark green colored peel with soft hair and clear oil glands. The internal appearance of the 'Siam Red Ruby' pumelo fruit is characterized by a dark pink to red color and a sour-sweet taste [1]. Recently, the demand for 'Siam Red Ruby' fruit has increased by 40% in domestic and 60% in international markets, especially markets in China, Taiwan, Hong Kong, Malaysia, Singapore, and Brunei. Today, the cultivation site of 'Siam Red Ruby' pumelo is expanding to commercial, continuous plantation operations due to the high value and marketing needs [2]. The 'Siam Red Ruby' pumelo is rich in bioactive compounds such as ascorbic acid (vitamin C), α-tocopherol (vitamin E), and flavonoids (i.e., naringin, narirutin, hesperidin, and neohesperidin). In addition, 'Siam Red Ruby' pumelo juice has high amounts of fructose, glucose, and sucrose, at 6.07, 4.52, and 2.09 g L^{-1}, respectively [3]. Although fruit growers had good experience in pumelo orchard management, they lacked relevant information on the harvesting index of 'Siam Red Ruby' pumelo in these areas. The harvesting index information of "Tabtim Siam" pumelo is limited as most research has only focused on the effect of tree age and fruit age on fruit quality in Pakpanang district, Nakhon Si Thammarat province. For pumelo var. Tabtim Siam, fruit quality of 6 to 8-year-old trees was suitable for harvesting at 210 days after fruit set, reported [4] that the harvesting index of 4 and 7 years old of pumelo were 160 and 220 days after fruit set, respectively. In addition, the fruit quality of pumelo depends on the planting area such as soil nutrients, microclimate, and cultural practices of the grower. Flavors and some morphological characteristics of pumelo fruit in each area were different [5] Thus, the objective of this research was to evaluate fruit

morphological characteristics and potential production of 'Siam Red Ruby' pumelo in Pak Panang district, Nakhon Si Thammarat provinces.

II. Materials and Methods

A. Plant Material

The experiment was conducted in Pakpanang district, Nakhon Si Thammarat province. The 'Siam Red Ruby' (*Citrus grandis*) at the harvesting stages (90, 120, 150, 180, and 210 days), was harvested and transported to the laboratory of Plant Science, Faculty of Technology and Community Development, Thaksin University, Phatthalung campus. Fruit samples were selected for uniformity of color and size and did not show the disease. These fruits of normal and senescence are to be determined by morphology using a microscope.

B. Study on the Morphology Using Compound Microscopy

In this study, we are using the peel of normal and senescence zones (210 days after fruit set) to observe changes in the physical morphology surface of the 'Siam Red Ruby' pumelo during senescence (storage at 25°C for 1 month). Cutting the tissue area to be studied for 5×5 cm. Then a sample on a microscope [6] the software for recording pictures using LASV 4.5.

C. Study on the Morphology Using Scanning Electron Microscope

The peel of 'Siam Red Ruby' pumelo has been cut to size 1×1 cm for normal and senescence zones. In this study, we have conducted fixation of the sample with Formaldehyde Acetic Acid (FAA) for SEM, components of FAA include Ethanol 25 ml and acetic acid 5 ml FAA 5 ml distilled water 5 ml Remove FAA, then dehydration of sample with T-Butanol (TBA) at different concentrations. The concentration TBA is 10, 20, 35, 55, 75, and 100%, these concentrations for 2 hr. After dehydration, keep the sample at -20°C for 24 hr before SEM analysis.

Color measurement with a colorimeter (Model CR300, Minolta, Japan). The CIE 1976 L* a* b* color scale was adopted. The hue angle (Ho) was calculated as Ho = arctangent (b*/ a*). Color readings were taken three times at the equatorial region of each fruit.

The Total Soluble Solids (TSS) of fruit juice were determined by using a Hand refractometer (Model N; Atago Co., Tokyo, Japan) and reported as percent soluble solids in fruit juice. The Titratable Acidity (TA) of fruit juice was determined by titrating 1 mL of fruit juice diluted 10 times with distilled water against 0.1 mol L-1 NaOH, using 1-2 drops of 1% (v/v) phenolphthalein as an indicator of the reaction end-point. TA was expressed as percent citric acid (meq. citric acid = 0.064).

Estimation of Chlorophyll: Five hundred mg of fresh leaf material was taken and ground with the help of a pestle and mortar with 10 ml of 80% acetone. The homogenate was centrifuged at 3000 rpm for 15 minutes the supernatant was stored. The residue was re-extracted with 5 ml of 80% acetone. The extract was utilized for chlorophyll estimation. Absorbance was read at 645 and 663 nm in the UV-spectrophotometer.

Texture Analyzer (TA-XT2) was used to measure the penetration force needed to press a plunger with a diameter of 4 mm 5 mm into the fruit at a speed of 1.7 mm s^{-1}. The fruit was cut into halves and the firmness was tested for both halves. The readings were in Newton (N).

D. Statistical Analysis

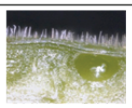
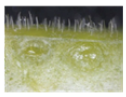
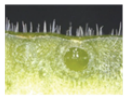
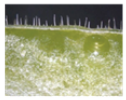
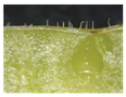
All values were shown in the present study as the mean ± Standard Error (SE) for three replicates. The data were analyzed, and Tukey's Honest Significant Difference (HSD) test was used to compare the means at $P < 0.05$. Calculations were performed using JMP software (SAS Institute, Cary, NC).

III. Results

A. The Morphology Associated Change with the Senescence of 'Siam Red Ruby' Pumelo

All Anatomical associated with the senescence of 'Siam Red Ruby' pumelo using the peel of normal and senescence zone using compound microscopy. The 'Siam Red Ruby' pumelo was shown a green color inside a peel 90 days after fruit setting and a peel showed a trichome or hair covering all the fruit (100%) then decreased during fruit development until 210 days after fruit setting showed the hair cover 20% in all fruit compared with immature stage. During the development or maturity stage, the pulp was yellow-colored from 90 to 120 days after the fruit set. From 150 days after the fruit set, the pulp gradually turned red. In addition, the structure of the peel was changed during senescence. The first day of the peel or normal zone showed high firmness and peel green color. In the senescence zone, the peel changes to the yellow color (Table I). While the changes of hue angle were in parallel with color development in the pulp. In addition, the Titratable Acidity (TA) was constant from 90 to 150 days after the fruit setting and then decreased significantly from 180 days after the fruit set. On the other hand, the Total Soluble Solid (TSS) value of the fruit juice increased significantly during the fruit process. Among them, the carotenoid in the pulp of 'Siam Red Ruby' pumelo, increased rapidly during the stage and accounted for the maturation stage (210 days after the fruit setting) (Table I).

TABLE I
CHANGE OF PHYSICAL MORPHOLOGY AND INTERNAL QUALITY ON PEEL AND PULP COLOR CHANGE DURING THE 'SIAM RED RUBY' PUMELO FRUIT DEVELOPMENT STAGE

Fruit stage	Outer pericarp section	Time (d)	Firmness (N)		Hue angle		Total Chlorophyll (mg/100 FW)		Total Carotenoid (mg/kg^{-1} FW)		%TA	TSS (°Brix)
			Peel	Pulp	Peel	Pulp	Peel	Pulp	Peel	Pulp		
		90	43.64	11.39	110.26	65.37	18.55	4.10	1.21	2.84	0.56	9.2
		120	43.75	10.75	109.12	65.59	18.21	3.22	2.35	4.55	0.58	9.8
		150	42.77	10.80	109.2	55.44	17.86	3.54	2.33	4.87	0.55	9.7
		180	41.38	9.32	108.35	45.03	17.23	3.21	2.89	5.34	0.34	10.35
		210	39.42	8.12	104.67	38.90	15.23	3.45	3.01	5.79	0.28	12.54

The data is not statistical and SE or SD (±)

B. Anatomical of 'Siam Red Ruby' Pumelo During Senescence

The anatomical and surface of the peel normal cell on the upper show differences with senescence cells. The peel normal cell of 'Siam Red Ruby' pumelo showed freshness and structure, and the shape of the cells was round and turgid of cells. , we found stomata were turgid cells, especially guard cells, these show close stomata (Fig. 1). While the peel senescence cell of 'Siam Red Ruby' pumelo as shown in Fig. 1 and Fig. 2, The peel senescence cell collapsed of surface cells and the arrangement of cells was incomplete, the peel cell found loss of turgid, show stomata of petal blackening cell were opening more than peel normal cell and lost turgid of guard cell (Fig. 2).

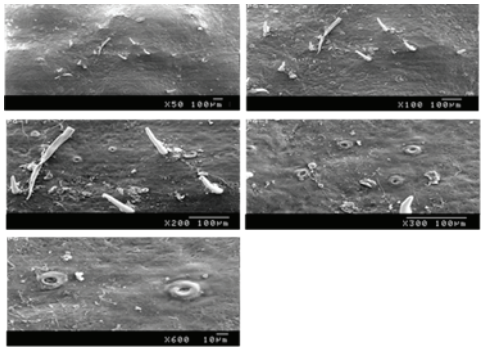


Fig. 1. Change of anatomical in the peel of 'Siam red ruby' pumelo by using scanning electron microscope; Sem on the normal fruit.

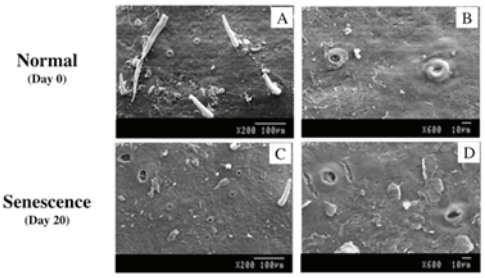


Fig. 2. Change of anatomical in the peel of 'Siam Red Ruby' pumelo by using scanning electron microscope; Sem on the senescence of the fruit (A, B, C, & D).

IV. DISCUSSION

The results of this study showed significant differences in the pulp color, total carotenoid content, and carotenoid composition in 'Siam Red Ruby' pumelo during the development stage. During fruit development, there was a difference in pulp color from 60 to 210 days after fruit setting. The hue angle decreased sharply during this time. In addition, color development in the pulp was related to the carotenoid content, with the yellow color generally being greater in immature fruit and reddening occurring during the development stages due to changes in the carotenoid composition. There were differences during fruit development in TSS and TA values (Table I). TSS increased significantly during fruit development. During fruit development, sugar, and pigment content

accumulated while acid content decreased in the pulp of fruit. Determining the different sugar/acid ratios during ripening is often used as a major postharvest maturity. As TSS content increases, this parameter is a very practical index of internal fruit quality. The understanding of the physiological and biochemical determinants for TSS and TA content in fruits will allow the enhancement of fruit quality during growth and pre and post-harvest practices, the improvement of citrus fruit sweetness and fruit acidity, the characterization of physiological disorders that depend on TSS and TA fruit content and physicochemical composition of the fruits, especially ascorbic acid, TSS and TA depends on the maturity stage. Moreover, a high correlation between TSS and TA has been described Obenland et al. [7].

Stomata are pores in the gastight waxy cuticle that cover the outer surface of aerial parts of plants. They make the uptake of CO2 possible, which is needed for photosynthesis. At the same time, water vapor will leave the plant via the stomata [8]. To optimize photosynthesis, while at the same time preventing excess water loss, stomata control their opening by a signaling network of pathways that respond to environmental conditions such as light and darkness, water Vapor Pressure Deficit (VPD), temperature, CO2, and ethylene. Water loss is one of the most obvious changes in harvested vegetables, often limiting marketing life, and a negative water balance (uptake of water is insufficient to compensate for transpiration) is one of the most important reasons for the end of the vase life of cut flowers [9]. In addition, the postharvest quality of some fruits was negatively affected by water loss. In leafy vegetables, stomata are portals that make the invasion of bacteria into the inner tissue possible and protect in that way bacteria from sanitizers in washing solutions with the risk of food-borne bacterial diseases. This chapter discusses how several environmental factors, during preharvest cultivation and postharvest storage, influence stomata closing control in harvested fruits. Also, the role of the number of stomata and their variability between genotypes and due to cultivation conditions are discussed in relation to postharvest life. One of the most striking factors is low VPD (high humidity) during the growth of plants: after exposure for several days to low VPD, the control of stomata closure is largely disturbed; stomata do not respond anymore to stimuli that normally induce closure. This malfunctioning is persistent and results in high water loss afterward in the harvested products. Fast cooling of produce can close the stomata of some crops, while in others, the stomata stay open until wilting [10]

Plant stomata, consisting of a pair of guard cells, are dynamic structures that open or close to modulate gas exchange and water loss and allow plants to respond appropriately to diverse pathogens invasion. Therefore, stomata play essential roles in abiotic and biotic stress responses [11]. The regulation of stomata movement is complex, and as time progresses, researchers discover new signaling elements that make the signaling networks of stomata movement more complex [12]. In recent years, in addition to plant growth and development, plant small signaling peptides have been implicated in stomata aperture regulation [13].

V. Conclusion

The specific fruit shape of this cultivar was obovoid and piriform. The fruit peel color was light green to dark green with a hairy cover. Peel color development of 'Siam Red Ruby' pumelo developed faster during fruit development. The pulp color was dark red similar to a ruby. The harvesting index of the 5-6-year-old pomelo tree in these production regions was at 6 months after the fruit set presenting the highest overall acceptance. In 'Siam Red Ruby' pumelo, Total Soluble Solid (TSS) and Titratable Acidity (TA) were 10.4 °Brix, 0.56% and 20.3, respectively.

Acknowledgment

This work was supported by the Thailand Research Fund under the Research and Researchers for Industries (RRI) to Nopparat Tatmala (PHD60I0076).

References

[1] C. Preecha and S. Na Nakorn, "Fruit Growth And Development of Pummelo cv. Tubtim Siam at the Difference Tree Age and Fruit Age for the Optimal Harvesting Time under the Climate Variation," *International Journal of Agricultural Technology*, vol. 14, no. 7, pp. 1685-1692, Dec. 2018.

[2] N. Tatmala, G. Ma, L. Zhang et al., "Characterization of Carotenoid Accumulation and Carotenogenic Gene Expression During Fruit Ripening in Red Colored Pulp of 'Siam Red Ruby' Pumelo (*Citrus grandis*) Cultivated in Thailand," *The Horticulture Journal*, vol. 89, no. 3, pp. 237-243, Feb. 2020.

[3] S. Kaewsuksaeng and P. Sangwanangkul, "Bioactive Compound Contents in Citrus Family Locally Cultivated in Southern Thailand," *Khon Kaen Agriculture Journal*, vol. 43, no. Special 1, pp. 801-804, Jan. 2015.

[4] C. Preecha and S. Na Nakorn, "Effects of Fruit Harvest Periods, Tree Ages and Fruit Development Periods on Fruit Quality of Pomelo cv. Tubtim Siam," *Rajamangala University of Technology Srivijaya Research Journal*, vol.13 no.1, pp. 191-200, Apr. 2011.

[5] S. Kongsri and P. Nartvaranant, "Fruit Morphological Characteristics and Fruit Quality of Pomelo cv. Tabtim Siam Grown in Nakhon Pathom and Nakhon Si Thammarat Provinces," *Journal of Thai Interdisciplinary Research*, vol. 15, no. 1, pp. 5-11, Mar. 2019.

[6] G. H. Brown and R. Barmore "Effect of Chlorophyllase Activity and Chlorophyll Content in Calmodin Rind Tissue," *HortScience*, vol. 10, no. 6, pp. 595-596, Dec. 2012.

[7] Y. Maeda, H. Kurata, and K. Shimikawa, "Chlorophyll Catabolism in Ethylene Treated Citrus Unshui Fruits," *Journal of Japanese Society of Horticultural Science*, vol. 67, no. 4, 497-502. Sep. 2008.

[8] Jr. M. E. Salviet and L. L. Morris. *Chilling Injury of Horticultural Crops*. Florida: CRP Press, 2010, pp. 3-15.

[9] D. Obenland, S. Collin, J. Sievert et al., "Commercial Packing and Storage of Navel Oranges Alters Aroma Volatiles and Reduces Flavor Quality," *Postharvest Biology and Technology*, vol. 47, no. 2, pp. 159-167, Feb. 2008.
[10] E. K. H. Katz, H. Y. Boo, and R. A. Kim, "Label-Free Shotgum Proteomics and Metabolite Analysis Reveal a Significant Metabolic Shift During Citrus Fruit Development," *Journal of Experimental Botany*, vol. 62, pp. 5367-5384, Nov. 2011.
[11] D. E. Kachanovsky, S. Filler, T. Isaacson et al., "Epistasis in Tomato Color Mutations Involves Regulation of Phytoene Synthase 1 Expression by Cis-Carotenoids," *PNAS*, vol. 109, No. 456, pp. 19021-19026, Oct. 2012.
[12] J. Yang, Z. Guo, and N. Télef, "Accumulation of Carotenoids and Expression of Carotenoid Biosynthetic Genes During Maturation in Citrus Fruit," *Journal of Agricultural and Food Chemistry*, vol. 63, no. 2, pp. 2631-2648, Feb. 2008.
[13] L. Zhang, G. Ma, M. Kato et al., "Regulation of Carotenoid Accumulation and the Expression of Carotenoid Metabolic Genes in Citrus Juice Sacs in Vitro," *Journal of Experimental Botany*, vol. 63, no. 2, pp. 871-886, Jan. 2012.

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 his M.Sc. (Postharvest Technology) international program from King Mongkut's University of Technology Thonburi (KMUTT) in Bangkok, Thailand. During her Master's, she was a research exchange student at a laboratory of the Center for Environment, Health and Field Science of Chiba University, Japan, which was supported by the Japan Student Services Organization (JASSO) Scholarship. Her research topic was "Chlorophyll Degradation Mechanism and Its Control in Lime Fruit". She received her Ph.D. in Biotechnology from Thaksin University, Thailand in 2020. During their Ph.D. course, she was 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.

Samak Kaewsuksaeng is the Vice President for Research and Innovation of Thaksin University, Thailand. He is an expert in post-harvest technology. He was a visiting scholar at the Faculty of Agriculture, Yamaguchi University, Japan, and many esteemed universities in Korea, and the USA. Furthermore, he was granted for training course on "The Postharvest Physiology, Pathology and Handling of Fresh Commodities" Agricultural Research Organization (ARO), Israel. He has 14 years of teaching experiences and has published numerous research papers in national and international publications. His research was "Postharvest Heat Treatment Delays Chlorophyll Degradation and Maintain Quality in Thai Lime (*Citrus aurantifolia Swingle* cv. Paan) fruit".

Chairat Burana is a lecturer and researcher in the Faculty of Innovative Agricultural Management, Panyapiwat Institute of Management. He was born on 5th June 1983 in Sisaket Province. He received a B.Sc. (Agricultural Science) from Mahidol University, Thailand. He obtained his M.Sc. (Postharvest Technology) international program from King Mongkut's University of Technology Thonburi (KMUTT) Bangkok, Thailand. During his Master's, He was a research exchange student at a laboratory of horticulture, the Faculty of Agriculture, Utsunomiya University, which was supported by the Japan Student Services Organization (JASSO) Scholarship. His research topic was "Effects of 1-methylcyclopropene (1-MCP), Modified Atmosphere Packaging (MAP) and Intermittent Temperatures on the Display Quality and Display Life of Potted Carnation". In 2013, he presented his work at the national and international conferences. Additionally, he received a research award called the "Dr. Adel A. Kader Award for Young Scientists" at the XI Controlled & Modified Atmosphere Research International Conference in 2013, held in Trani, Italy. He received his Ph.D. in Bio-production Science from the Tokyo University of Agriculture and Technology, Tokyo, Japan in 2014. During their Ph.D. course, he was supported by the Japanese Government (MEXT) Scholarship and the Tsuji Asia Foundation Scholarship. His research interests are Postharvest Technology: Maintain the Qualities Deterioration and Handling Systems for Agricultural Products.

Kornlawat Tantivit was born in Kathu, Phuket, Thailand in1989. He received the B.S. degrees in plant science (Horticulture) from the University of Chiang Mai, Thailand in 2011. During B.Sc., he has a Maejo internship at a Horticultural Research Farm, University of Kentucky, KY., USA. He obtained an M.Sc. degree in agriculture from Kagawa University, Miki, Kagawa, Japan in 2014 and a Ph.D. degree in Plant Cytogenetics from Ehime University, Matsuyama, Ehime, Japan in 2017 under the support of a Japanese Government (MEXT) Scholarship. After graduation, he was a postdoctoral researcher at the Biochemistry Laboratory, at Chulalongkorn University. From 2019-2022, he was an agricultural innovation senior officer and specialist of Doikham Food Products Co. Ltd. Since 2023, He is a lecturer and researcher in the Faculty of Innovative Agricultural Management, Panyapiwat Institute of Management.

His research interests are plant genetics, physiology, and soilless culture. He is the 1st author of three articles in the field of plant cytogenetics. His published work in The Journal Cytologia was selected to receive the 2017 Wada Memorial Award of The Japan Mendel Society.

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' 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. Rererences

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.htp

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 Archi-tecture: M anaging 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 pas- sively 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, Pleases 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