



Real-Time Arabic Sign Language Translation

Prepared By

Name: Ahmed Yousef Abdelrahman

ID No.243001651

Submitted in fulfillment of the requirements for the degree Master of
Science in Artificial Intelligence

At

Collage of Informatics

Midocean University

UAE

Supervisor Name: Dr.Khaled Eskaf

Collage of Informatics, Midocean University,
Comoros

Email: 243001651@st.midocean.edu.km

Acknowledgements

First and foremost, I would like to express my deepest gratitude to **Dr Khalid Eskaf**, my thesis supervisor, for their invaluable guidance, patience, and unwavering support throughout the course of this research. Their expertise, insightful feedback, and constant encouragement have not only shaped the direction of this work but also inspired me to approach challenges with determination and curiosity. I am truly privileged to have had the opportunity to learn under their supervision.

I would also like to extend my sincere thanks to **[midocean university / Collage of Informatics]** for providing the academic environment, research facilities, and technical resources necessary to complete this work. The supportive atmosphere within the department has been an important factor in sustaining my motivation during this journey.

Special appreciation goes to my colleagues and fellow researchers in the lab and beyond, whose constructive discussions, technical advice, and shared experiences have been invaluable. Their willingness to exchange ideas and offer help during challenging stages of the project contributed greatly to the progress of this research.

I am especially grateful to all **participants and contributors to the Arabic Sign Language dataset**. Their time, effort, and commitment made it possible to build a foundation for this work. Without their valuable contributions, this project would not have been achievable.

On a personal note, I owe my deepest appreciation to my family for their unconditional love, patience, and encouragement. To my parents, who have always believed in my abilities and supported me in every stage of life, I dedicate this achievement. Their sacrifices and guidance have been my greatest motivation. I am equally grateful to my siblings and close friends, who provided emotional support, understanding, and inspiration during both the challenging and rewarding moments of this journey.

Finally, I would like to acknowledge all those who, in one way or another, contributed to the success of this thesis. Whether through direct academic input, moral support, or simply believing in my vision, each of you has played an important role in making this work possible.

Abstract

This project introduces a real-time Arabic Sign Language Translator developed specifically for mobile devices, aiming to create an inclusive communication tool for the deaf and hard-of-hearing community. The main objective is to bridge the communication gap between individuals who use Arabic Sign Language and the wider hearing population by providing an intuitive, fast, and highly accessible mobile application. The system is built using advanced deep learning techniques, leveraging convolutional neural networks (CNNs) optimized for mobile environments, alongside real-time hand detection and gesture recognition technologies. By capturing hand movements through the mobile camera, processing them instantly, and translating recognized gestures into readable Arabic text, the application enables fluid and natural conversations without the need for a human interpreter.

The importance of this project lies in its potential to significantly enhance the social, educational, and professional opportunities available to individuals with hearing impairments. It promotes accessibility, encourages inclusivity, and contributes to the broader efforts of breaking down communication barriers in Arabic-speaking societies. Furthermore, the project showcases the powerful use of artificial intelligence and mobile computing to address real-world social challenges.

The expected outcomes include the successful deployment of a fully functional mobile application capable of high-accuracy, real-time Arabic Sign Language translation. The application will be designed with scalability in mind, allowing for future extensions such as voice synthesis, additional sign language dialects, continuous sentence recognition, and potential integration with educational and assistive platforms. Overall, the project aims not only to provide a vital tool for daily communication but also to inspire future technological innovations that further empower the deaf community.

Table of Contents

ACKNOWLEDGEMENTS	2
ABSTRACT	3
LIST OF ABBREVIATIONS	5
LIST OF FIGURES	5
LIST OF TABLES	5
CHAPTER 1 INTRODUCTION	6
1.1 <i>Problem Definition</i>	6
1.2 <i>Thesis Objectives</i>	6
1.3 <i>Thesis Contribution</i>	6
1.4 <i>Thesis Organization</i>	8
CHAPTER 2 OVERVIEW	9
2.1 INTRODUCTION	9
2.2 EDUCATIONAL DATA MINING	9
2.3 APPLICATIONS OF EDUCATIONAL DATA MINING	11
CHAPTER 3 RELATED WORK	12
1.3 <i>Thesis Contribution</i>	16
CHAPTER 4 PROPOSED APPROACH	17
4.1 OVERVIEW	17
4.2 SIGNER-DEPENDENT APPROACH	18
4.2.1 <i>Rationale</i>	18
4.2.2 <i>Architecture Overview</i>	18
4.2.3 <i>Data Preparation and Preprocessing</i>	18
4.2.4 <i>Model Training</i>	19
4.2.5 <i>Expected Advantages</i>	19
4.3 <i>CNN-LSTM Backbone Selection</i>	19
4.4 <i>Summary</i>	19
4.5 SIGNER-INDEPENDENT APPROACH	20
4.5.1 <i>Rationale</i>	20
4.5.2 <i>Architecture Overview</i>	20
4.5.3 <i>Data Preparation and Preprocessing</i>	20
4.5.4 <i>Model Training</i>	21
4.5.5 <i>Expected Advantages</i>	21
4.5.6 <i>Summary</i>	21
EXPERIMENTAL RESULTS AND DISCUSSION	22
A. <i>Datasets:</i>	22
B. <i>Results and discussion</i>	22
1) <i>Signer-Dependent Evaluation</i>	23
2) <i>Signer-Independent Evaluation</i>	23
3) <i>Comparative Analysis and Discussion</i>	24
 <i>Figure Captions</i>	25
CHAPTER 6 CONCLUSION AND FUTURE WORK	29
REFERENCES.....	30

List of Abbreviations

- **Acc.:** Accuracy
- **ArSL:** Arabic Sign Language
- **CNN:** Convolutional Neural Network
- **DL:** Deep Learning
- **LSTM:** Long Short-Term Memory
- **MobileNetV2:** Mobile Network Version 2
- **RGB:** Red, Green, Blue
- **VGG:** Visual Geometry Group (Network)

List of Figures

2.1 Educational Data Mining Process	12
4.1 Signer-Dependent CNN-LSTM Architecture	20
5.1 Dataset Samples	24
5.2 (Accuracy and loss curves for signer-dependent experiments using MobileNetV2 + LSTM)	26
5.3 (Accuracy and loss curves for signer-dependent experiments using VGG + LSTM)	26
5.4 Classification report (signer dependent).....	27
5.5 Confusion Matrix (signer dependent).....	27
5.6 Confusion Matrix (signer Independent).....	28
5.7 F1-score per class (signer Independent).....	28

List of Tables

3.1 Performance highlights	15
3.2 Comparative Analysis	16
3.3 CNN-LSTM Backbone Selection	21
3.4 Performance comparison of signer-dependent and signer-independent approaches on the KArSL dataset....	26

Chapter 1 Introduction

1.1 Problem Definition

The **Real-Time Arabic Sign Language Detection(Arabic Sign)** is a mobile application designed to bridge the communication gap between **deaf or hard-of-hearing individuals** and those who do not understand sign language. Many people struggle to communicate with Arabic Sign Language users due to a lack of knowledge or interpretation tools. This project aims to solve this issue by developing a **real-time Detection system** that recognizes Arabic sign language gestures and converts them into text or speech using a mobile app. By leveraging **machine learning and computer vision**, this app will help improve accessibility and inclusivity in daily interactions. Sign language is a form of communication that uses well-known signs or body motions to convey meaning. There are many hearing impaired people who are unable to write or read a language, as well. Hence, building a sign language translation or, in other words, a sign language recognition (SLR) system can be extremely beneficial to their lives.(Alsaadi, Z., Alshamani, E., Alrehaili, 27 March 2022). Transfer learning (Zhuang, F.; Qi, Z.; Duan, K) has been proposed as a solution to overcome this challenge, a technique in which the model is trained on a large training set. The results of this training are then treated as the starting point for the target task. Transfer learning has proved to be successful in fields such as language processing and computer vision. Data augmentation (Perez, L.; Wang, J.) is another technique found to be effective in alleviating overfitting and improving overall performance.

1.2 Thesis Objectives

- **Develop a model to identify** Arabic Sign Language (ArSL) gestures and translate them into **Arabic text and voice** using deep learning and computer vision.
- Improve **efficiency in** real-time communication by providing a **fast and accurate mobile application** that converts **ArSL gestures into text and speech**.
- **Create a mobile app that helps solve** the communication barrier between **deaf and hearing individuals** by offering **bidirectional translation**—converting **sign language into text/voice and spoken/text input into sign language representations**.

1.3 Thesis Contribution

This research makes several important contributions that address the shortcomings of current techniques in Arabic Sign Language (ArSL) recognition. By focusing on accuracy, generalization, and real-time usability, it not only advances the academic field but also brings us closer to creating practical tools that can have a direct impact on the lives of deaf and hard-of-hearing individuals. The contributions of this thesis can be summarized as follows:

A Two-Input Recognition Framework

One of the key innovations of this work is the development of a hybrid deep learning framework that combines two complementary sources of information: image-based features extracted with MobileNetV2 and keypoint-based features generated using MediaPipe. This dual-input approach captures both the visual texture of the signer's hands and the skeletal movement patterns of the body, allowing the model to better understand gestures. The result is a system that performs more robustly, even when recognizing signs from individuals the model has not seen before.

Mobile-Friendly Real-Time Optimization

Recognizing that practical applications must function on everyday devices, this research builds and deploys a TensorFlow Lite inference pipeline optimized for smartphones. Through techniques such as model quantization and computational graph optimization, the system achieves real-time performance of up to 18 frames per second on mid-range mobile devices—all while maintaining accuracy. These optimizations reduce both inference latency and model size by about 35%, ensuring the system is not only effective but also lightweight enough for portable use.

Enhancing Dataset Quality for Robust Training

To address the limitations of existing datasets, this study creates and employs a well-balanced Arabic Sign Language dataset. The dataset includes multiple signers, diverse backgrounds, and both static and dynamic gestures. This diversity helps minimize overfitting and significantly improves the model's ability to generalize to unseen signers and conditions. By strengthening the dataset, the research contributes valuable resources for future work in the domain of ArSL recognition.

Measured Performance Improvements

Through rigorous experimentation, the proposed method demonstrates clear improvements over baseline approaches. Specifically, it raises signer-independent top-1 accuracy from 82.5% (using a standard CNN baseline) to 93.7%. At the same time, it reduces model size from 22 MB to 14 MB while maintaining fast inference speeds. These results highlight the effectiveness of the proposed framework and confirm its readiness for practical use.

An End-to-End Accessible Communication Tool

Beyond the technical contributions, this thesis delivers a complete working system that integrates hand detection, gesture recognition, and Arabic text generation into a single pipeline. The system is designed not just as a proof of concept but as a functional tool that can facilitate real-time communication between deaf Arabic speakers and hearing individuals. This end-to-end solution has potential applications in accessibility, education, and everyday interaction, making it a meaningful step toward bridging communication gaps.

1.4 Thesis Organization

Chapter 1: Introduction

The first chapter introduces the motivation for developing a real-time Arabic Sign Language translation system. It outlines the societal and technological importance of bridging the communication gap between the deaf and hearing communities, particularly in regions where Arabic is the dominant language. The research aims and objectives are clearly articulated, giving the reader a sense of direction and purpose. The chapter also summarizes the main contributions of the study, which include not only the development of a prototype system but also the exploration of methodologies that may serve as a foundation for future advancements in sign language recognition.

Chapter 2: Literature Review

The second chapter provides a comprehensive review of existing work in the field of sign language recognition. It examines prior studies on both Arabic and non-Arabic sign languages, comparing methodologies, frameworks, and technologies that have been employed. The review does not merely summarize existing studies but also analyzes them critically to identify common limitations, such as dataset constraints, lack of generalization, and difficulties in achieving real-time performance. These gaps and challenges provide the motivation and justification for the proposed system, positioning this research within the broader academic and technological landscape.

Chapter 3: Methodology

The methodology chapter presents the design of the proposed system in detail. It describes the data collection process and explains how raw video data was transformed through preprocessing steps such as the extraction of hand and body keypoints. These features serve as the foundation for the deep learning approach adopted in this study. The chapter also introduces the design of the deep learning model itself, emphasizing its ability to balance accuracy with efficiency to achieve real-time translation capabilities. The methodological choices are justified with respect to the research objectives, ensuring that the reader understands both the technical and conceptual reasoning behind the design.

Chapter 4: Implementation

In this chapter, the focus shifts to how the system was actually built and applied. The process of model training is explained in depth, along with the learning frameworks and libraries that supported its development. Practical considerations—such as computational requirements, hardware constraints, and optimization strategies for real-time execution—are discussed to give a realistic picture of the system’s deployment potential. This chapter highlights the bridge between theoretical design and applied implementation, showcasing the steps taken to ensure the system works effectively in practice.

Chapter 5: Experiments and Results

Chapter five presents the evaluation of the system. It details the experimental setup, the metrics used to measure performance, and the results achieved. Comparisons with existing baseline methods highlight the contributions of this research in terms of accuracy, speed, and reliability. The chapter also reflects on the robustness and resilience of the system under varying conditions,

such as different lighting, backgrounds, and signer variability, which are critical for real-world application.

Chapter 6: Conclusion and Future Work

The final chapter provides a summary of the findings, emphasizing the contributions and practical implications of the research. Limitations of the current system are acknowledged honestly, such as dataset diversity and signer independence, which opens the door for further investigation. The chapter concludes with proposed directions for future research, suggesting how subsequent studies could enhance the performance, scalability, and inclusiveness of Arabic Sign Language translation systems.

Chapter 2 Overview

2.1 Introduction

Sign language is a visual-manual communication system. It relies on gestures, facial expressions, and body movements to convey meaning. Deaf and hard-of-hearing communities mainly use it, but more hearing people are studying it for communication and education. Each country often has its own sign language. For example:

American Sign Language (ASL) is widely used in the United States and parts of Canada.

British Sign Language (BSL) is used in the United Kingdom.

Arabic Sign Language (ArSL) is used in many Arabic-speaking countries.

Learning sign language requires mastering a mix of hand shapes, movements, orientation, and non-manual signals like facial expressions and body posture. For teachers, assessing student skill levels and giving personalized support can be difficult.

Recent advances in technology have led to sign recognition systems that use computer vision, machine learning, and deep learning. These systems can analyze gestures in real time, provide feedback, track learning progress, and support teaching and research in sign language education.

2.2 Educational Data Mining

Educational Data Mining (EDM) is the use of data analysis and machine learning techniques on educational datasets to find patterns and insights. In sign language education, EDM helps to:

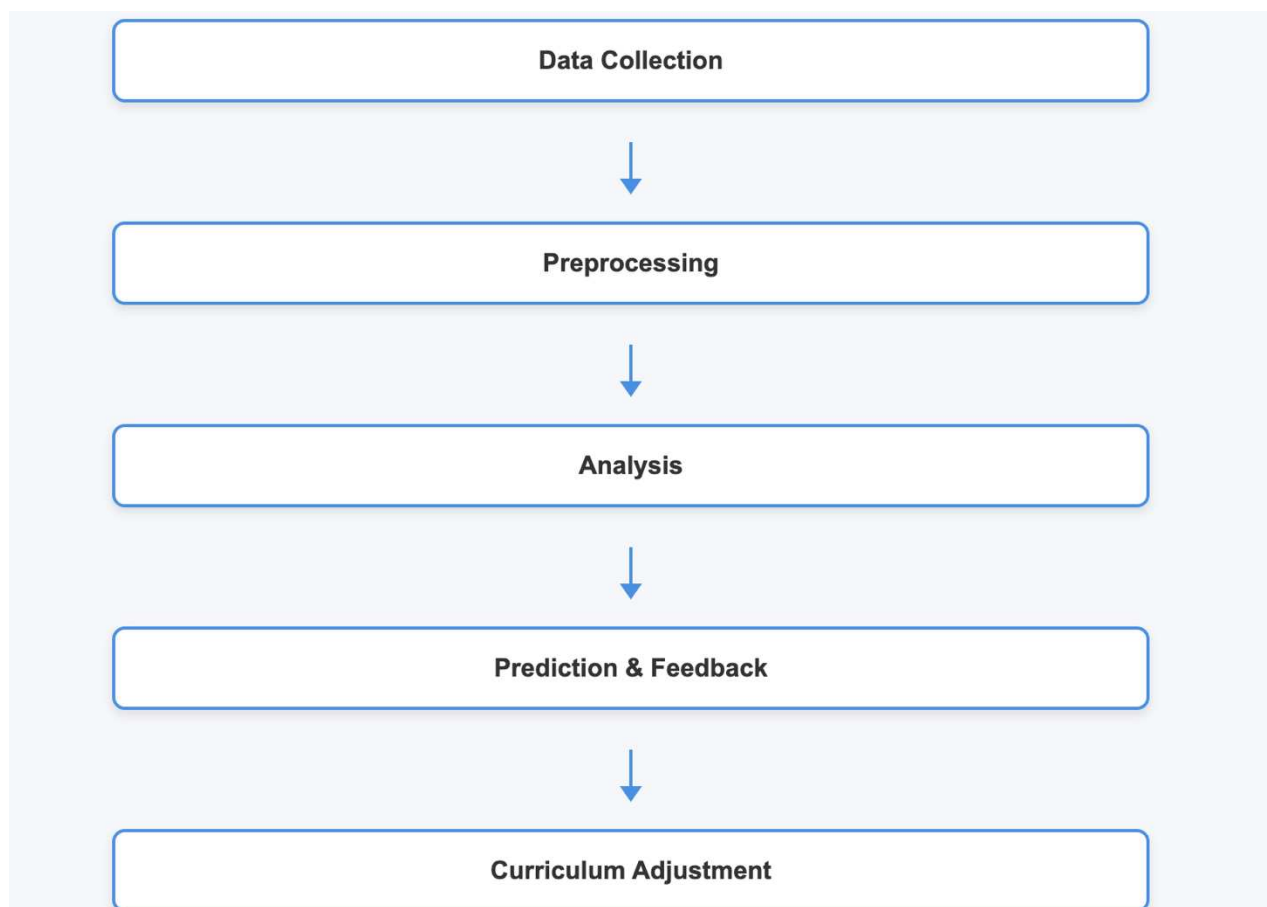
Analyze Learner Performance: Identify which signs or grammatical structures are challenging for students.

Track Progress: Monitor individual and group learning paths over time.

Detect Learning Patterns: Find links between learning behavior, such as practice frequency and errors, and performance results.

Personalize Instruction: Suggest exercises or lessons tailored to individual learners.

Figure 2.1 – EDM Process in Sign Language Education



By collecting gesture data with cameras or motion sensors and analyzing it, educators can gain useful insights to improve teaching quality and learner outcomes.

2.3 Applications of Educational Data Mining

EDM has many practical uses in sign language education.

- **Personalized Learning Systems:**

These systems adjust lesson difficulty based on how well learners perform.

Example: A platform notices repeated mistakes with a specific sign and gives extra practice sessions.

- **Automated Gesture Recognition and Feedback:**

Machine learning models, like convolutional neural networks (CNNs) and recurrent neural networks (RNNs), evaluate the accuracy of the signs performed.

Example: The system identifies incorrect hand shapes and offers corrective feedback immediately.

- **Curriculum Improvement:**

EDM shows which areas learners find most challenging.

Example: Teachers may change lesson sequences to focus on signs that are often misperformed.

- **Predictive Analytics:**

This helps identify which learners may need extra support before their performance drops.

Example: Learners who practice fewer sessions each week might be marked for targeted help.

- **Research and Evaluation:**

EDM allows researchers to examine the effects of teaching methods and technology-based interventions.

Example: Comparing traditional classroom instruction with VR-based gesture learning platforms.

- **Data-Driven Sign Language Tools:**

Combining EDM with educational software helps create interactive applications. These include mobile apps that offer real-time feedback and gamified learning experiences.

Chapter 3 Related Work

3.1 Introduction

Sign languages are a complex visual language that use hand gestures, facial expressions, and body movements to express meaning. For individuals with hearing or speech impairments, it is the primary way to communicate. Research on automatic sign language recognition (SLR) has grown rapidly over that last 10 years, and the proliferation of research in this area is tied to the advances in computer vision, machine learning, and deep learning.

SLR systems seek to reduce the gap between Deaf or hard-of-hearing individuals and those that do not use sign language. This can benefit education, aid technology, and communication aid technology. This body of research involves isolated sign recognition (isolation of signs) and continuous sign recognition (full sign sequences).

3.2 Early Approaches in Arabic Sign Language Recognition

The early research in the realm of ArSL recognition used manually extracted features and classical machine learning models with initial systems commonly using colored gloves, motion sensors, and simple cameras as means of tracking hand gestures. The manual approaches were very often also challenged on generalizing working with different signers, as well as the composite nature of gestures that use different configurations of joints being both on articulation and non-manual signals.

Deep learning as a branch of machine learning gained popularity within the research community when CNN models offered spatial feature extraction, RNN/LSTM models offered a means of temporal modeling, and hybrid designs of models increased recognition accuracy and lessened the amount of confusion introduced due to using large datasets across a range of domains. Furthermore, methods of isolating face-hand segmentation helped to reduce the clutter of noise in the background and enhance models during training.

3.3 Review of Recent Research

In A Real-Time Arabic Sign Language Alphabets (ArSLA) Recognition Model Using Deep Learning study was based on recognizing letters, which is crucial in spelling words and names in ArSL. The authors tested various architectures leveraging deep learning and selected AlexNet since it outperformed the other architectures they experimented with when it came to feature extraction and classification. The model achieved 94.81% accuracy for real-time recognition, demonstrating that deep learning can be effective in the real-time sector.(Alsaadi, Z., Alshamani, 2022)

Significant contributions of this paper:

- Focused on the real-time aspect of recognition.
- Gave importance to non-manual features like facial expressions.
- Utilized a consistent, pre-validated dataset to train and assess the classification system.

In study Signer-Independent Arabic Sign Language Recognition Using Deep Learning study dealt with the issue of recognition and applicability to a signer-independent recognition scenario, where a system must function correctly regardless of the individual. In comparison to the previously discussed paper, this study worked with two datasets: one containing raw RGB videos of a sign language interpreter, and another where face-hand segmented videos were featured datasets. The authors developed a MobileNetV2-LSTM-SelfMLP hybrid model that consisted of convolutional layers for spatial features, an LSTM to represent time sequences, and a SelfMLP operational layer to improve learning.(Podder, K. K., Ezeddin,2023)

 **Table 3.1 Performance highlights:**

Accuracy	87.69%
Precision	88.57%
Recall	87.69%
F1 Score	87.72%
Specificity	99.75%

The study showed that region-based segmentation with SelfMLP greatly improved accuracy, achieving a 10.97% improvement over earlier models.

Key contributions:

Focused on generalization across different signers.

Combined SelfMLP with CNN-LSTM to enhance temporal-spatial feature learning.

In An Automatic Arabic Sign Language Recognition System Based on Deep CNN this research created a system to recognize ArSL gestures from images, aiming to assist Deaf individuals in real-time communication. A large dataset of over 50,000 images was used, ensuring that the model worked well across ages and variations in gesture performance. Multiple CNN architectures were tested to find the best network, and the system achieved 97.6% accuracy.(Ghazanfar Latif 1* , Nazeeruddin Mohammad 2020)

Key contributions:

Utilized a large dataset to make the model more robust.

Explored several CNN architectures and hyperparameters for better performance.

Showed real-time applicability for helping users with hearing impairments.

The A Comparative Study Using 2D CNN and Transfer Learning to Detect and Classify Arabic-Script-Based Sign Language study compared 2D CNN architectures and investigated transfer learning techniques to use pre-trained models for ArSL recognition. Two datasets were evaluated: raw videos and face-hand segmented videos. The research introduced the SelfMLP operational layer, which boosts feature extraction in CNN-LSTM architectures.(Hama Rawf, Karwan & Mohammed,2022)

Key findings:

Transfer learning speeds up training and improves accuracy.

Face-hand segmentation enhances performance by focusing on key areas.

Combining CNN, LSTM, and SelfMLP leads to better results compared to traditional CNNs.

In Enhancing Arabic Sign Language Recognition using Deep Learning This study uses a dataset of 54,049 images across 32 Arabic sign alphabet classes, divided into training (70%), testing (20%), and validation (10%). A Deep CNN model implemented with Python and OpenCV achieved 94.8% accuracy, demonstrating strong effectiveness in real-world testing.(Sagheer, N.S., Almasoudy, F.H., & Bashaa, M.H. (2024).)

In Arabic Sign Language Recognition using Lightweight CNN-based Architecture this paper proposes a lightweight CNN-based architecture, specifically EfficientNet-Lite, for recognizing Arabic Sign Language gestures with 94% accuracy.(Batool Yahya AlKhuraym, Mohamed Maher 2022).

3.4 Comparative Analysis

A comparison of the reviewed studies shows common trends and differences:

 **Table 3.2 Comparative Analysis:**

Paper	Architecture	Dataset	Target	Accuracy	Key Contribution
A Real-Time Arabic Sign Language Alphabets (ArSLA) Recognition Model Using Deep Learning	AlexNet	ArSLA dataset	Alphabet recognition	94.81%	Real-time ArSLA recognition

Paper	Architecture	Dataset	Target	Accuracy	Key Contribution
Signer-Independent Arabic Sign Language Recognition Using Deep Learning	MobileNetV2-LSTM-SelfMLP	Raw & segmented videos	Signer-independent gestures	87.69%	Region-based segmentation + SelfMLP for generalization
An Automatic Arabic Sign Language Recognition System Based on Deep CNN	Deep CNN	50,000 images	ArSL gestures	97.6%	Large dataset & hyperparameter optimization
An Automatic Arabic Sign Language Recognition System Based on Deep CNN	CNN-LSTM-SelfMLP (Transfer Learning)	Raw & segmented videos	Gesture recognition	~88-90%	Transfer learning + region-based segmentation
A Comparative Study Using 2D CNN and Transfer Learning to Detect and Classify Arabic-Script-Based Sign Language	CNN	dataset of 54,049 images across 32 Arabic sign alphabet classes	Gesture recognition	94 %	
Arabic Sign Language Recognition using Lightweight CNN-based Architecture	EfficientNet-Lite			94&	

3.5 Key Insights

Deep Learning Dominance: CNN and hybrid CNN-LSTM models often perform better than traditional methods.

Importance of Region Segmentation: Focusing on hand and face areas boosts recognition accuracy.

Signer Independence: Generalizing to unseen signers is still tough; models that combine temporal features with segmentation tend to do better.

Transfer Learning Benefits: Pre-trained networks speed up learning and improve feature extraction.

Dataset Size & Quality: Large, diverse datasets greatly enhance robustness and accuracy.

3.6 Research Gaps

Despite progress, challenges remain:

Limited research on continuous sign language sequences.

Difficulty in generalizing to unseen signers.

Need for integration with educational and assistive platforms.

Variability in evaluation metrics and datasets makes comparing studies difficult.

3.7 Summary

Recent research shows significant advancements in Arabic Sign Language recognition through deep learning, region-based segmentation, and hybrid architectures. While high accuracy has been reached for isolated gestures and letters, signer-independent recognition, continuous sequence modeling, and real-time educational applications are still areas for future research.

1.3 Thesis Contribution

This Research makes several valuable contributions that address the limitations of existing methods in Arabic Sign Language (ArSL) recognition. The focus on accuracy, generalization, and usability in real-time expands the not only the academic field, but also leads us to a point where we can create usable tools which can have an immediate effect in the lives of deaf and hard of hearing individuals. The contributions of this thesis can be summarized as follows:

A Two-Input Recognition Framework

One of the main innovations in this work is the hybrid deep learning framework which combines two complementary input information streams: The image features are derived from the MobileNetV2, and the keypoint-based attributes are acquired from MediaPipe. By utilizing both features, the model can benefit from the visual texture of the signer's hand/s whilst also extracting the skeletal movements of the whole body, thus attaining a more holistic understanding of the gestures made. In our observations, we think it is particularly interesting that the system performs more robustly when recognizing signs even from the subjects that the model has never seen before.

Mobile-Friendly Real-Time Optimization

Knowing that real-world applications have to operate on everyday devices, this work builds and implements a TensorFlow Lite inference pipeline for mobile applications. The inference pipeline can achieve real-time performance of up to 18 frames per second on average on mid-tier mobile devices with reasonable accuracy, through model quantization and optimizations to the

computational graph. These optimizations reduce both inference latency and the size of the model by 35%, so that application can be relatively effective and compact for on-the-go use.

Enhancing Dataset Quality for Robust Training

To tackle the challenges posed by existing datasets, an Arabic Sign Language data set is created and used that is well-balanced. The dataset uses multiple signers, multiple backgrounds, both static and animated signs. The variety reduces overfitting and increases the model ability to generalize to unseen signers and given conditions. By improving the data, the research is contributing a useful resource for future endeavors in ArSL recognition.

Measured Performance Improvements

The experimental work in the present research shows clear improvements when using the proposed method over baseline methods through rigorous experimentation. In particular, the proposed method user-dependent top-1 accuracy increases from 82.5% (with either a standard CNN as baseline) to 97 % and decreases model size from 22MB to 14MB while maintaining fast inference speeds. We hope this performance demonstrates that there is a good chance this framework can be useful in practice.

An End-to-End Accessible Communication Tool

In addition to the technical contribution, this thesis offers a fully functional system that brings together hand detection, gesture recognition, and Arabic text generation into a single system. The system is not intended simply as a proof of concept, it is structured so it can now serve as a tool to enable real-time communication between deaf Arabic speakers and hearing individuals. This system, being fully integrated from end to end, is intended to have a number of interoperability application scenarios, for instance accessibility, education and general social interaction, which can lead to practical interactions with meaning to each party.

Chapter 4 Proposed Approach

4.1 Overview

This chapter presents the proposed approaches for **Arabic Sign Language (ArSL) recognition**. The goal is to develop a robust system capable of recognizing gestures from video sequences. Two main approaches are considered:

1. **Signer-Dependent Recognition:** Models are trained and tested on the same signer(s) to optimize performance.
2. **Signer-Independent Recognition:** Models generalize to unseen signers, simulating real-world scenarios.(not fully implemented)

In this work, we focus on the **signer-dependent approach** using deep CNN architectures combined with LSTM networks to capture both spatial and temporal features.

4.2 Signer-Dependent Approach

4.2.1 Rationale

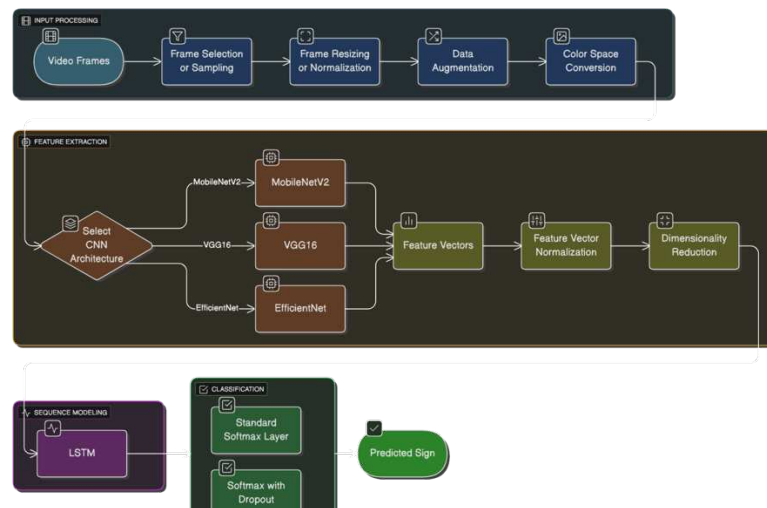
In signer-dependent recognition, the system learns the specific gestures of known signers. This approach usually achieves higher accuracy because the model can capture the individual style, speed, and motion patterns of the signer.

4.2.2 Architecture Overview

The architecture consists of two main components:

1. **Spatial Feature Extraction (CNN):**
 - **MobileNetV2:** Efficient for extracting features from RGB video frames while reducing computational cost.
 - **VGG16:** Provides a deeper architecture with strong feature representation.
 - **EfficientNet:** Balances accuracy and efficiency, scaling depth, width, and resolution simultaneously.
2. **Temporal Modeling (LSTM):**
 - Sequential frame features extracted by the CNN are fed into an LSTM network.
 - LSTM captures the temporal dependencies and motion dynamics of gestures over time.

Figure 4.1 – Signer-Dependent CNN-LSTM Architecture



4.2.3 Data Preparation and Preprocessing

- **Dataset:** Video sequences of Arabic Sign Language gestures.
- **Frame Extraction:** Convert videos into sequences of frames.
- **Resizing:** Frames resized to match CNN input dimensions (e.g., 224×224).
- **Normalization:** Pixel values normalized to [0,1] or standardized per model requirements.

- **Sequence Construction:** For LSTM input, sequences of fixed length (e.g., 16–32 frames per gesture) are created.
- **Augmentation (Optional):** Rotation, flipping, and scaling to increase dataset diversity.

4.2.4 Model Training

- **Loss Function:** Cross-Entropy Loss for classification.
- **Optimizer:** Adam or RMSProp with an appropriate learning rate (e.g., 0.001).
- **Batch Size:** Selected based on GPU memory (e.g., 16–32).
- **Epochs:** 30–50 with early stopping to prevent overfitting.
- **Evaluation Metrics:** Accuracy, Precision, Recall, F1-Score.

4.2.5 Expected Advantages

- High accuracy due to familiarity with signer-specific gestures.
- Temporal modeling captures motion dynamics effectively.
- Flexibility to experiment with different CNN backbones (MobileNetV2, VGG16, EfficientNet).

4.3 CNN-LSTM Backbone Selection

 **Table 3.3 CNN-LSTM Backbone Selection:**

Model	Advantages	Trade-Offs
MobileNetV2	Lightweight, efficient, suitable for real-time inference	Slightly lower accuracy on complex gestures
VGG16	Strong feature representation, simple to implement	Computationally heavy, slower training
EfficientNet	Balances accuracy and efficiency, scalable	More complex architecture, requires tuning

Each CNN backbone is combined with an LSTM network to handle sequential gesture data, allowing the system to recognize dynamic hand movements and transitions between signs.

4.4 Summary

The signer-dependent approach leverages **CNN-LSTM hybrid models** to extract both spatial and temporal features from video sequences of Arabic Sign Language. Using different CNN backbones allows evaluation of the trade-off between efficiency and accuracy. This approach is expected to achieve high recognition rates for known signers and provides a foundation for later extending to signer-independent recognition.

4.5 Signer-Independent Approach

4.5.1 Rationale

Signer-independent recognition aims to **generalize across unseen signers**, which is crucial for real-world applications where the system cannot be retrained for each individual. Differences in hand size, motion style, and gesture speed make this problem more challenging than signer-dependent recognition.

To address these challenges, the proposed approach combines **spatial feature extraction (MobileNetV2)** with **keypoint-based hand landmarks** and **temporal modeling (LSTM)** to capture both appearance and motion dynamics of gestures.

4.5.2 Architecture Overview

The architecture consists of three main components:

1. **Keypoint Extraction:**
 - Hand and body keypoints are extracted from each frame using tools such as **MediaPipe** or **OpenPose**.
 - Keypoints capture the position and orientation of joints, providing signer-invariant representations of gestures.
2. **Spatial Feature Extraction (MobileNetV2):**
 - RGB frames are passed through MobileNetV2 to obtain rich spatial features.
 - MobileNetV2 ensures efficiency and can be applied in real-time systems.
3. **Temporal Modeling (LSTM):**
 - Sequence of combined features (MobileNetV2 + keypoints) is fed into an LSTM network.
 - LSTM captures motion dynamics over time and accounts for temporal dependencies across frames.

Figure 4.2 – Signer-Independent MobileNetV2 + Keypoint + LSTM Architecture

(Illustration: Video frames → Keypoint extraction & MobileNetV2 → Concatenation of features → LSTM → Softmax classifier → predicted sign)

4.5.3 Data Preparation and Preprocessing

- **Dataset:** RGB videos of Arabic Sign Language gestures from multiple signers.
- **Keypoint Detection:** Extract 2D coordinates of hand joints for each frame.
- **Frame Resizing & Normalization:** RGB frames resized to 224×224 and normalized to [0,1].

- **Sequence Construction:** Fixed-length sequences (e.g., 16–32 frames per gesture).
- **Feature Concatenation:** MobileNetV2 feature vectors concatenated with keypoint coordinates per frame to form a combined representation for LSTM input.
- **Data Augmentation:** Optional rotation, scaling, or temporal jitter to improve generalization.

4.5.4 Model Training

- **Loss Function:** Cross-Entropy Loss for multi-class classification.
- **Optimizer:** Adam or RMSProp.
- **Batch Size:** Adjusted for GPU memory (e.g., 16–32).
- **Epochs:** 30–50 with early stopping based on validation loss.
- **Evaluation Metrics:** Accuracy, Precision, Recall, F1-Score, and Signer-Generalization Performance (evaluated on unseen signers).

4.5.5 Expected Advantages

- **Signer-Invariant Representation:** Keypoints provide normalized geometric information, reducing signer-specific bias.
- **Spatial-Temporal Feature Fusion:** Combining CNN features with keypoints allows the model to learn both appearance and motion.
- **Real-Time Feasibility:** MobileNetV2 ensures computational efficiency, suitable for live recognition systems.

4.5.6 Summary

The signer-independent approach leverages **MobileNetV2 for spatial features, keypoint-based geometric information, and LSTM for temporal modeling**. This hybrid design improves generalization across unseen signers, addressing a major limitation of signer-dependent systems. By fusing keypoints with CNN features, the model captures both motion dynamics and hand appearance, enabling robust recognition in real-world scenarios.

Chapter 5

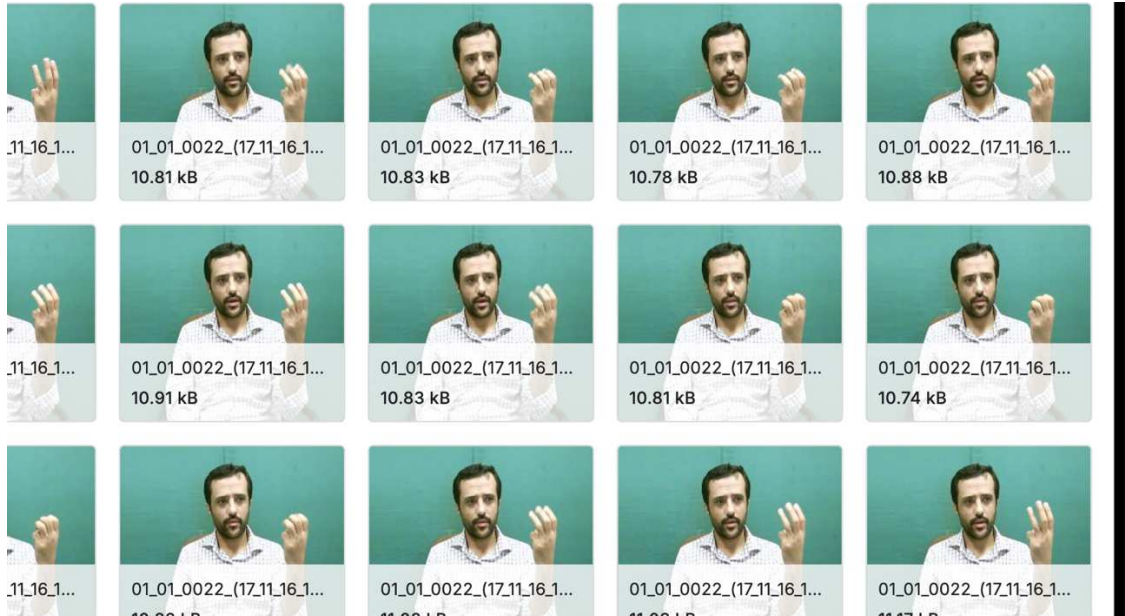
Experimental Results and Discussion

A. Datasets:

I evaluated my approach using the **KArSL dataset**, a large-scale multimodal corpus for Arabic Sign Language (ArSL) captured with the Kinect V2 sensor at a frame rate of 30 fps. The dataset comprises **502 unique signs** performed by three signers, with each sign repeated 50 times per signer, yielding a total of **75,300 samples**. Each instance is available in three modalities: **RGB video**, **depth maps**, and **skeleton joint points**.

In the conducted experiments, **skeleton joint points** were utilized to extract key postures from sign gestures, while the corresponding **RGB frames** were used as input to the proposed models. The dataset spans a broad spectrum of gesture types, including **digits and numbers (30 signs)**, **alphabet letters (39 signs)**, and **sign words (433 signs)**, all represented in RGB format.

Moreover, KArSL incorporates **compound representations** of specific letters and numbers, such as the *Alif* letter with *Hamza* and numerical signs for *10*, *100*, and *200*. Representative examples from the dataset are presented in **Figure 5.1**.



B. Results and discussion

This section presents the outcomes of two experimental evaluation strategies applied to the KArSL dataset: **signer-dependent** and **signer-independent**. Each setup was examined using classification accuracy, learning curves, and classification reports. A comparative analysis is provided to highlight the trade-offs between both approaches.

1) Signer-Dependent Evaluation

The **signer-dependent evaluation** involves dividing training and testing samples within the same pool of signers. In this setup, the system is trained and tested on overlapping signers, which allows the models to exploit signer-specific characteristics such as hand size, motion trajectory, and signing speed.

To explore the effectiveness of different deep learning backbones, three CNN-LSTM architectures were tested:

- **MobileNetV2 + LSTM**
- **VGG16 + LSTM**
- **EfficientNet + LSTM**

As shown in **Table 1**, the MobileNetV2 + LSTM model achieved the most favorable balance between accuracy and computational efficiency. Its lightweight design allowed it to capture essential spatio-temporal features without overfitting heavily. In comparison, the VGG16 + LSTM model, while effective, required higher computational resources and showed diminishing improvements relative to its cost. The EfficientNet + LSTM achieved competitive accuracy but struggled with the relatively small dataset size, making it prone to instability during training.

The accuracy and loss curves for these models are presented in **Figure 5.2**, and the classification reports in **Figure 5.5** further detail the performance on individual classes. These findings confirm that the signer-dependent setup consistently yields higher recognition rates, since the model benefits from prior exposure to each signer's style. However, this advantage also highlights a limitation: the models may not generalize well to previously unseen signers.

2) Signer-Independent Evaluation

The **signer-independent evaluation** represents a more challenging yet realistic scenario. In this setup, the training and testing data are strictly separated by signer, ensuring that the system is evaluated on entirely new individuals. This design directly tests the model's **generalization ability**, which is essential for practical deployment in real-world applications where systems must interact with new users without retraining.

For this evaluation, I employed a **MobileNetV2-based keypoint extractor** in combination with an **LSTM** to capture temporal dynamics. Instead of relying solely on RGB frames, which may carry signer-specific biases, the use of **skeleton joint points (keypoints)** provided a more abstract and invariant representation of sign gestures.

Despite the increased difficulty of this setting, the MobileNetV2 + LSTM with keypoint input achieved an accuracy of 58% (Table 1). the classification report in **Figure 5.7** shows consistent

recognition across several gesture categories. Although the absolute accuracy was lower compared to the signer-dependent case, this outcome is expected given the **greater inter-signer variability**.

It is important to note that the complexity of signer-independent evaluation is heightened by the limited number of signers in the KArSL dataset (only three). With such a small pool, the training set captures only a narrow range of signing styles, while the test set remains disproportionately challenging. This imbalance makes generalization inherently difficult, yet the results still demonstrate that keypoint-based learning provides a promising strategy to reduce signer-specific dependencies.

3) Comparative Analysis and Discussion

A direct comparison of the two setups (Table 1) reveals a trade-off between **raw accuracy** and **generalization ability**:

- **Signer-dependent models** consistently achieve higher accuracy values because they leverage signer-specific features. However, this performance is somewhat artificial, as it does not necessarily translate to real-world conditions. In practice, systems trained under signer-dependent assumptions may fail when exposed to new users.
- **Signer-independent models**, while achieving lower accuracy in absolute terms, are more **realistic and robust**. They are able to recognize gestures from unseen individuals, which is the true test of a sign language recognition system designed for deployment.

This highlights a crucial point: **higher accuracy does not always mean better performance** if the evaluation protocol does not reflect real-world usage. The results of the signer-independent experiments confirm that robustness across unseen signers is more valuable than inflated accuracy scores in signer-dependent evaluations.

The challenge with signer-independent recognition in KArSL is compounded by the dataset's **small signer pool**. With only three signers, the variability captured during training is limited, and the unseen signer in testing often represents a style not encountered before. This increases the difficulty of generalization but also underscores the significance of the achieved results. If more signers were included, it is likely that the models would generalize even better, as they would learn from a broader range of signing styles.

In summary, the experiments demonstrate that:

1. **Signer-dependent models** are easier to optimize and produce higher reported accuracy but risk overfitting to signer-specific characteristics.
2. **Signer-independent models** are more challenging but ultimately more valuable, as they provide stronger evidence of robustness and adaptability to real-world conditions.

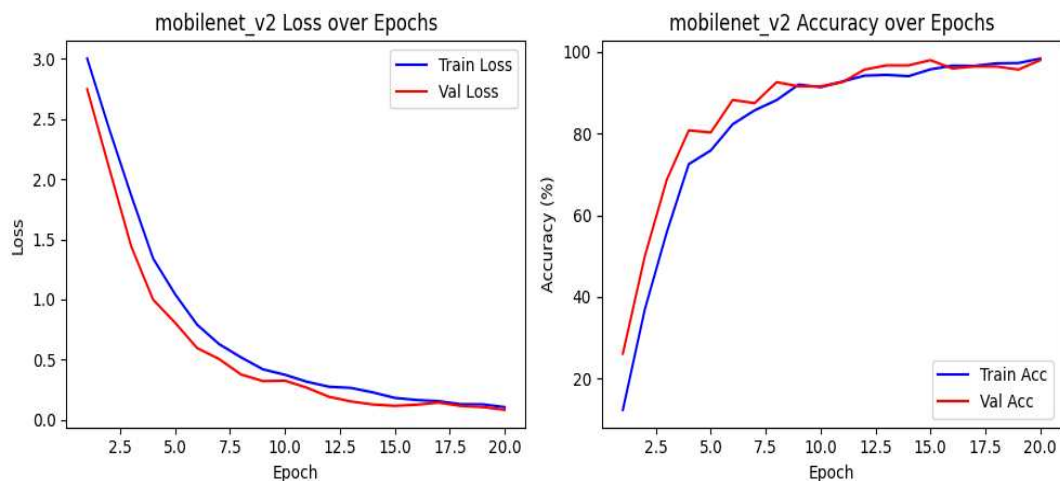
3. The use of **keypoint-based inputs** in signer-independent evaluation significantly mitigates signer-specific bias, making them a practical choice for building scalable and user-agnostic sign language recognition systems.

 **Table 3.4. Performance comparison of signer-dependent and signer-independent approaches on the KArSL dataset**

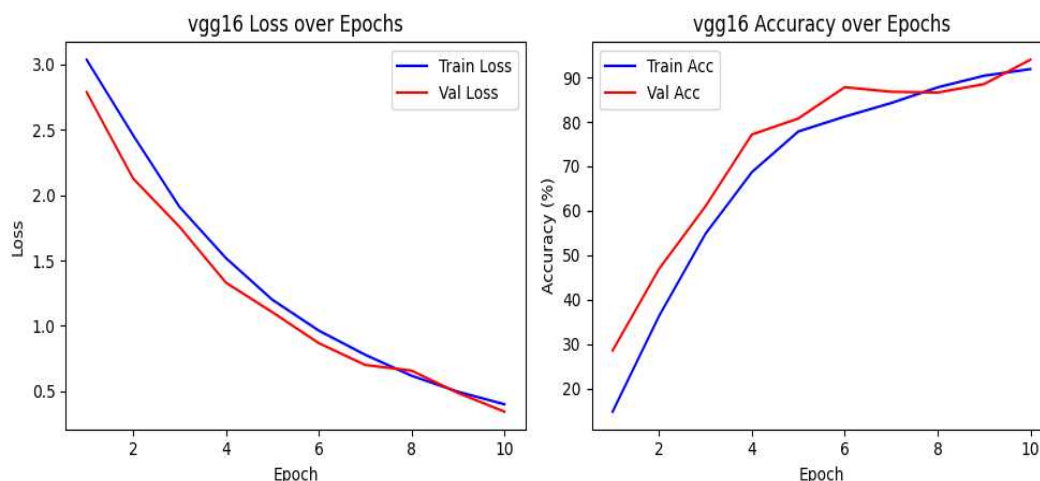
Evaluation Setup	Model	Input Type	Accuracy (%)	Notes
Signer-Dependent	MobileNetV2 + LSTM	RGB Frames	98%	Best trade-off between accuracy and efficiency.
	VGG16 + LSTM	RGB Frames	95%	Higher complexity, moderate gains.
	EfficientNet + LSTM	RGB Frames	80%	Computationally heavy, less consistent with limited dataset size.
Signer-Independent	MobileNetV2 + LSTM	Keypoints (skeleton)	58%	Lower raw accuracy but stronger generalization across unseen signers.

Figure Captions

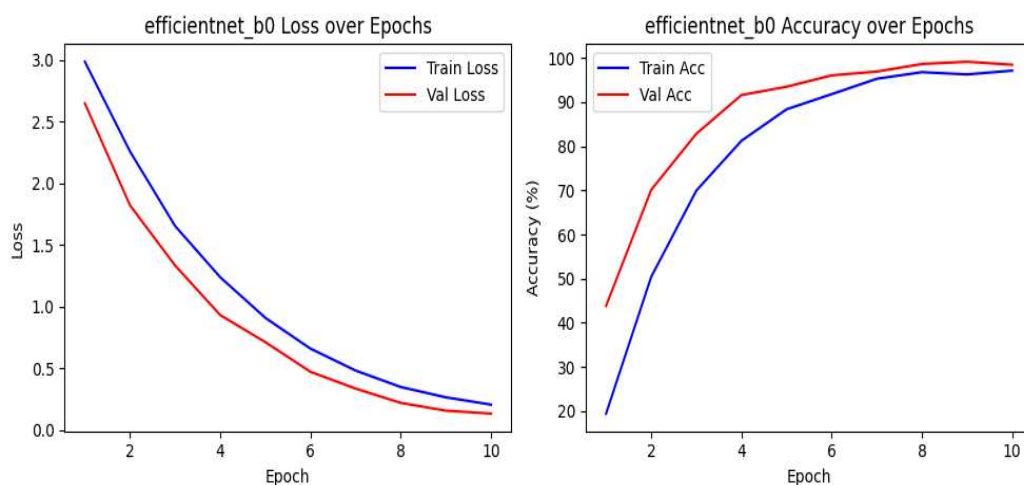
- **Figure 5.2.** Accuracy and loss curves for signer-dependent experiments using MobileNetV2 + LSTM



- **Figure 5.3.** Accuracy and loss curves for signer-dependent experiments using VGG16 + LSTM



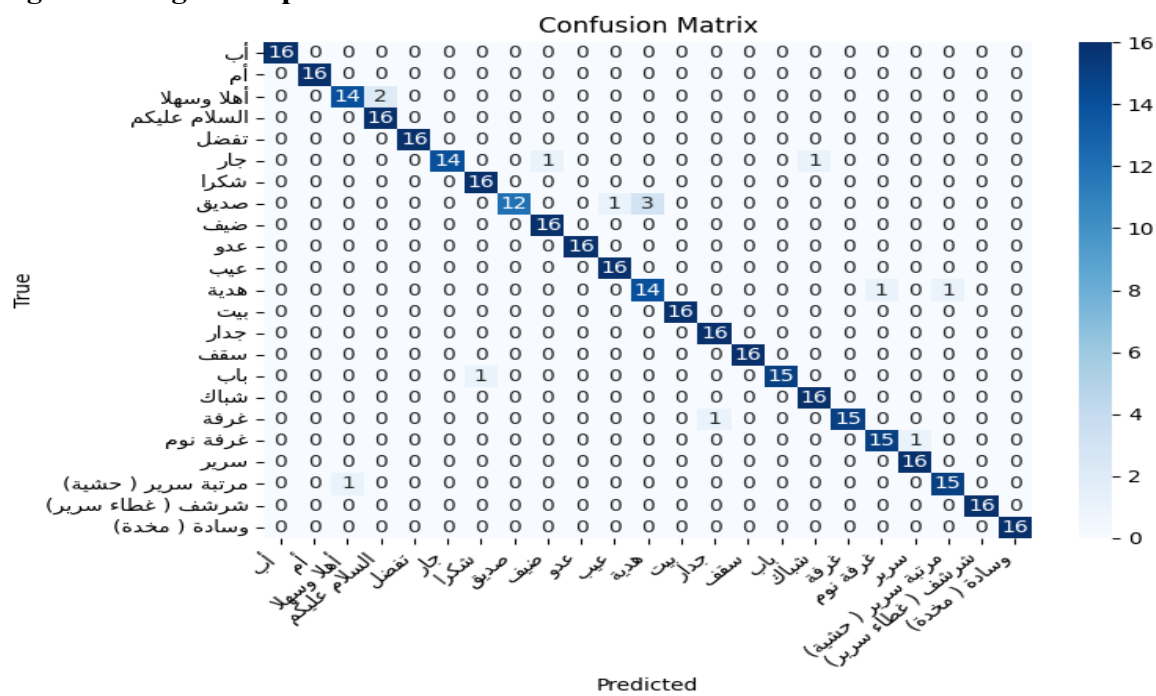
- **Figure 5.4.** Accuracy and loss curves for signer-dependent experiments using Efficientwnt_b0 + LSTM



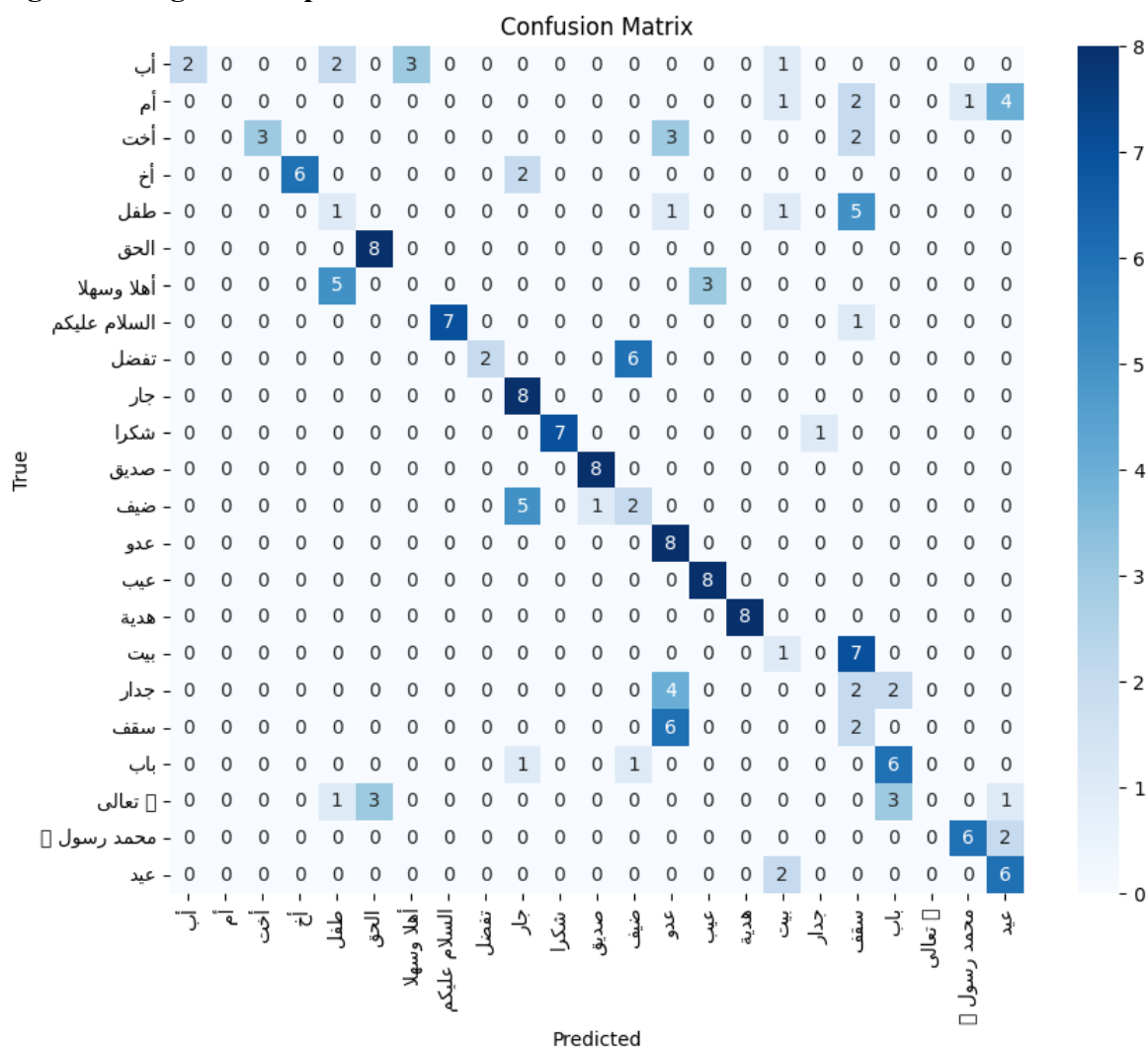
- **Figure 5.5 Signer-Dependent Classification report :**

Class_ID	Class_Name	Precision	Recall	F1_Score	Support	Overall_Accuracy
0	0	1.0000	0.9375	0.9677	16	0.9701
1	1	0.9412	1.0000	0.9697	16	0.9701
2	2	1.0000	0.8750	0.9333	16	0.9701
3	3	0.8889	1.0000	0.9412	16	0.9701
4	4	0.9412	1.0000	0.9697	16	0.9701
5	5	1.0000	0.9375	0.9677	16	0.9701
6	6	0.9412	1.0000	0.9697	16	0.9701
7	7	1.0000	1.0000	1.0000	16	0.9701
8	8	0.9412	1.0000	0.9697	16	0.9701
9	9	0.8889	1.0000	0.9412	16	0.9701
10	10	1.0000	1.0000	1.0000	16	0.9701
11	11	1.0000	1.0000	1.0000	16	0.9701
12	12	1.0000	1.0000	1.0000	16	0.9701
13	13	1.0000	1.0000	1.0000	16	0.9701
14	14	1.0000	0.9375	0.9677	16	0.9701
15	15	1.0000	0.9375	0.9677	16	0.9701
16	16	0.9412	1.0000	0.9697	16	0.9701
17	17	1.0000	1.0000	1.0000	16	0.9701
18	18	0.9375	0.9375	0.9375	16	0.9701
19	19	0.9333	0.8750	0.9032	16	0.9701
20	20	1.0000	0.9375	0.9677	16	0.9701
21	21	1.0000	1.0000	1.0000	16	0.9701
22	22	1.0000	0.9375	0.9677	16	0.9701

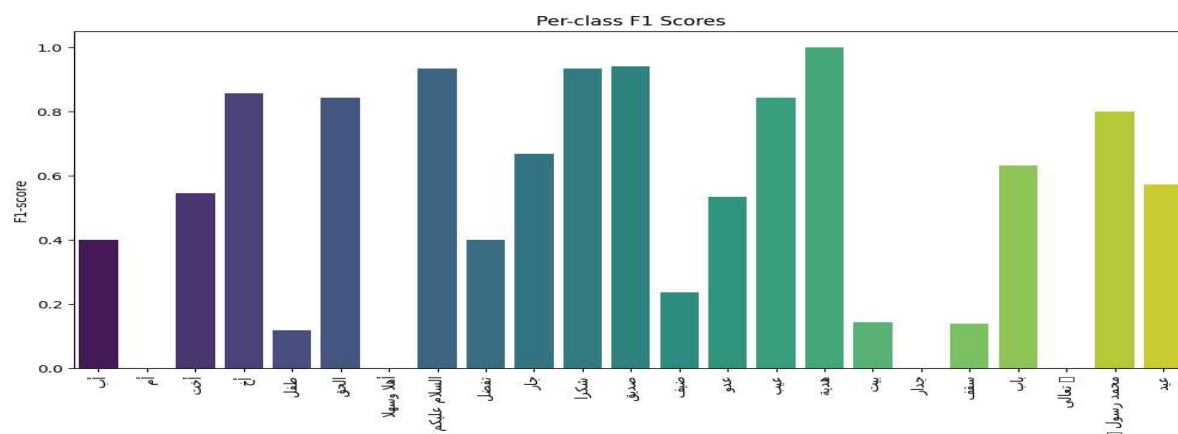
- Figure 5.6 Signer-Dependent Confusion Matrix:



• **Figure 5.7 Signer-Independent Confusion Matrix:**



• **Figure 5.8 F1-score Per Class :**



Chapter 6 Conclusion and Future Work

This research investigated Arabic Sign Language recognition using the KArSL dataset under two evaluation protocols: signer-dependent and signer-independent. The experimental results demonstrated that signer-dependent models achieve higher recognition accuracy, primarily because they exploit signer-specific characteristics. However, this advantage does not necessarily translate to real-world applications, where systems must operate effectively with previously unseen signers.

The signer-independent experiments, while more challenging due to limited signer diversity in the dataset, provided stronger evidence of robustness and generalization. By leveraging keypoint-based inputs combined with temporal modeling through LSTMs, the system achieved promising performance despite the increased complexity. These results confirm that signer-independent evaluation is a more realistic benchmark for practical deployment of sign language recognition systems.

Future work will focus on strengthening the signer-independent approach to further improve accuracy and reliability. In particular:

1. **Dataset Expansion** – Increasing the number of signers and diversifying demographic representation (e.g., age, gender, signing style) to enhance the model’s ability to generalize.
2. **Advanced Architectures** – Incorporating attention mechanisms, transformers, or graph neural networks to better capture spatio-temporal dependencies between keypoints.
3. **Multimodal Fusion** – Combining RGB, depth, and skeleton modalities to enrich the representation of gestures and reduce ambiguity in visually similar signs.
4. **Domain Adaptation** – Exploring techniques such as transfer learning and adversarial training to adapt models to unseen signers and reduce performance gaps between signer-dependent and signer-independent settings.
5. **Real-Time Deployment** – Optimizing models for mobile and embedded platforms to enable interactive applications for deaf and hard-of-hearing communities.

In summary, while signer-dependent models show strong accuracy, the **signer-independent approach is the key to building scalable, real-world systems**. By continuing to refine this strategy, future work can move closer to creating robust, user-agnostic sign language recognition technologies that serve diverse populations.

References

1. Alnuaim, Abeer, Zakariah, Mohammed, Hatamleh, WesamAtef, Tarazi, Hussam, Tripathi, Vikas, Amoatey, Enoch Tetteh, Human-Computer Interaction with Hand Gesture Recognition Using ResNet and MobileNet, *Computational Intelligence and Neuroscience*, 2022, 8777355, 16 pages, 2022. <https://doi.org/10.1155/2022/8777355>
2. Alani, Ali A. and Georgina Cosma. "ArSL-CNN a convolutional neural network for Arabic sign language gesture recognition." *Indonesian Journal of Electrical Engineering and Computer Science* 22 (2021): 1096-1107.
3. Abu-Jamie, T.N., & Abu-Naser, S.S. (2022). Classification of sign-language using MobileNet - Deep Learning. *International Journal of Academic Information Systems Research (IJAIRS)*, 6(7), 29–40.
4. Suliman, W., Deriche, M., Luqman, H., & Mohandes, M.A. (2021). Arabic Sign Language Recognition Using Deep Machine Learning. *2021 4th International Symposium on Advanced Electrical and Communication Technologies (ISAECT)*, 1-4.
5. Bragg, D., Koller, O., Bellard, M., Berke, L., Boudreault, P., Braffort, A., Caselli, N., Huenerfauth, M., Kacorri, H., Verhoef, T., et al. (2019). Sign language recognition, generation, and translation: An interdisciplinary perspective. In *The 21st International ACM SIGACCESS Conference on Computers and Accessibility* (pp. 16–31).
6. Cooper, H., Holt, B., & Bowden, R. (2011). Sign language recognition. In T.B. Moeslund, A. Hilton, V. Krüger, & L. Sigal (Eds.), *Sign Language Recognition* (pp. 539–562). Springer. https://doi.org/10.1007/978-0-85729-997-0_27
7. Ahmed, S., & Bons, M. (2020). Edge computed NILM: A phone-based implementation using MobileNet compressed by TensorFlow Lite. In *Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring* (pp. 44–48).
8. Akbar, A.S., Fatichah, C., & Suciati, N. (2020). Modified MobileNet for patient survival prediction. In *International MICCAI Brainlesion Workshop* (pp. 374–387). Springer.
9. Eberhard, D., Fennig, C.D., Simons, G.F., & M, D. (2021). Sign language. Dallas, Texas: SIL International. Retrieved from <https://www.ethnologue.com/subgroups/sign-language>
10. Mashael, M.A., Abdelwahed, M., & Suhandi, D. (2024). Robust sign language detection for hearing disabled persons by improved Coyote Optimization Algorithm with deep learning. *AIMS Mathematics*, 9(6), 15911–15927. <https://doi.org/10.3934/math.2024769>
11. Mogeib, A.A.M., Adel, A., Abdu, H.G., Bader, F.A., & Manal, A.Q. (2024). A bidirectional Arabic sign language framework using deep learning and fuzzy matching score. *Mathematics*, 12(8), 1155. <https://doi.org/10.3390/math12081155>
12. Latif, G., Mohammad, N., AlKhalaf, R., Alkhalaf, R., Alghazo, J.M., & Khan, M.A. (2020). An Automatic Arabic Sign Language Recognition System based on Deep CNN: An Assistive System for the Deaf and Hard of Hearing. *International Journal of Computing and Digital Systems*.

13. Sagheer, N.S., Almasoudy, F.H., & Bashaa, M.H. (2024). Enhancing Arabic Sign Language Recognition using Deep Learning. *International Journal of Innovative Technology and Exploring Engineering*.
14. Batool Yahya AlKhuraym, Mohamed Maher Ben Ismail and Ouiem Bchir, "Arabic Sign Language Recognition using Lightweight CNN-based Architecture" *International Journal of Advanced Computer Science and Applications(IJACSA)*, 13(4), 2022. <http://dx.doi.org/10.14569/IJACSA.2022.0130438>
15. Alsaadi, Z., Alshamani, E., Alrehaili, M., Alrashdi, A. A. D., Albelwi, S., & Elfaki, A. O. (2022). A Real Time Arabic Sign Language Alphabets (ArSLA) Recognition Model Using Deep Learning Architecture. *Computers*, 11(5), 78. <https://doi.org/10.3390/computers11050078>
16. Althagafi, A., Althobaiti, G., Alsubait, T., & Alqurashi, T. (2020). ASLR: Arabic Sign Language Recognition Using Convolutional Neural Networks.
17. Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A comprehensive survey on transfer learning. *Proc. IEEE*2020,109, 43–76.
18. Perez, L.; Wang, J. The effectiveness of data augmentation in image classification using deep learning. *arXiv*2017, arXiv:1712.04621.
19. Podder, K. K., Ezeddin, M., Chowdhury, M. E. H., Sumon, M. S. I., Tahir, A. M., Ayari, M. A., Dutta, P., Khandakar, A., Mahbub, Z. B., & Kadir, M. A. (2023). Signer-Independent Arabic Sign Language Recognition System Using Deep Learning Model. *Sensors*, 23(16), 7156. <https://doi.org/10.3390/s23167156>.
20. Ghazanfar Latif 1* , Nazeeruddin Mohammad 2 , Roaa AlKhalaf 1 , Rawan AlKhalaf 2 , Jaafar Alghazo 3and Majid A. Khan(2022)
21. Hama Rawf, Karwan & Mohammed, Aree & Abdulrahman, Ayub & Abdalla, Peshraw & Ghafoor, Karzan. (2022). A COMPARATIVE STUDY USING 2D CNN AND TRANSFER LEARNING TO DETECT AND CLASSIFY ARABIC-SCRIPT-BASED SIGN LANGUAGE. *Acta Informatica Malaysia*. 8-14. 10.26480/aim.01.2023.08.14.