Universiti Teknologi MARA

Sign Language Recognition using You Only Look Once-Neural Architecture Search

MUHAMMAD IMRAN BIN NASHARUDIN
2021101813

Thesis submitted is fulfillment of the requirements for Bachelor of Computer Science (Hons.)
College of Computing, Informatics and Mathematics

August 2023

ACKNOWLEDGEMENT

Alhamdulillah praises and thanks to Allah because of His Almighty and His utmost blessings, I was able to finish this research within the time duration given. Firstly, my special thanks go to my supervisor, Dr. Raseeda Hamzah for giving me supports and supervised me throughout completing this research to make sure that I delivered an excellent output.

I would also like to give a special thanks to my lecturer, Dr. Raihah Binti Aminuddin for guiding me through two semesters in completing this research and to my examiner, Dr. Ahmad Firdaus Bin Ahmad Fadzil for giving me amounts of useful comments in order for me to correct and strengthen my research. Special appreciation and a thousand thanks also go to my beloved parents and siblings for constantly praying for me, giving me support and encourage me to endure all the stress so that I can finish this research successfully and happily.

A lot of thank also addressed to UITM Kampus Jasin, Melaka, for giving a chance on doing this project to gain knowledge and acknowledgement. Last but not least, I would like to thank all my friends for being understanding and supportive throughout the semester.

ABSTRACT

The American Sign Language (ASL) is a nonverbal communication language that uses visual sign patterns formed with the hands or any part of the body and is usually utilized by persons who have hearing or listening disabilities. The deaf and mute people have difficulty communicating their thoughts, needs, and feelings through spoken language. There is a need to have an alternative method based on computer technology for those who are deafen or hard of hearing people since vocal communication is not available to them. One of the issues of computer-based sign language recognition is latency which creates delay in executing the interpretation of certain gestures. To balance latency vs. throughput, the architecture is discovered automatically using a Neural Architecture Search (NAS) technology called AutoNAC. The innovative features of YOLO-NAS include the quantization aware modules QSP and QCI, which combine re-parameterization for 8-bit quantization to minimize accuracy loss during post-training quantization. The architecture is designed to identify tiny objects, increase localization accuracy, and improve the performance-per-compute ratio, making it appropriate for real-time edge-device applications. As using YOLO-NAS for the sign language recognition, we succeeded to deploy average of detection percentage of 86% of all sign language alphabets. YOLO-NAS networks are successfully used in sign language recognition, with a reported 96.41 (mAP@50).

TABLE OF CONTENT

CONTENTS **PAGE**

TABLE OF CONTENT v

LIST OF FIGURES viii

LIST OF TABLES ix

CHAPTER ONE: INTRODUCTION 1

1.1 Background of Study 1

1.2 Problem Statement 3

1.3 Project Objectives 5

1.4 Project Scope 5

1.5 Project Significance 6

1.6 Summary of Project 6

CHAPTER TWO: LITERATURE REVIEW 7

2.1 Overview of Sign Language 8

2.1.1 Language of Sign Language 8

2.1.2 Hearing and Listening Disabilities 9

2.2 Computer Vision 10

2.2.1 Object Detection Deep Learning 11

2.3 Related Work 12

2.3.1 Real-time Vernacular Sign Language Recognition using MediaPipe and Machine Learning 12

2.3.2 Arabic Sign Language Recognition using Faster R-CNN 14

2.3.3 Using YOLOv5 Algorithm to Detect and Recognize American Sign Language 16

2.4 Algorithm used in Object Detection 18

2.4.1 You Only Look Once version 1 (YOLOv1) 18

2.4.2 You Only Look Once version 4 (YOLOv4) 19

2.4.3 You Only Look Once version 5 (YOLOv5) 20

2.4.4 You Only Look Once version 8 (YOLOv8) 21

2.4.5 You Only Look Once – Neural Architecture Search (YOLO-NAS) 22

2.4.6 Summary of Selected Algorithm for Proposed System 24

2.5 System Platform Development 25

2.5.1 Web Based Application 25
2.6 Summary of Chapter 27
CHAPTER THREE: METHODOLOGY 28
3.1 Software Development Methodology 28
3.2 Project Framework 29
3.3 Phase 1: Requirements Analysis 30
3.3.1 Information Gathering 30
3.3.2 Data Collection 30
3.3.3 Data Pre-processing 31
3.4 Phase 2: System Design 32
3.4.1 System Architecture 32
3.4.2 System Flowchart 35
3.4.3 Use Case Diagram 37
3.4.4 User Interface Design 39
3.5 Phase 3: Implementation 41
3.5.1 Hardware and Software Requirements 42
3.5.2 YOLO-NAS development for ASLR 43
3.5.3 Installing the Packages 45
3.5.4 Importing all the Required Libraries 46
3.5.5 Exporting Roboflow Dataset into Google Colab Notebook 48
3.5.6 Load Dataset Parameters into Dictionary 49
3.5.7 Pass the values for '*dataset_params'* 50
3.5.8 Instantiating the model 52
3.5.9 Define Metrics and Training Parameters 53
3.5.10 Training the model 56
3.5.11 Evaluation method 58
3.6 Phase 4: Testing 60
3.6.1 Test Case 61
3.6.2 Black Box Testing 61
3.7 Project Timeline 62
3.8 Summary of Chapter 63
CHAPTER FOUR: RESULT AND DISCUSSION 64
4.1 Analysis Requirements of the System 64
4.1.1 Data Requirements 64
4.2 System Design 68