

CONTENTS

PREFACE	vi
CONTENTS	viii
LIST OF TABLES	xii
LIST OF FIGURES	xiv
INTISARI	1
ABSTRACT	2
CHAPTER I INTRODUCTION	3
1.1 Research Background.....	3
1.2 Research Problem.....	5
1.3 Research Limitations.....	5
1.4 Research Objectives	6
1.5 Research Benefits.....	6
CHAPTER II LITERATURE REVIEW	7
2.1 Overview and Application of Convolutional Neural Network for Plant Leaf Disease Detection	7
2.2 Chili Plant Disease Detection Using Convolutional Neural Network Approach.....	10
2.3 Chili Plant Disease Detection Using ResNet-18 Architecture on a Convolutional Neural Network Approach	12
2.4 AI-Powered IoT Algorithm using ESP32-CAM for Monitoring.....	13
CHAPTER III RESEARCH METHODOLOGY	16
3.1 Research Description	16
3.2 Hardware Design and System Setup	17

3.2.1	ESP32-CAM Model System	19
3.3	Dataset Description	21
3.3.1	Dataset Acquisition	22
3.4	Model Classification Development.....	24
3.4.1	Data Preprocessing.....	24
3.4.2	Data Splitting	25
3.4.3	Data Augmentation	25
3.4.4	Model Architecture	26
3.4.5	Hyperparameter Settings and Optimization Strategy.....	28
3.4.6	Model Training.....	30
3.4.7	Model Testing	31
3.4.8	Model Scenarios.....	31
3.4.9	Model Fine-Tuning	32
3.5	System Integration	33
3.6	Real Time Detection via Telegram Bot API.....	34
3.7	Evaluation Metrics and Procedure	34
CHAPTER IV IMPLEMENTATION		37
4.1	Hardware Implementation.....	37
4.1.1	ESP32-CAM Configuration Implementation.....	38
4.1.2	Power Supply Implementation.....	39
4.1.3	ESP32-CAM Box Enclosure Implementation	39
4.1.4	ESP32-CAM Mounting Implementation	40
4.1.5	Black Box Hardware Assembly Implementation.....	41
4.2	Implementation Tools	42
4.3	Library Preparation	43

4.4	Data Pre-processing Implementation	48
4.4.1	Pre-processing Configuration.....	48
4.4.2	Pre-processing Execution.....	49
4.4.3	Data Splitting Implementation	50
4.4.4	Data Augmentation Implementation	51
4.5	Model Development Implementation	52
4.5.1	ResNet-18 Custom Components	53
4.5.2	ResNet-18 Model Implementation.....	54
4.6	Training and Optimization	55
4.6.1	Model Training Implementation	55
4.6.2	Fine-Tuning Implementation	56
4.7	Model Conversion and Local Evaluation Implementation	58
4.7.1	TFLite INT8 Conversion.....	58
4.7.2	Local TFLite Testing of Baseline Models	60
4.7.3	Local Evaluation of Fine-Tuned TFLite Model.....	61
4.8	System Integration and Deployment.....	62
4.8.1	Communication Architecture Overview	63
4.8.2	MQTT Topic Specification	64
4.8.3	ESP32-CAM Publisher Implementation.....	64
4.8.4	Python Subscriber Implementation	66
4.8.5	Telegram Integrated Command Flow	68
4.8.6	Runtime Scheduling and Model Switching Logic	70
4.8.7	Error Handling and Edge-Case Behavior.....	72
4.9	End-to-End System Execution	76
CHAPTER V RESULTS AND DISCUSSION.....		78

5.1	Pre-Processing Results	78
5.2	Model Training and Evaluation Results.....	80
5.2.1	Performance Evaluation Across 72 Model Configuration.....	80
5.2.2	Detailed Results of the Two Best Baseline Models.....	86
5.2.3	Baseline Offline Evaluation of the Two Best Models	89
5.2.4	Fine-Tuning Outcomes.....	91
5.2.6	Final Offline Performance of the Fine-Tuned Model	94
5.3	System Integration Result	95
5.3.1	Real-World ESP32-CAM Evaluation on Unseen Leaf Samples ..	95
5.3.2	Qualitative System Output Result.....	107
5.4	Discussion	110
5.4.1	Comparative Analysis with Related Works	110
5.4.2	Baseline Model Performance and Hyperparameter Insights.....	112
5.4.1	Baseline Model Limitations Under ESP32-CAM Conditions	113
5.4.2	Fine-Tuning as Domain Adaptation Strategy	113
5.4.3	Improved Performance After Fine-Tuning	116
5.4.4	Real-World Deployment Evaluation.....	116
5.4.5	Implications and Reflections.....	119
CHAPTER VI CONCLUSION AND RECOMMENDATION		121
6.1	Conclusion	121
6.2	Recommendation.....	122
REFERENCES.....		123