

TABLE OF CONTENTS

ABSTRACT	i
INTISARI.....	ii
TABLE OF CONTENTS.....	iii
LIST OF FIGURES	v
LIST OF TABLES	vi
LIST OF ABBREVIATIONS.....	vii
CHAPTER I INTRODUCTION.....	1
1.1 Research Background.....	1
1.2 Research Problem.....	4
1.3 Research Scope.....	4
1.4 Research Objectives	4
1.5 Research Benefit.....	5
CHAPTER II LITERATURE REVIEW	6
CHAPTER III THEORETICAL BASIS	12
3.1 Convolutional Neural Network (CNN)	12
3.1.1 Matrix Multiplication	13
3.1.2 Convolutional Neural Network Layers	14
3.1.3 MobileNetV1 Model	20
3.1.4 ResNet50 Model	20
3.2 Quantization	21
3.2.1 Quantization Scheme	22
3.2.2 Asymmetric and Symmetric Quantization	23
3.2.3 Quantization-Aware Training	23
3.2.4 Quantized Inference	24
3.3 Faults, Errors, and Fault Injection	26
3.4 Correctness Metrics	28
CHAPTER IV ANALYSIS AND DESIGN.....	30
4.1 Research Description.....	30
4.2 Models to be Quantized.....	31
4.2.1 MobileNetV1	31
4.2.2 ResNet50.....	32
4.3 Dataset	33
4.3.1 Dataset Description	33
4.3.2 Dataset Pre-processing	34
4.4 Model Quantization and Training with Quantization-Aware Training	34
4.5 Interpreter for Quantized Inference	36
4.6 Fault Injection Mechanism	37
4.7 Evaluation.....	38
CHAPTER V IMPLEMENTATION	43
5.1 Software Specification	43
5.2 Implementation of Quantized Models	43
5.2.1 Quantized MobileNetV1 Architecture	43
5.2.2 Quantized ResNet50 Architecture.....	46
5.2.3 Exporting the Models.....	48
5.3 Implementation of QAT	49
5.3.1 Pre-processing Phase.....	49
5.3.2 Training Setup.....	49
5.3.3 Training Loop	50

5.4	Implementation of Fault Injection Tool	51
5.4.1	Quantized Interpreter Implementation	51
5.4.2	Fault Injection Mechanism Implementation	52
5.5	Evaluation Tools.....	53
CHAPTER VI RESULTS AND DISCUSSIONS		54
6.1	Training with Quantization-Aware Training.....	54
6.2	Graph Representation of Quantized Models	55
6.3	Fault Injection Experiments	57
6.3.1	Number of Fault Injections to Perform	57
6.3.2	Fault Injection Results on MobileNetV1 and ResNet50.....	58
6.3.3	Vulnerable Bit Positions	60
6.3.4	Effects of Layer Positioning	61
CHAPTER VII CONCLUSION AND SUGGESTION		63
7.1	Conclusion.....	63
7.2	Suggestion	63
REFERENCES.....		64

LIST OF FIGURES

Figure 3.1: A simple CNN and its main layers	12
Figure 3.2: Visualization of a multi-dimensional matrix with three dimensions	13
Figure 3.3: Illustration of feature maps resulting from each kernel.....	14
Figure 3.4: Visualization of the results of Max and Average Pooling.....	17
Figure 3.5: Illustration of Fully Connected Layers.....	18
Figure 3.6: Transformation of a vector into probabilities by the softmax layer...	19
Figure 3.7: Architecture of MobileNet model.....	20
Figure 3.8: Visualization of the residual block.....	21
Figure 3.9: Distribution represented by symmetric and asymmetric quantization	23
Figure 3.10: Training with fake quantization.....	24
Figure 3.11: Visualization of a bit-flip fault and the resulting propagating error	25
Figure 3.12: Inference with simulated quantization (Gholami et al., 2022).....	25
Figure 3.13: Visualization of a bit-flip fault and the resulting propagating error (Y. He et al., 2020)	27
Figure 4.1: Structure of the research	30
Figure 4.2: QAT process visualized.....	34
Figure 4.3: Visualization of the quantized interpreter performing inference....	36
Figure 4.4: Fault injection in the weight tensor, affecting the input of the subsequent operation in the model	37
Figure 4.5: Flowchart depicting comprehensive fault injection across all layers and tensors	39
Figure 4.6: Visualization of single fault injection.....	40
Figure 4.7: (a) Golden output, (b) Different top-1 label, (c) Same top-1 label, fault-ridden output.....	40
Figure 5.1: Quantized MobileNetV1 Implementation	45
Figure 5.2: Quantized Convolutional Block Implementation	45
Figure 5.3: Quantized Depthwise and Pointwise Convolutional Layers.....	46
Figure 5.4: Quantized ResNet50 Implementation	47
Figure 5.5: Quantized Residual Connection	48
Figure 5.6: Implementaion for Exporting the Model.....	48
Figure 5.7: Pre-processing for QAT	49
Figure 5.8: Training Setup	50
Figure 5.9: Training Loop	50
Figure 5.10: Inference Implementation.....	51
Figure 5.11: Execution for Individual Graph Operations	52
Figure 5.12: Fault Injection Mechanism	52
Figure 5.13: Fault Injection Operations.....	53
Figure 5.14: Validation Tools	53
Figure 6.1: Quantized Depthwise and Pointwise Convolutions in MobileNetV1	55
Figure 6.2: Quantized Residual Block in ResNet50	56
Figure 6.3: Number of Mismatches per bp	60
Figure 6.4: Mismatches Based on Layer Regions.....	61
Figure 6.5: Granular View of l Positioning.....	62

LIST OF TABLES

Table 2.1: Quantization and Bit-Flip Literature Review	10
Table 2.2: Fault Injection Literature Review	11
Table 4.1: MobileNetV1 Architecture	32
Table 4.2: Architecture of Resnet50	33
Table 4.3: Hyperparameters for Training	35
Table 6.1: Final Results of Quantization-Aware Training (160 epochs) and Baselines	54
Table 6.2: Number of Fault Injections with respect to 95% Confidence Interval, 0.1% Margin of Error	58
Table 6.3: Results of Fault Injection Experiments	58
Table 6.4: Results of Various Datatypes	59

LIST OF ABBREVIATIONS

CNN	: Convolutional Neural Network
DNN	: Deep Neural Network
DLA	: Deep Learning Accelerator
QAT	: Quantization-Aware Training