People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
AKLI MOHAND OULHADJ UNIVERSITY -BOUIRA
Faculty of Sciences and Applied Sciences
Computer Science Department

Master Thesis

In Computer Science

Speciality : *Information Systems and Software Engineering -ISIL-*

Theme

A Computer-Assisted Tool to Learn Algerian Sign Language

Supervised by

— Dr. ZERROUKI Taha
— Dr. BADIS Lyes

Realized by

— SAHALI Farouk

2023/2024

Acknowledgments

Alhamdolilah, I thank God for giving me everything and for never leaving me alone.

I would like to express my deepest gratitude to my father, an expert in information systems, an electronics engineer, and a Microsoft MSCE, for his invaluable help, challenges in the field, and constant advice. I also extend my heartfelt thanks to my mother, an engineer in chemistry, for always being there for me, giving me courage, and supporting me throughout this journey.

Special thanks to my little sister for her constant encouragement and support.

I am profoundly grateful to my supervisor, Dr. Taha Zerroki, for his trust in me and for agreeing to guide me in this project. My sincere thanks also go to Dr. Badis Lyes, who provided significant assistance in this work.

Additionally, I want to acknowledge my master, Boulil Moussa, 4th Dan Aikido and 2nd Dan Judo, for his advice, support in sports and studies, and unwavering encouragement.

Lastly, I would like to thank all my friends for their continuous support and encouragement.

Dedications

I dedicate this work to my family—my father, my mother, and my younger sister—for their unwavering support and encouragement.

To all my friends and sports companions, your camaraderie has been invaluable.

Sahali Farouk.

Abstract

Sign language plays a pivotal role in facilitating communication for the hearing impaired, providing a visual medium for expression through gestures and signs. This thesis delves into the multifaceted realm of sign language, exploring its definitions, cultural diversity, and the evolution of recognition technologies. The exploration commences with an examination of the foundational definitions and diverse manifestations of sign language across different nations, with a particular emphasis on Algerian Sign Language (LSA) and its distinctive structural attributes.

The study further scrutinizes various techniques employed for sign recognition, encompassing advancements in computer vision, neural networks, and traditional classification methodologies. These methodologies are dissected to elucidate their practical applications and inherent challenges within real-world contexts. Notably, the thesis elucidates the complexities and nuances involved in accurately interpreting sign language gestures, underscoring the pivotal role of technology in bridging communication barriers.

Central to this research is the proposed system architecture, meticulously designed to encompass the processes of sign extraction, classification, and interpretation. Emphasizing the integration of sophisticated machine learning algorithms, the architecture aims to bolster precision and operational efficiency in sign language recognition systems. By advancing these technologies, this research contributes significantly to fostering inclusivity and accessibility for the hearing impaired community, thereby promoting equitable participation across social, educational, and professional domains.

Keywords

Sign Language, Sign Recognition, Algeria, Characteristics of Signs, Types of Signs, Computer Vision Technologies, Machine Learning, Neural Networks, Deep Learning, Natural Language Processing (NLP), System Architecture, Sign Extraction, Sign Classification, Translation Systems, Mobile Applications,Accessibility,Detection Techniques,Facial Expressions,Dataset,Development Environment,Libraries and Frameworks,Sign language recognition, Algerian sign language,Computer vision, Long Short Term Memory

Résumé

La langue des signes joue un rôle central dans la facilitation de la communication pour les malentendants, en fournissant un moyen visuel d'expression à travers des gestes et des signes. Cette thèse explore le domaine multiforme de la langue des signes, explorant ses définitions, sa diversité culturelle et l'évolution des technologies de reconnaissance. L'exploration commence par un examen des définitions fondamentales et des diverses manifestations de la langue des signes dans différents pays, avec un accent particulier sur la langue des signes algérienne (LSA) et ses attributs structurels distinctifs.

L'étude examine en outre diverses techniques utilisées pour la reconnaissance des signes, englobant les progrès de la vision par ordinateur, des réseaux neuronaux et des méthodologies de classification traditionnelles. Ces méthodologies sont décortiquées pour élucider leurs applications pratiques et les défis inhérents dans des contextes du monde réel. Notamment, la thèse élucide les complexités et les nuances impliquées dans l'interprétation précise des gestes en langue des signes, soulignant le rôle central de la technologie dans la réduction des barrières de communication.

Au cœur de cette recherche se trouve l'architecture système proposée, méticuleusement conçue pour englober les processus d'extraction, de classification et d'interprétation des signes. En mettant l'accent sur l'intégration d'algorithmes sophistiqués d'apprentissage automatique, l'architecture vise à renforcer la précision et l'efficacité opérationnelle des systèmes de reconnaissance en langue des signes. En faisant progresser ces technologies, cette recherche contribue de manière significative à favoriser l'inclusion et l'accessibilité pour la communauté des malentendants, favorisant ainsi une participation équitable dans les domaines sociaux, éducatifs et professionnels.

Mots-clés

Langue des signes, Reconnaissance des signes, Algérie, Caractéristiques des signes, Types de signes, Technologies de vision par ordinateur, Apprentissage automatique, Réseaux de neurones, Apprentissage profond, Traitement du langage naturel (NLP), Architecture du système, Extraction de signes, Classification des signes, Systèmes de traduction, Applications mobiles, Accessibilité, Techniques de détection, Expressions faciales, Ensemble de données, Environnement de développement, Bibliothèques et frameworks, Reconnaissance de la langue des signes, Langue des signes algérienne, Vision par ordinateur, Mémoire à long et court terme.

ملخص

تلعب لغة الإشارة دوراً مركزياً في تسهيل التواصل لدى ضعاف السمع، من خلال توفير وسيلة بصرية للتعبير من خلال الإيماءات والإشارات. تستكشف هذه الأطروحة المجال المتعدد الأوجه للغة الإشارة، وتستكشف تعريفاتها وتنوعها الثقافي وتطور تقنيات التعرف عليها. يبدأ الاستكشاف بفحص التعريفات الأساسية والمظاهر المختلفة للغة الإشارة في مختلف البلدان، مع التركيز بشكل خاص على لغة الإشارة الجزائرية (اصى) وخصائصها الهيكلية المميزة.

تتناول الدراسة أيضًا التقنيات المختلفة المستخدمة للتعرف على الإشارات، بما في ذلك التطورات في رؤية الكمبيوتر والشبكات العصبية ومنهجيات التصنيف التقليدية. ويتم تشريح هذه المنهجيات لتوضيح تطبيقاتها العملية والتحديات الكامنة في سياقات العالم الحقيقي. ومن الجدير بالذكر أن الأطروحة توضح التعقيدات والفروق الدقيقة التي ينطوي عليها التفسير الدقيق للإيماءات في لغة الإشارة، وتسلط الضوء على الدور المركزي للتكنولوجيا في تقليل حواجز الاتصال.

في قلب هذا البحث توجد بنية النظام المقترحة، المصممة بدقة لتشمل عمليات استخراج العلامات وتصنيفها وتفسيرها. ومع التركيز على تكامل خوارزميات التعلم الآلي المتطورة، تهدف البنية إلى تعزيز الدقة والكفاءة التشغيلية لأنظمة التعرف على لغة الإشارة. ومن خلال تطوير هذه التقنيات، يساهم هذا البحث بشكل كبير في تعزيز الإدماج وإمكانية الوصول لمجتمع ضعاف السمع، وبالتالي تعزيز المشاركة العادلة في المجالات الاجتماعية والتعليمية والمهنية.

الكلمات المفتاحية

لغة الإشارة، التعرف على الإشارات، الجزائر، خصائص الإشارات، أنواع الإشارات، تقنيات رؤية الكمبيوتر، التعلم الآلي، الشبكات العصبية، التعلم العميق، معالجة اللغة الطبيعية (بع)، بنية النظام، استخراج الإشارات، تصنيف الإشارات، أنظمة الترجمة، التطبيقات المحمولة، إمكانية الوصول، تقنيات الكشف، تعبيرات الوجه، مجموعة البيانات، بيئة التطوير، المكتبات والأطر، التعرف على لغة الإشارة، لغة الإشارة الجزائرية، رؤية الكمبيوتر، الذاكرة الطويلة والقصيرة الأجل

Table des matières

Table des figures

Liste des tableaux

General introduction

Sign language serves as a crucial means of communication for the hearing impaired, enabling expressive and interactive communication through visual gestures and signs. This thesis delves into the multifaceted realm of sign language, encompassing its definitions and the diverse variations found across different countries. By exploring these linguistic variations, including Algerian Sign Language (LSA), the study aims to shed light on the cultural and structural nuances that shape this unique form of communication.

Advancements in machine learning and computer vision have revolutionized the field of sign language recognition, offering promising avenues for enhancing accessibility and inclusivity. These technologies enable the development of intelligent systems capable of interpreting and translating sign language into text or spoken language, thereby bridging communication gaps and empowering individuals within the hearing impaired community. Understanding the complexities of sign language—its types, structures, and alphabets—is pivotal for designing effective recognition systems.

This thesis explores various techniques employed in sign recognition, such as hand and gesture detection using computer vision, and the application of neural networks like Recurrent Neural Networks (RNNs). These technologies not only enhance the accuracy and efficiency of sign language interpretation but also pave the way for innovative applications in education, accessibility, and communication technology.

By synthesizing theoretical insights with practical applications, this research contributes to the evolving landscape of sign language technology. It aims to foster a deeper understanding of how technological advancements can be leveraged to empower individuals with hearing impairments, promoting inclusivity and enhancing quality of life through improved communication channels.

This thesis is structured into five chapters, each addressing crucial aspects of sign language recognition and technology :

Chapter 1 : Introduction This chapter sets the stage by highlighting the importance of sign language, outlining the scope of the thesis, and emphasizing the transformative potential of machine learning and computer vision in this field.

Chapter 2 : Sign Recognition This chapter explores the definition of sign recognition, its applications, and various techniques used in the field, including computer vision and deep learning methods.

Chapter 3 : Machine Learning and Computer Vision This chapter focuses on the applications of machine learning and computer vision techniques specifically for sign language recognition, detailing methods such as hand and gesture detection and neural network architectures.

Chapter 4 : Design of the Proposed System This chapter presents the design architecture of a proposed system for sign language recognition, discussing its components and the rationale behind its development.

Chapter 5 : Implementation This chapter details the practical implementation aspects of the proposed system, including setting up the environment, data processing, and system testing.

Each chapter contributes uniquely to advancing the understanding and application of sign language technology, aiming to improve accessibility and communication for the hearing impaired community.

Chapter 1

Sign Language

1.1 Introduction

Sign language plays a crucial role in sign language learning apps by enabling effective and inclusive communication for people who are deaf or hard of hearing.Sign language learning apps use sign language as a means of teaching and communication, providing users with the opportunity to learn and practice sign language in an interactive and accessible way.

Thanks to sign language, people who are deaf or hard of hearing can communicate, express themselves and interact with others in a natural and fluid way. Sign language learning apps integrate videos, images, and interactive exercises to help learners understand and master sign language gestures and expressions.

So in this chapter we will see the definition of sign languages and these examples and these characteristics.

1.2 Definition of sign language

Sign languages are visual-gestural languages that use the hands and arms as well as non-manual means such as facial expressions, head movements, and body postures to convey linguistic messages [1].

Sign language, any means of communication through bodily movements, especially of the hands and arms, used when spoken communication is impossible or not desirable. The practice is probably older than speech. Sign language may be as coarsely expressed as mere grimaces, shrugs, or pointings, or it may employ a delicately nuanced combination of coded manual signals reinforced by facial expression and perhaps augmented by words spelled out in a manual alphabet[2].

Sign languages(SL) are the natural languages of deaf people. We write "the sign languages" in the plural, because contrary to what the majority of people think, it there is not a single international sign language, but one sign language for country where there is to deaf people [3].

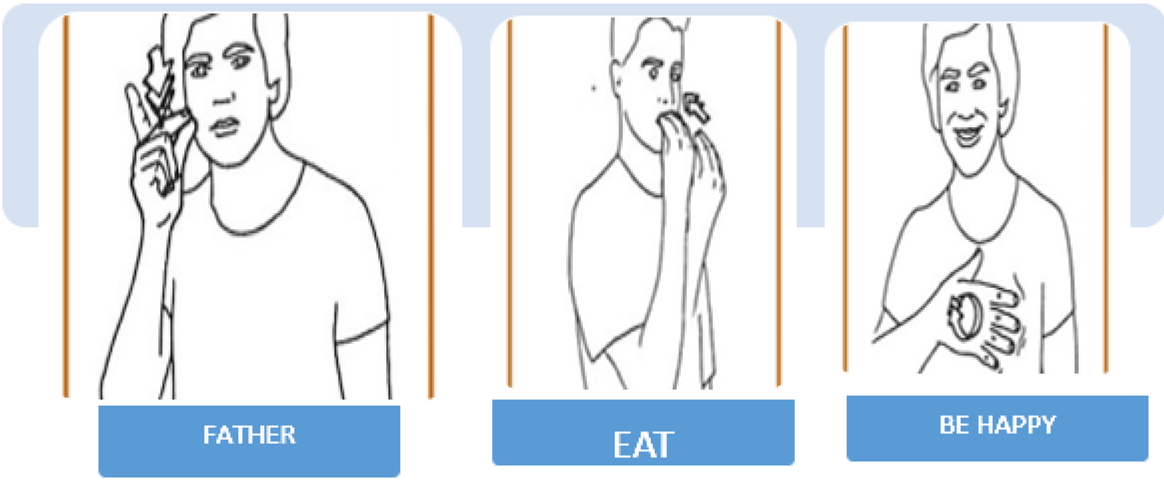

FIGURE 1.1 – Example Of Words In Sign Language .

.

1.3 The varieties of sign language

Today, there are more than 300 different sign languages in the world, spoken by more than 72 million deaf or hard-of-hearing people worldwide.[4]

As we said above, around 300 sign languages are used worldwide today, and most of them vary significantly.

Along with BSL, there are several sign languages used by English-speaking countries, including the US (ASL), Auslan and NZSL. Ireland also has its own sign language (ISL), which is derived from French Sign Language but shares similarities with BSL.

One of the most widely used sign languages around the world is Chinese Sign Language (CSL or ZGS), which has up to 20 million users.

Brazilian Sign Language has around three million users worldwide, while Indo-Pakistani Sign Language has about 1.8 million users across South Asia.

Back in the UK, English Sign Language Support (ESS) and Makaton are both used alongside BSL (British Sign Language) to support deaf and hard of hearing people with additional learning needs. This diverse approach demonstrates the UK's commitment to providing support that is inclusive and tailored to the individual needs of deaf and hard of hearing people.[5]

1.3.1 Examples of different sign languages by country

— British sgn language, auslan and new zealand sign language (BANZSL)
— French Sign Language (LSF)
— American Sign Language (ASL)
— Irish Sign Language (ISL)
— Chinese Sign Language (CSL or ZGS)
— Arabic Sign Language (ArSL)

I.British sgn language, auslan and new zealand sign language (BANZSL) :

British Sign Language, codified in British schools for the deaf in the 1700s, spread around the world as the British Empire and Commonwealth did. This included reaching both Australia and New Zealand. Thus, New Zealand Sign Language and Auslan, Australian Sign Language, share the same manual alphabet, grammar and much of the same lexicon (that is to say the same signs) as BSL. So much so that a single phrase – BANZSL – was coined to represent them as a single language with three dialects [6].

II.French Sign Language (FSL) :

LSF currently has around 100,000 native language users (in France and in French-speaking parts of Switzerland). It consists of a one-handed fingerspelling system, which has been borrowed by many other signed languages such as ASL and Brazilian Sign Language (LIBRAS). Before the emergence of LSF, there was Old French Sign Language. Supposedly created by Charles Michel de l'Épée, who opened the first free deaf school in France. After 1880, sign language was banned from schools as it was thought to be restricting the students in their learning. Therefore, instead they were taught using the oral approach of communication. In 2005 LSF was officially recognised as a language in its own right and has "become a pillar in the identity of deaf culture"[7].

III.American Sign Language (ASL) :

American Sign Language (ASL) is a complete, natural language that has the same linguistic properties as spoken languages, with grammar that differs from English. ASL is expressed by movements of the hands and face. It is the primary language of many North Americans who are deaf and hard of hearing and is used by some hearing people as well [8].

IV.Irish Sign Language (ISL) :

Irish Sign Language (ISL) is a unique and indigenous sign language used in Ireland. It became a recognized language in the country with the passing of the Irish Sign Language Act in 2017. ISL is crucial for the Deaf community in Ireland, as highlighted by the efforts of Deaf activists who played a significant role in the ISL recognition movement leading up to the passing of the Act. Researchers have been exploring various aspects of ISL, such as its linguistic framework and computational models. A linguistically motivated computational framework for ISL is being developed to define the architecture of ISL in linguistic terms . Additionally, studies have focused on language planning in ISL, particularly in terms of developing new terminology and vocabulary for different domains within the language[9].

V.Chinese Sign Language (CSL or ZGS) :

Chinese Sign Language (CSL), also known as ZGS, is a unique sign language used by the deaf community in China. The abbreviation CSL stands for Chinese Sign Language, while ZGS is another term used to refer to the same sign language system. The use of multiple names or abbreviations for sign languages is not uncommon, as different regions or communities may have their own ways of referring to the same language. In this case, both CSL and ZGS are used interchangeably to denote the sign language used in China by the deaf population[10].

VI.Arabic Sign Language (ArSL) :

Arabic Sign Language (ArSL) is a visual language used by deaf Arabs and some hard of hearing to translate their thinking. It is a language in its own right as well as spoken languages such as Arabic or English. It is produced by body, face, and gestures of the hands. Hundreds of thousands of deaf people around the world currently practice it [11].

1.4 Algerian sign language (ASL)

The Algerian Sign Language (LSA) is the gestural language used by deaf individuals and their associates in Algeria to communicate through signs. Algerian Sign Language is officially recognized by the law of May 8, 2002, as the primary language of the deaf community in Algeria, which is the only country in the Arab world and Africa to officially recognize sign language.[12]

LSA is linked to French sign language and has no direct link with Arabic sign languages.[13]

LSA is entirely based on gestures (signs), each sign being made using different parts of the body, such as the hand(s), face, shoulder, or even the entire body. Intuitively speaking, LSA is a language like any other. Indeed, it has a vocabulary and organized syntax just like spoken languages. Therefore, to learn and understand LSA, one simply needs to know its alphabet. In fact, every sign in the LSA alphabet is generated by one or two specific hand postures.[14]

1.4.1 Varieties of algerian sign language

it seems that almost every Algerian Deaf community living in a different province or village in Algeria is likely to develop its own dialect of Algerian Sign Language.

Probably, different Algerian Sign Languages are being used as many Deaf communities are in Algeria, at least in some big cities, regardless of their similarities and/or differences.

These Algerian Sign Languages could be developed mainly in some villages and used by the Algerian Deaf individuals, and perhaps, some of their family members, relatives, friends, or the ones who work/Deal with them in general[15].

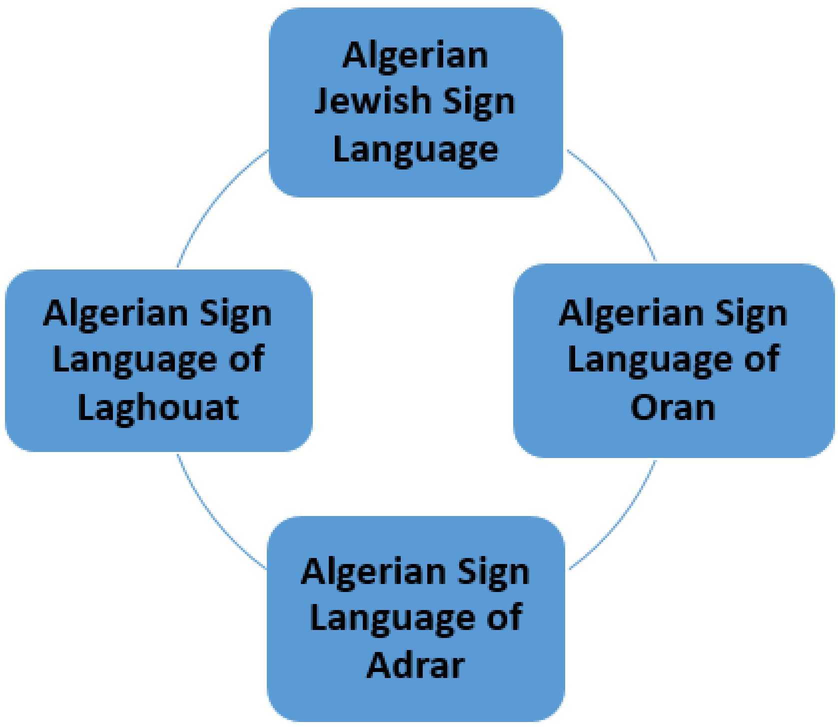

FIGURE 1.2 – Varieties of algerian sign language

I.Algerian Jewish Sign Language : Language since it was mainly developed and used in the village of Ghardaia by the Algerian Jewish individuals at that time .

II.Algerian Sign Language of Laghouat : Which is used by many Deaf people in Laghouat province and other cities (or villages) around it (Djama, 2016).

III.Algerian Sign Language of Oran :It is used by the Deaf in the North of Algeria, particularly in the city of Oran (Mansour, 2007).

IV.Algerian Sign Language of Adrar : Which is used by the Algerian Deaf community in Adrar, in the South of Algeria (Abdelouafi,2018).

1.5 Characteristics of sign language

1.5.1 Types of signs

A sign is a class of gestures which depends or not on a certain duration in time[16]. We can first classify gestures according to the parts of the body involved. We generally distinguishes three types of gestures [17] :

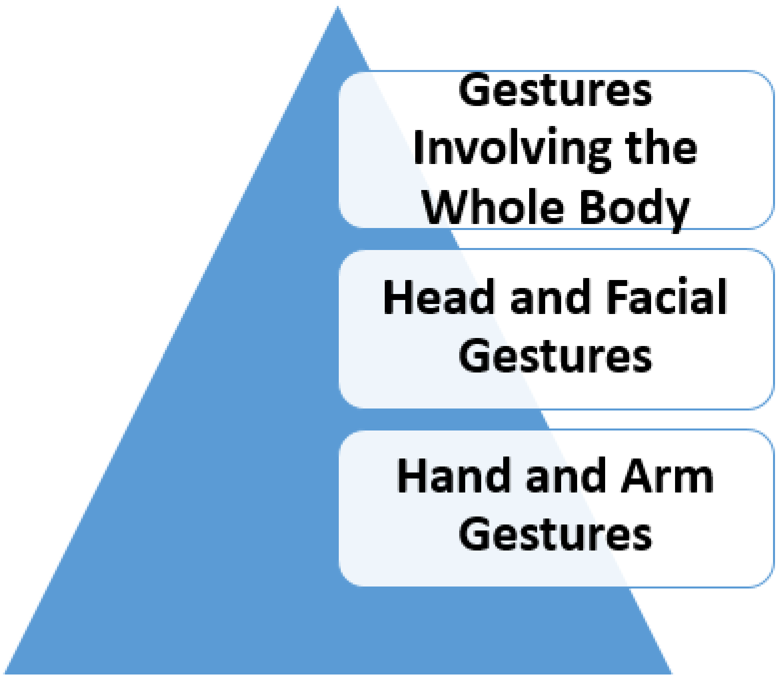

FIGURE 1.3 – Three Types Of Gestures

I.Gestures involving the whole body

When considering gestures involving the whole body, it is important to recognize that different types of gestures can utilize various parts of the body for communication.For instance, whole-body gestures can involve movements of the hands, forearms, arms, and even the entire body itself .These gestures are not limited to just hand movements but can encompass a wide range of bodily expressions.

Furthermore, in the context of gesture elicitation studies, researchers have explored the dissimilarity-consensus approach to analyze agreement in whole-body gestures. This method aims to understand the variability and consensus in gestures elicited from users, especially in scenarios involving children where maximizing the variance of elicited gestures is crucial[18]"See figure 1.4".

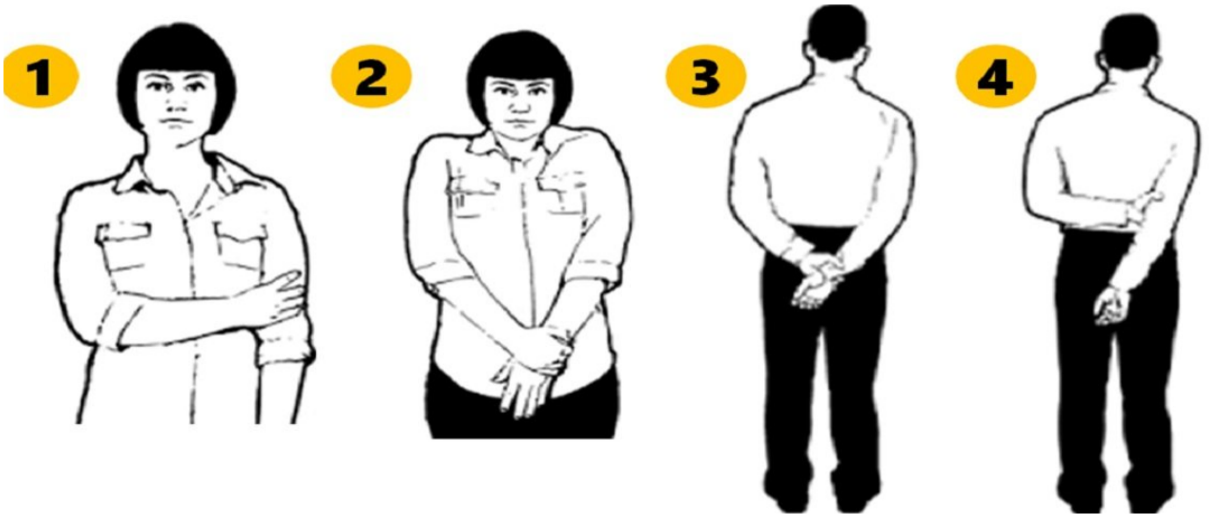

FIGURE 1.4 – Gestures Involving the Whole Body.

II.Head and facial gestures

Few head gestures have a specific meaning, the orientation of the head is as to it very useful for detecting the field of vision [17] "See Figure 1.5" .

III.Hand and arm gestures

They form the main category of interactive gestures. The hand allows you to make precise and complex gestures. Research around these gestures mainly concerns the recognition of hand positions, interpretation of sign language and allowing manipulation and interaction with data or elements of an environment[17] "See Figure 1.6".

1.5.2 Structure of sign language

Each gesture of a hand can be broken down into five parameters which are independent and can be both dynamic and invariant during the emission of the sign. These parameters

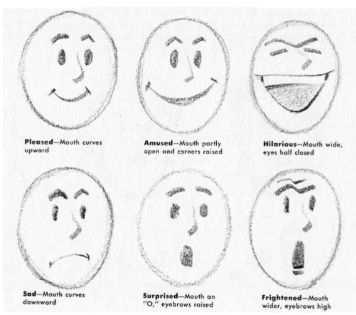

FIGURE 1.5 – Facial Gestures.

FIGURE 1.6 – Arm Gestures .

are defined as follows [19] "See Figure 1.7" :

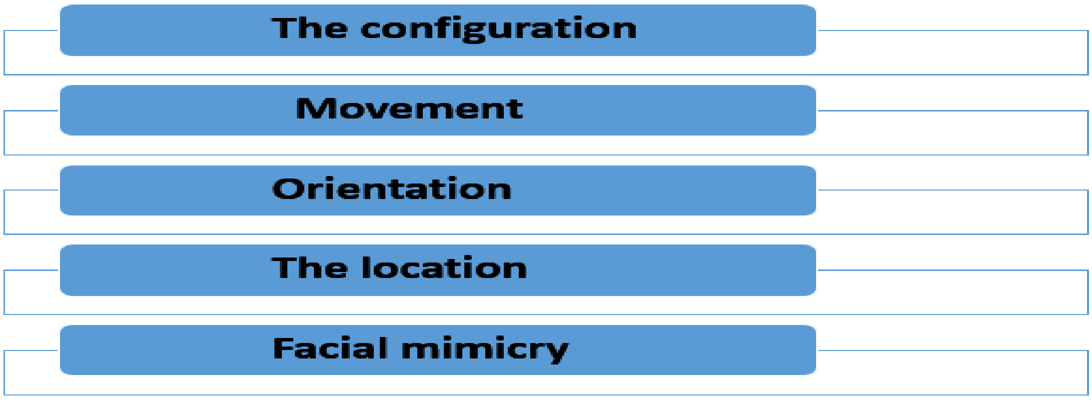

FIGURE 1.7 – Structure of sign language

I.The configuration

In a monomorphemic sign, the handshape consists of one or more selected fingers in a particular position– extended, closed, curved, or bent. Exemplifies these positions for shapes that select all fingers (ignoring the thumb, for simplicity) "See Figure 1.8"[20] :

II.Movement

Movement in structured sign language refers to the dynamic component of sign language that involves the motion and gestures made by the signer to convey meaning. In structured sign languages, such as American Sign Language (ASL) or British Sign Language (BSL), movement plays a crucial role in distinguishing between signs and expressing various linguistic elements.

Here are some key aspects of movement in structured sign language :

Handshape Changes : Movement can involve changes in handshapes to represent different signs. The movement of the hands and fingers in specific configurations contributes

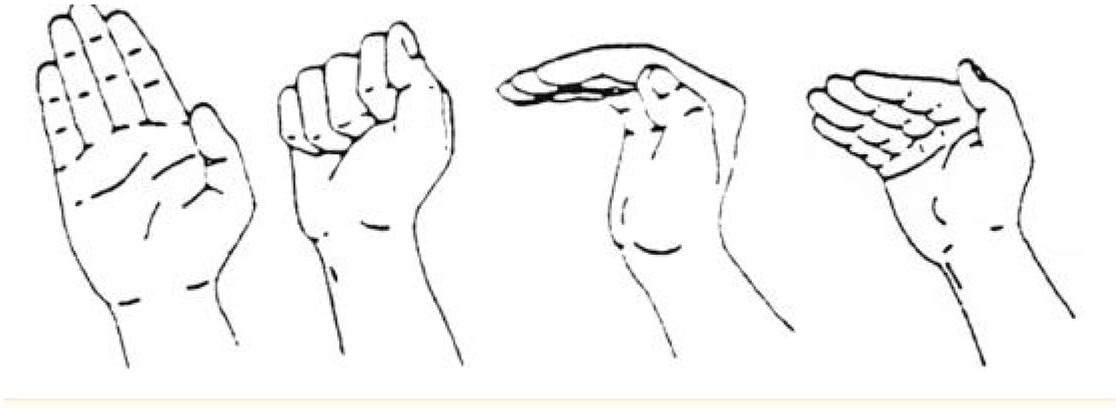

FIGURE 1.8 – Example For Configuration

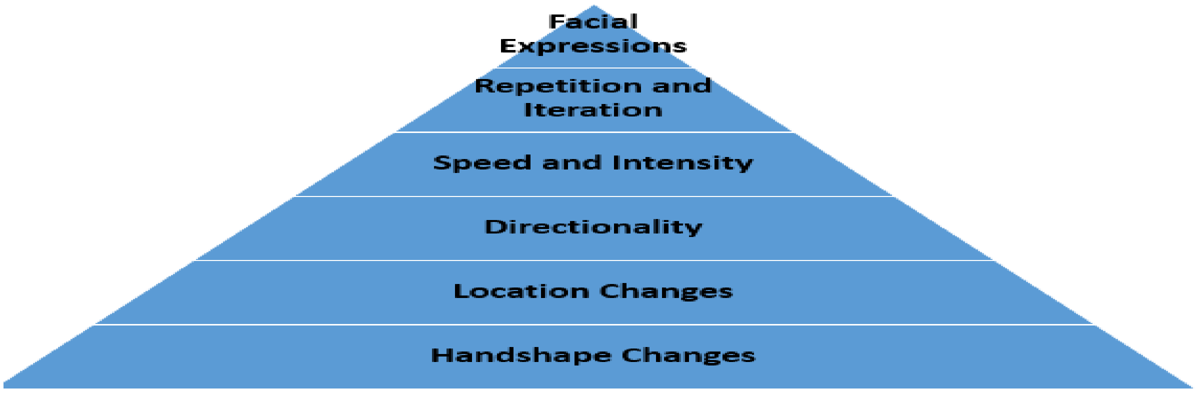

FIGURE 1.9 – Aspects Of Movement

to the meaning of the sign.

Location Changes : Movement can also involve changes in the location of the signs relative to the signer's body or in the signing space. The spatial aspect of movement is essential for indicating relationships between objects or concepts.

Directionality : Movement in sign language can have specific directions, such as upward, downward, left, or right. These directional movements can convey grammatical information, such as verb conjugation or spatial relationships.

Speed and Intensity : The speed and intensity of movement in sign language can affect the meaning and emphasis of signs. Faster movements may indicate urgency or intensity, while slower movements can convey deliberation or emphasis.

Repetition and Iteration : Movement can involve repetition or iteration of signs or gestures to emphasize or clarify meaning. Repetitive movements can also be used for rhythmic or stylistic purposes.

Facial Expressions : Facial expressions are an integral part of movement in sign language. Expressive facial movements, such as eyebrow raises, mouth movements, and eye gaze, complement manual signs and convey emotions, attitudes, and grammatical markers[21] "See Figure 1.10" :

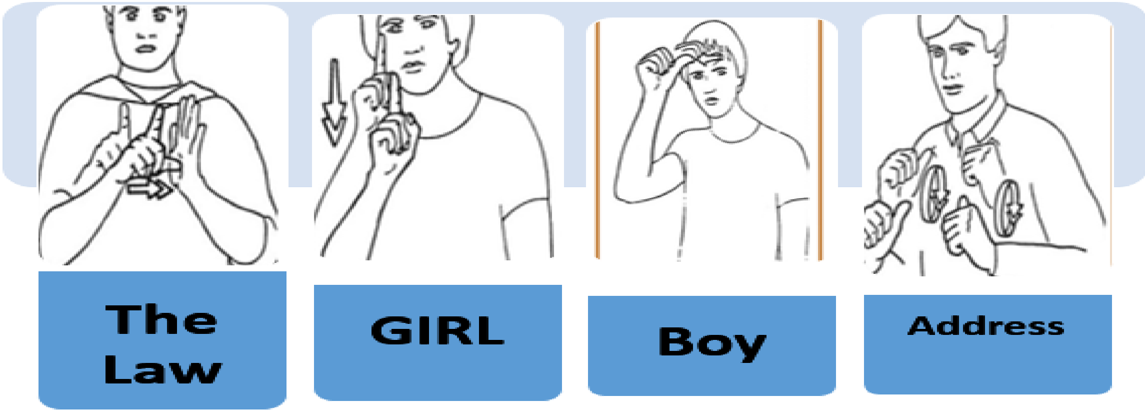

FIGURE 1.10 – Same Sign Different Movement Different Meaning

III.Orientation

The directions towards which the hands and fingers are directed,[22].Orientation refers to the direction of the hand in relation to the signer.As a result, the management of the hand is determined by the position of the palm of the hand relative to the signer[16]

IV.The location

The location designates the location of the hand in space in relation to the body of the signer, or to a particular object, and which can completely change the meaning of a gesture.By example, the hand can generally be in front of the signer, in the area next to the head of the signer, on the mouth, eyes, arm, palm, etc [21].

V.Facial mimicry

In reality, some signs consist of a hand gesture, or hands, and a facial expression, or rather facial expression.In fact, the latter gives meaning to a sign isolated, and is fundamental in the construction of a sentence .

FIGURE 1.11 – Satisfied

1.5.3 Algerian sign language alphabet (ASLA)

In Algerian sign language, there are 42 signs of the Algerian alphabet, among which there are has 37 static signs and 5 dynamic signs.In fact, these signs are represented by a single hand.Furthermore, each of the static signs is determined by means of two parameters which are : configuration and orientation [23].

1.6 Conclusion

In this chapter we presented the definition of sign language as well as these some types in the world and the definition of Algerian sign language and these varieties.

In the next chapter we will present sign recognition information.

Chapter 2

Sign Recognition

2.1 Introduction

Sign recognition, often called character recognition or pattern recognition, is an essential branch of artificial intelligence (AI) and computer vision. This technology aims to enable machines to understand, interpret, and respond to various types of symbols and visual characters, such as letters, numbers, gestures, and other graphic signs.

Sign recognition has its roots in early work on pattern recognition in the 1960s and 1970s. With the evolution of computer technologies and the increase in data processing capabilities, it has progressed to become a key component of many modern applications. The increasing digitalization of information and the need to make systems more interactive and intuitive have increased the importance of this technology.

This chapter explores the foundations, techniques, applications, and challenges associated with sign recognition, providing a comprehensive overview of this dynamic and expanding field.

2.2 Definition of sign recognition

Sign recognition refers to the identification of specific gestures, symbols, or visual cues that allow other people to identify someone or a group. In general, this recognition can apply to various fields such as non-verbal communication, deaf culture, psychiatry, maritime law, etc [24].

It is important to note that in the context of nonverbal communication, sign recognition can also refer to gestures, symbols, or signals used to convey meaning or intention without resorting to speech. Additionally, in specific contexts such as recognition of sign languages, this may also encompass legal or official recognition of sign language in a given country, as mentioned in the search results [25] .

2.3 Applications of sign recognition

These applications demonstrate the diversity of areas in which sign recognition can have a positive impact by facilitating communication, improving accessibility and promoting the inclusion of people who are deaf or hard of hearing in various aspects of daily life [26](see figure 2.1).

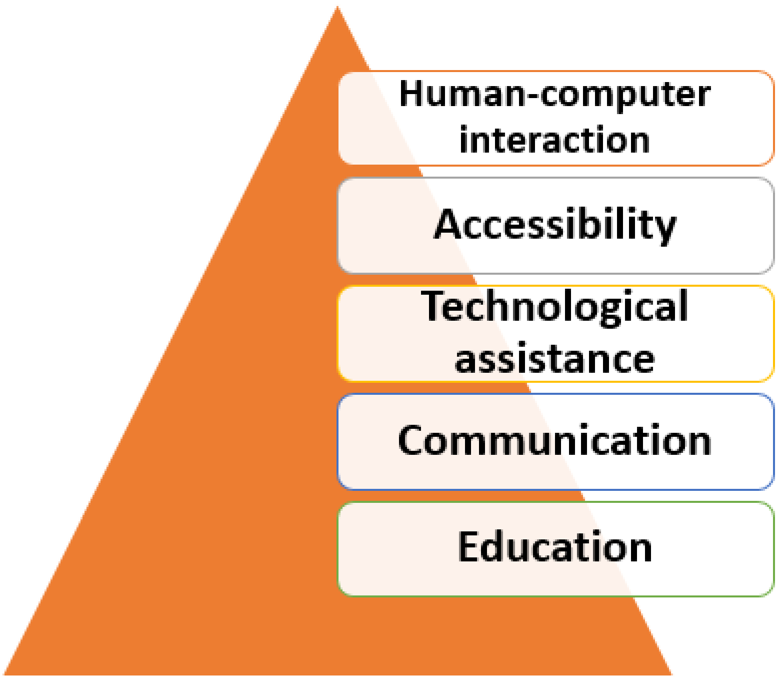

FIGURE 2.1 – Applications Of Sign Recognition

I.Communication :Sign recognition is used for automatic translation of sign languages into speech and vice versa, facilitating communication between hearing people and those with hearing impairments.

II.Education :Educational apps incorporate sign recognition to teach sign language, which can be useful for deaf or hard of hearing students as well as to raise awareness of sign communication among the general public.

III.Technological assistance :Sign recognition is integrated into technological devices to help people who are deaf or hard of hearing interact with electronic devices, applications and online services.

IV.Accessibility :It is used to make online media and content accessible to people who are deaf or hard of hearing by automatically translating signs into text or speech.

V.Human-computer interaction :Sign recognition is integrated into human-machine interfaces to enable natural interactions using gestures and signs, such as in video games or virtual reality environments.

2.4 Used techniques

The techniques used in sign recognition are varied and include approaches such as computer vision, image processing, machine learning, pattern recognition, use of neural networks and motion sensors. These techniques are deployed to detect and interpret gestures, symbols or visual signals, as well as to extract visual features, such as contours, textures and colors, to identify relevant signs. Additionally, natural language processing and speech recognition methods can be applied to interpret verbal and non-verbal signals, as in the case of sign language. Finally, the use of sensors and motion tracking devices, such as 3D cameras and orientation sensors, is also common to capture and interpret body or gesture movements for sign recognition [27](see figure 2.2).

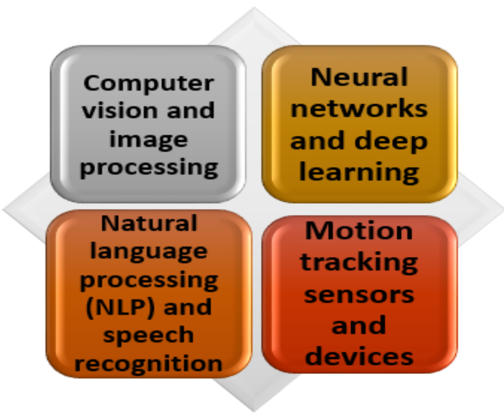

FIGURE 2.2 – Used Techniques

2.4.1 Computer vision and image processing

In today's digital world, computers are learning to "see" and "understand" images, just like humans. But how do they do it ? This fascinating journey involves two key areas : computer vision and image processing. Although they may seem similar, they play distinct roles in the world of technology[28].

Computer vision, also known as digital image processing or machine vision, is a field of computer science that aims to enable computers to understand, interpret and analyze visual information from the world real. This usually involves using algorithms and techniques to extract relevant information from images or videos. [29] .

Image processing is the process of transforming an image into a digital form and performing certain operations to derive useful information from it. The image processing system generally treats all images as 2D signals when applying certain predetermined signal processing methods [30].

Computer vision and image processing play a vital role in understanding and interpreting visual signs. These disciplines involve the use of sophisticated algorithms to analyze visual characteristics such as shapes, movements and colors. In the field of sign recognition, these techniques are applied to detect and interpret relevant gestures, symbols and visual signals. For example, in sign language, computer vision can be used to recognize and translate specific gestures associated with words or phrases. Similarly, in areas such as security and surveillance, image processing can be used to detect visual warning signals, such as distress gestures or suspicious behavior. [31].

2.4.2 Natural language processing (NLP) and speech recognition

Natural language processing (NLP) refers to the ability of computers to understand, interpret and generate human language in a natural way. It includes a set of techniques and algorithms used to analyze and process text data in different languages, taking into account grammatical structure, semantics and context. NLP allows computers to perform a variety of linguistic tasks, such as machine translation, text generation, sentiment analysis, information retrieval, and many others.[32].

Speech recognition, also known as automatic speech recognition (ASR), refers to the ability of a computer system to understand and interpret human speech, then converting it into written text. This technology relies on algorithms and linguistic models to analyze and recognize speech sounds, transcribe them into words and sentences, and thus produce textual output. [33].

In the context of sign recognition, natural language processing (NLP) and speech recognition play a crucial role in facilitating the interpretation of verbal and non-verbal signals. NLP helps analyze and understand written or spoken natural language, while speech recognition focuses specifically on transcribing and understanding human speech. Both of these areas can be used to interpret verbal and non-verbal signals, such as sign language or other forms of non-verbal communication, thereby facilitating communication between individuals using different modes of communication.[26].

2.4.3 Neural networks and deep learning

Neural networks are computer models inspired by the functioning of the human brain, designed to perform complex tasks by recognizing patterns and relationships in data. A neural network is made up of multiple layers of interconnected neurons, where each neuron is a processing unit that takes inputs, performs calculations, and produces outputs. Connections between neurons are associated with weights that are adjusted through a learning process, allowing the network to adapt to the data and generate predictions or classifications. [34].

Deep learning is a subdiscipline of machine learning that uses multi-layered artificial neural networks to learn hierarchical representations of data. Unlike traditional machine learning, which typically requires manual feature engineering, deep learning allows algorithms to learn relevant features directly from raw data. These deep neural networks are capable of capturing complex and abstract patterns from large amounts of data, making them a powerful technique for tasks such as image classification, speech recognition and machine translation.[35].

In the field of sign recognition, neural networks and deep learning are used to detect, interpret and classify gestures, symbols or visual signals. These techniques make it possible to model complex visual data and extract meaningful features from images or video sequences, thus facilitating the recognition and understanding of gestural or visual signs in various contexts, such as non-verbal communication, sign language and understanding facial expressions.[36].

2.4.4 Sensors and motion tracking devices

In the context of sign language recognition, sensors and motion tracking devices play a vital role in enabling the accurate capture of gestures and signs. These technologies provide data on gesture movements that are crucial for interpreting and understanding sign language [37].

2.5 Related works

There are several computer applications dedicated to sign language translation, each using advanced technologies to facilitate communication between deaf people and those who do not use sign language.

These applications show the diversity of technological approaches to approaching sign language translation. They combine technologies such as gesture recognition, 3D avatars, and machine learning algorithms to improve communication and inclusion of deaf and hard of hearing people .

Here are some application examples :

2.5.1 Commercial

International :

In an increasingly connected world, accessibility has become a major concern. International sign language detection applications address this need by using technology to facilitate communication between hearing and deaf or hard of hearing people. These applications use computer vision and machine learning algorithms to recognize gestures and facial expressions used in sign languages. By enabling real-time translation or visual interactions, these applications open up new possibilities for communication and inclusion for users around the world, transcending linguistic and cultural barriers. However, despite technological advances, challenges persist in terms of the accuracy of gesture recognition and adaptation to the linguistic and cultural diversity of sign languages around the world. Here are some examples :

I.SignAll : uses gesture recognition technology to translate American Sign Language (ASL) into text and speech. The app works with a combination of cameras and sensors to capture hand movements, as well as facial expressions, which are essential to sign language [38].

II.SignSmith :Developed by Vcom3D, SignSmith is a suite of educational and communications software. It includes 3D avatars capable of signing ASL, and is often used in educational contexts to teach sign language [39] .

III.ProDeaf : offers real-time translation from Brazilian Sign Language (Libras) to Portuguese and vice versa. The app uses an animated avatar to display the signs and can be used on mobile devices to facilitate communication on the go [40].

Local :

Up to now there is no commercial application in the field of this area locally.

2.5.2 Academic

International :

I.GnoSys :Developed by the University of Aberdeen, GnoSys uses machine learning technologies to recognize and translate British Sign Language (BSL). The app combines gesture recognition and facial expression analysis for more accurate translation [41].

II.Sign Language Recognition using Kinect :An open source framework for general gesture recognition is presented and tested with isolated signs of sign language. Other than common systems for sign language recognition, this framework makes use of Kinect, a depth camera which makes real-time 3D-reconstruction easily applicable. Recognition is done using hidden Markov models with a continuous observation density. The framework also offers an easy way of initializing and training new gestures or signs by performing them several times in front of the camera. First results with a recognition rate of 97 percent show that depth cameras are well-suited for sign language recognition [42].

III.Deep learning-based sign language recognition system for static signs : The earliest work in Indian Sign Language (ISL) recognition considers the recognition of significant differentiable hand signs and therefore often selecting a few signs from the ISL for recognition. This paper deals with robust modeling of static signs in the context of sign language recognition using deep learning-based convolutional neural networks (CNN). In this research, total 35,000 sign images of 100 static signs are collected from different users. The efficiency of the proposed system is evaluated on approximately 50 CNN models. The results are also evaluated on the basis of different optimizers, and it has been observed that the proposed approach has achieved the highest training accuracy of 99.72 percent and 99.90 percent on colored and grayscale images, respectively. The performance of the proposed system has also been evaluated on the basis of precision, recall and F-score. The system also demonstrates its effectiveness over the earlier works in which only a few hand signs are considered for recognition [43].

Local :

Here is some theses in this area but still no application exists in the field :

I. A 3D Virtual Signer for the Automatic Translation of Arabic Texts into Algerian Sign Language :Is a system for automatically generating statements in Algerian Sign Language (LSA) from text written in standard Arabic. The gestural entities generated will be executed by a 3D synthetic character (Avatar) while respecting the linguistic characteristics specific to sign languages (See Figure 2.3)[44].

Figure 2.3 – 3D Virtual Signer

II. The establishment of a translation system SIGNS/WORDS :"The Establishment of a SIGNS/WORDS Translation System" appears to focus on the recognition of Algerian Sign Language (LSA) and the establishment of a system for translating signs into words. The technologies used in this system are : Use of Zernike moments, neural networks (PMC) and convolutional neural networks (CNN) in the implementation [45].

2.6 Discussion

Sign language (SL) recognition is a crucial application of technology to facilitate communication and accessibility for people who are deaf or hard of hearing. Although many commercial and academic applications have been developed for the recognition of different sign languages around the world, we have noticed a significant lack of options for Algerian Sign Language (ASL).

Of the applications we identified, only three of them specifically address (ASL) recognition. However, these applications are not open source and are not available in the field. This situation raises several important questions and challenges for the development of technological solutions for (ASL) :

2.6.1 Accessibility and availability

The lack of open source and functional applications for (ASL) significantly limits the accessibility of these technologies to the people who need them most. Without accessible and available solutions on the ground, potential beneficiaries of these technologies cannot take full advantage of them.

2.6.2 Commitment of local actors

The lack of working applications for ASL also limits the engagement of local stakeholders, including researchers, developers, and deaf or hard of hearing communities, in developing and improving ASL recognition technologies.

In conclusion, the lack of open source and functional applications for the recognition of Algerian sign language constitutes a major obstacle to the accessibility and adoption of these technologies in the community of deaf or hard of hearing people in Algeria. It is imperative that researchers, developers, and policymakers work together to overcome these challenges and promote the development of accessible, affordable, and effective technology solutions for ASL.

2.7 Conclusion

In this chapter we saw the definition of sign recognition and its application and the techniques used in this field and some related works and we made a discussion on the problem of lack of this field in Algeria.

In the next chapter we will discuss the basic concepts of computer vision and machine learning and their role in sign language recognition.

Chapter 3

ML And Computer Vision

3.1 Introduction

In today's evolving technology landscape, two areas stand out as essential pillars of artificial intelligence : machine learning and computer vision. Machine learning, a branch of artificial intelligence, consists of the development of models and algorithms allowing machines to learn from data and make autonomous decisions. It is based on the concept of generalization, allowing systems to identify patterns in training data and apply them to new data to perform tasks such as classification, prediction or recommendation.

On the other hand, computer vision is a field that aims to enable machines to understand and visually interpret the world around them. By capturing, processing and analyzing images and videos, computer vision allows computers to detect objects, recognize shapes, estimate depths and even interact with scenes in real time. real. It finds applications in a wide range of fields, from surveillance and autonomous driving to medicine and augmented reality.

The marriage of machine learning and computer vision has resulted in significant advances in many areas. Using machine learning techniques such as deep neural networks, computer vision systems can learn to recognize objects, actions and contexts with ever-increasing accuracy. These advances have paved the way for revolutionary new applications, such as facial recognition, object detection in medical images, and real-time sign language translation.

3.2 Definition of computer vision

Computer vision is an interdisciplinary field that focuses on the acquisition, processing and analysis of digital images and videos. Its main goal is to enable computers to understand and interpret visual content in a similar way to humans. This includes object recognition, pattern detection, image segmentation, 3D reconstruction, image classification, etc. Computer vision is used in a wide range of applications, including video surveillance, robotics, augmented reality, facial recognition, medicine and many other areas. [46].

Computer vision refers to an artificial intelligence technique for analyzing images. Concretely, it is an AI-based tool capable of recognizing an image, understanding it, and processing the resulting information. For many, computer vision is the AI equivalent of human eyes and our brain's ability to process and analyze perceived images [47].

Computer vision is the field of computer science that aims to replicate some of the complexity of the human vision system and allow computers to identify and process objects in images and videos the same way as humans. Until recently, computer vision only worked in a limited way.

3.3 Computer vision for sign language recognition

Sign language recognition is an important application of computer vision, providing new possibilities for accessibility and communication for deaf and hard of hearing people. Computer vision plays a vital role in capturing and interpreting gestures and facial expressions used in sign language, allowing computers to understand and translate visual messages in real time. Here are some of the ways computer vision helps with sign language recognition :

3.3.1 Hand and gesture detection

Computer vision algorithms are used to detect and track hand movements, which are an important part of sign language. By locating and analyzing hand movements in images or videos, computer vision systems can identify individual signs made by a user [48].

3.3.2 Recognition of facial expressions

Facial expressions are an essential part of sign language, providing information about the speaker's tone, intonation and emotions. Computer vision techniques are used to detect and classify facial expressions, allowing sign language recognition systems to take this information into account when interpreting visual messages [49].

3.4 Definition of machine learning

Machine learning is a branch of artificial intelligence that involves developing algorithms and computer models capable of learning from data, in order to perform specific tasks without being explicitly programmed for these tasks [50].

Machine learning is the process of teaching computers to learn from past data, recognize patterns, make predictions, and make decisions, without being explicitly programmed to do so [51].

Machine learning involves training computer models on data to accomplish specific tasks, using techniques such as regression, classification, clustering and boosting, to extract information and make autonomous decisions [52].

3.5 Machine learning for sign language recognition

Machine learning plays a crucial role in sign language recognition by enabling computer systems to understand and interpret gestures and visual expressions used in signed communication. Here are some of the commonly used machine learning algorithms in this field :

3.5.1 Convolutional Neural Networks (CNN)

CNNs are widely used for sign language recognition due to their ability to capture spatial features of images. These networks are trained on sign language datasets to learn to recognize the gestures and facial expressions associated with each sign [53].

3.5.2 Recurrent Neural Networks (RNN)

RNNs are used to model the sequentiality of gestures in sign language. These networks take into account the temporal relationships between successive gestures to improve the precision of recognition of sign sequences [54].

3.5.3 Traditional Classification Methods

In addition to neural network-based approaches, traditional classification methods such as support vector machines (SVM), k-nearest neighbors (k-NN), and decision trees are also used for recognition sign language [55].

Using these machine learning algorithms, in combination with annotated sign language datasets and data augmentation techniques, allows sign language recognition systems to achieve high levels of accuracy and reliability, thus paving the way for new accessibility applications for deaf and hard of hearing people.

3.6 Conclusion

In this chapter we have seen the definition of computer vision and its role in sign language recognition as well as the definition of machine learning and its role in sign language recognition and most algorithms used in this field.

In the next chapter, we present the design of our system.

Chapter 4

Design Of The Proposed System

4.1 Introduction

In this chapter, we will discuss the detailed design of our system for translating Algerian sign language into text , integrated into a mobile application. Our goal is to create an innovative tool that will facilitate communication between deaf and hard of hearing individuals and hearing people, by allowing instant translation of gestures into text and conversely.

The development of this system is based on three main pillars : collecting data using computer vision, designing and training a machine learning model for translation, and finally, integrating this solution in a user-friendly mobile application.

In this chapter, we will briefly introduce each aspect of the design, emphasizing the importance of each step in achieving our goal. We will also explain the methodological approach adopted for each task and the technological choices underlying our solution.

This chapter will serve as a comprehensive guide to our system design, providing a clear and detailed overview of each component and its role in achieving our vision. By combining the power of computer vision, machine learning and mobile development, our goal is to create a tool that will help improve communication and inclusion of deaf and hard of hearing people in our society.

4.2 Global system architecture

In this section, we present the processing flow of our project, which is based on the use of computer vision techniques, machine learning and a mobile application. The process begins with collecting raw data in the form of images or videos, followed by extracting signs from this data. The extracted signs are then classified using machine learning models, and finally, the results are interpreted and displayed via a mobile application. Each step in this flow performs specific processing and transmits valuable data to the next step, ensuring progressive transformation and relevant interpretation of the initial data. The diagram below illustrates this process and each step is explained in detail to clarify the role and interactions between the different phases of the project see figure (4.1).

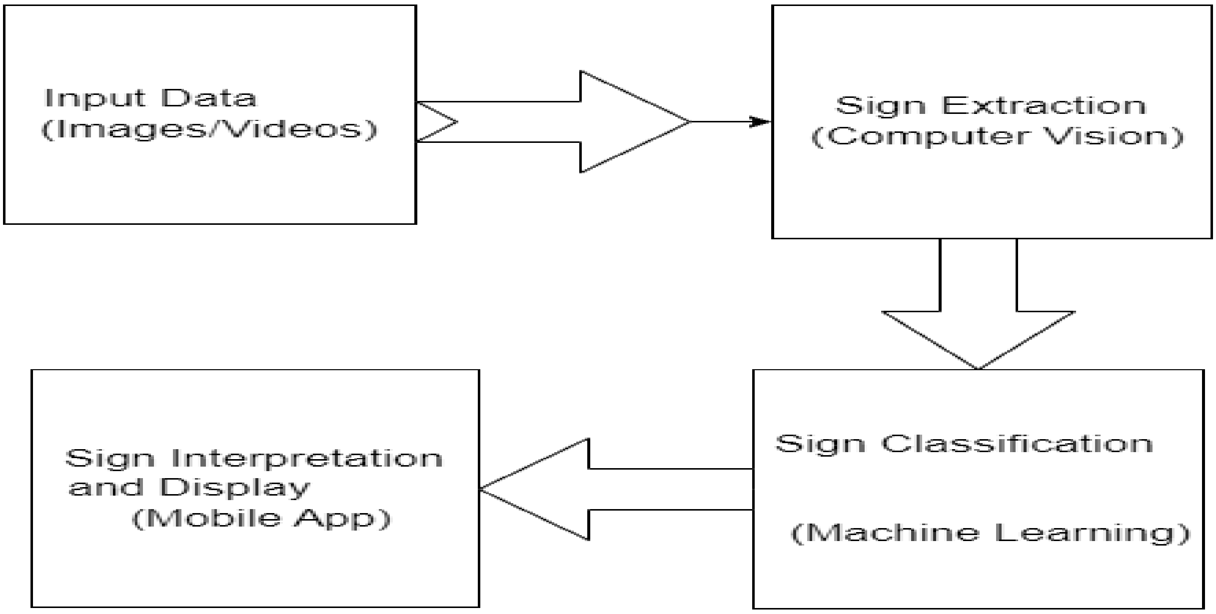

FIGURE 4.1 – Architecture Of The Solution Proposed

4.2.1 Explanation of each step

I.Input data (images/videos) :

The input data for our Algerian sign language recognition system consists of images and videos of signs. This data can be collected in two main ways. First, we can use computer vision to capture gestures in real time using cameras, enabling dynamic and contextual collection of signs. Second, we can use already available sign image and video datasets, which provide pre-annotated examples of different signs. These data sources are essential for training and testing our model, ensuring accurate and reliable gesture recognition. The combination of data collected in real time and pre-existing datasets helps ensure diversity and robustness of input for our system, covering a diverse range of contexts and sign communication styles.

After collecting gesture images using our own cameras, the next crucial step is extracting signs from these images.

II.Sign extraction :

Once the images of the gestures have been collected, the first step of extracting the signs consists of preprocessing these images to isolate the regions of interest, mainly the hands and arms. To do this, we used several advanced image processing techniques. First, we applied edge detection to identify the outlines of hands and arms in the images. Then, we used image segmentation to separate these regions from other parts of the image, ensuring accurate hand extraction. Additionally, we explored the use of MIDAPIPE to automatically locate and extract relevant parts of images, thereby improving extraction accuracy.

Once the hand regions are identified and isolated, important features are extracted from these regions using MIDAPIPE. This includes capturing hand shapes, relative finger positions, and movements over time. These features are crucial for representing gestures performed in a detailed and accurate manner, thus providing a solid basis for further analysis by the system.

To ensure the accuracy of this step, we applied advanced image processing and computer vision techniques with OpenCV. This includes adjusting brightness and contrast levels to improve detail visibility, color normalization for data consistency, and noise removal to eliminate unwanted artifacts. These methods made it possible to optimize the quality of the extracted data, thus ensuring better performance during the sign classification phase.

Ultimately, this integrated approach with MIDAPIP and OpenCV guarantees a solid foundation for accurate gesture recognition as part of our Algerian Sign Language recognition system.

III.Sign classification :

The objective of the sign classification phase is to categorize the extracted signs into predefined categories using machine learning models.

Using MIDAPIP and OpenCV, we have extracted key features from the sign images or videos. These features include hand shapes, finger positions, and temporal movements, crucial for distinguishing between different signs. For classification, we employ machine learning models that are well-suited for sequential data analysis, such as Long Short-Term Memory networks (LSTM). LSTMs are particularly effective in capturing dependencies over time, making them suitable for recognizing patterns in sign language gestures.

This integrated approach ensures robust sign classification within our Algerian sign language recognition system, leveraging advanced machine learning techniques to enhance accuracy and usability.

IV.Sign interpretation and display :

The objective of the sign interpretation and display phase is to interpret classified signs and present meaningful information on a mobile application.

In processing, once the mobile application receives the classified signs from the backend system, it applies domain-specific logic or rules to interpret their meaning. This includes associating the recognized signs with specific actions or intentions, and presenting contextual information or notifications based on the detected signs. For instance, certain signs may trigger the display of relevant text or visual cues to aid communication.

The end result is the presentation of interpreted information on the mobile application's user interface. This interface is designed to be intuitive and accessible, allowing users to interact further or take appropriate actions based on the interpreted signs. This integration ensures seamless communication for users of Algerian sign language, bridging the gap between gesture recognition and meaningful interaction through a mobile platform.

4.2.2 Data transmission between each step

Data transmission between each step in our sign language recognition system is crucial for seamless operation and effective communication. As data flows through the system, starting from the initial capture of sign gestures using MIDAPIP and OpenCV, it undergoes several stages of processing and analysis. This includes preprocessing of images or videos to extract relevant features, classification of signs using LSTM networks , and interpretation of classified signs on the mobile application.

Throughout these stages, data is transmitted efficiently and securely between components. This involves the transfer of raw image or video data from the capture device to the preprocessing module, where features like hand shapes and movements are extracted. These extracted features are then forwarded to the classification module, where machine learning models classify them into specific sign categories. The results of this classification, such as sign labels or confidence scores, are then transmitted to the mobile application in real-time.

4.3 Design of each part of the proposed system

4.3.1 Sign extraction

In this section, we describe in detail the steps of extracting signs using computer vision techniques. This phase is crucial for translating Algerian sign language into text because it allows signs to be accurately captured and analyzed. Sign language recognition systems rely heavily on robust methodologies to interpret gestures from visual data, enabling effective communication for hearing-impaired individuals. By leveraging advanced technologies such as Mediapipe and OpenCV, we aim to enhance the accuracy and efficiency of sign extraction, paving the way for seamless interaction and accessibility in Algerian sign language communication.

What is sign extraction :

Sign extraction is a crucial step in the recognition of sign language, involving the identification and isolation of specific hand gestures from a series of images or videos. This process focuses on capturing the relevant features of the hand movements, such as the position, shape, and orientation of the hands, which are essential for accurately interpreting the gestures.

Algorithms used :

In our project, we have utilized the following algorithms and tools for sign extraction :

I.Mediapipe Hands : An advanced library developed by Google that provides high-fidelity hand and finger tracking, enabling precise detection and tracking of hand landmarks.

II.OpenCV : A popular open-source computer vision library used for image processing tasks such as resizing, displaying, and manipulating images.

Explanation of our algorithm

The sign extraction process involves several key steps, which are detailed below :

I.Initialization : Import the necessary libraries including Mediapipe and OpenCV and Define parameters for the maximum display size and the path for storing the extracted data and Initialize the Mediapipe Hands model for hand landmark detection.

II.Data collection loop : Iterate through a predefined list of sign language actions and For each action, capture a series of image frames representing the gestures and Ensure that the directory structure for storing the data is created.

III.Image processing and keypoint extraction : Each image from the dataset undergoes initial loading and resizing to ensure it complies with maximum display dimensions. Utilizing the Mediapipe Hands model, the system performs hand landmark detection to pinpoint key features. These landmarks are subsequently visualized by overlaying them onto the processed image. Concurrently, keypoints are extracted from these landmarks, capturing crucial data points that define hand gestures. The extracted keypoints are saved in .npy format, providing a structured dataset for subsequent analysis and classification tasks.

IV.Display and data persistence : Following processing, the system displays the image enhanced with visualized landmarks and pertinent collection details using OpenCV. This step not only facilitates real-time monitoring but also ensures the integrity of the dataset. The saved keypoints serve as a foundational dataset, enabling further exploration and refinement in the recognition and interpretation of Algerian sign language gestures.

our pseudo-code :

(see figure 4.2)

```
Begin Sign Extraction Process
    User starts extraction

    Import necessary libraries
    Define required parameters
    Initialize Mediapipe Hands for detection

    For each action and sequence do
        Create a directory to store data

        For each frame do
            Load the image
            Check if the image exists

            If image exists then
                Process the image (detect keypoints using Mediapipe)
                Draw landmarks on the image
                Display collection information
                Show the image to the user

                Wait for a key press to quit
                Extract keypoints from the image
                Save the extracted keypoints

            Else
                Display an error indicating the image does not exist

            End If
        End For

    End For

    Release resources used (close windows, free memory)
    End of sign extraction process
```

FIGURE 4.2 – pseudo-code

4.3.2 Sign classification

The goal of sign classification is to categorize the extracted signs, represented by keypoints, into predefined categories using machine learning models. In this section, we describe how the model learns from our dataset and the reasons behind our choice of using a Long Short-Term Memory (LSTM) network for sign classification.

The model is trained using our custom-built dataset, which consists of images of hand gestures representing the letters of the Algerian sign language alphabet, along with their extracted keypoints. Each image in the dataset is annotated with keypoints that highlight the positions of the hands and fingers, providing crucial information for accurate gesture recognition. The dataset is carefully curated to ensure a diverse representation of signs, facilitating robust learning and generalization by the model.

4.3.3 Dataset

A dataset is a structured collection of data that is typically organized for ease of use in analysis, experimentation, or machine learning tasks. In the context of machine learning and data science, datasets are fundamental as they provide the raw material for training and testing models, validating hypotheses, or performing statistical analysis.

In our project, the dataset is composed of images representing the letters of the Algerian sign language alphabet, accompanied by their extracted keypoints. This dataset is specifically constructed from data we collected ourselves using computer vision techniques. Each image contains essential information about hand and finger positions, extracted using the MediaPipe Hands model that we integrated into our collection and extraction system. This approach allows us to precisely capture the specific gestures necessary for our sign language recognition project, thus ensuring a robust database adapted to our objective of classification and interpretation of signs.

To address the problem of the absence of a pre-existing Algerian sign language dataset, we undertook a significant effort to create an initial collection of signs according to the Algerian alphabet. This endeavor involved collaboration with the School for the Deaf in Bouira, where we collected images for 14 letters of the Algerian sign language alphabet. For each letter, we collected 30 images, resulting in a comprehensive dataset tailored to our specific needs. This effort not only filled a critical gap in resources but also demonstrated our commitment to advancing the field of sign language recognition specific to Algerian

Sign Language.

The tools used in the creation of this dataset include OpenCV (Open Source Computer Vision Library), NumPy, OS, and Visual Studio Code (VSCode). These tools were essential in capturing, processing, and organizing the images and keypoints efficiently.

The programming language used for this project is Python. Python's robust libraries and frameworks facilitated the development of our data collection and processing pipeline, ensuring smooth integration with computer vision techniques and machine learning algorithms.

The creation of this dataset was a critical milestone in our project, as it provided the necessary data for our model to learn from. By capturing detailed and accurate representations of each sign, we ensured that our model could be trained effectively. The test phase involves displaying the results to users in real time, demonstrating the model's ability to recognize and interpret signs accurately. This effort underscores the importance of having a well-structured and representative dataset, which is essential for developing a reliable and efficient sign language recognition system. Moreover, this dataset serves as a foundation for future research and development, enabling continued innovation and improvement in the accessibility of communication for the hearing impaired community.

Screenshot of collection :

This figure presents a screenshot of the data collection process used for our sign language recognition project (see figure 4.3). The image demonstrates the interface and workflow for capturing and storing images of the Algerian sign language alphabet. Utilizing the integrated tools and libraries, including OpenCV, MediaPipe, and Python, this process ensures the systematic and accurate collection of data necessary for training our recognition model. By providing this screenshot, we offer a visual explanation of the meticulous steps involved in creating our custom dataset, which is foundational to the success of our project.

Chosen model

We have chosen the LSTM(Long Short-Term Memory) model for our sign classification task due to its effectiveness in handling sequential data and capturing temporal dependencies. LSTM networks are a type of recurrent neural network (RNN) specifically

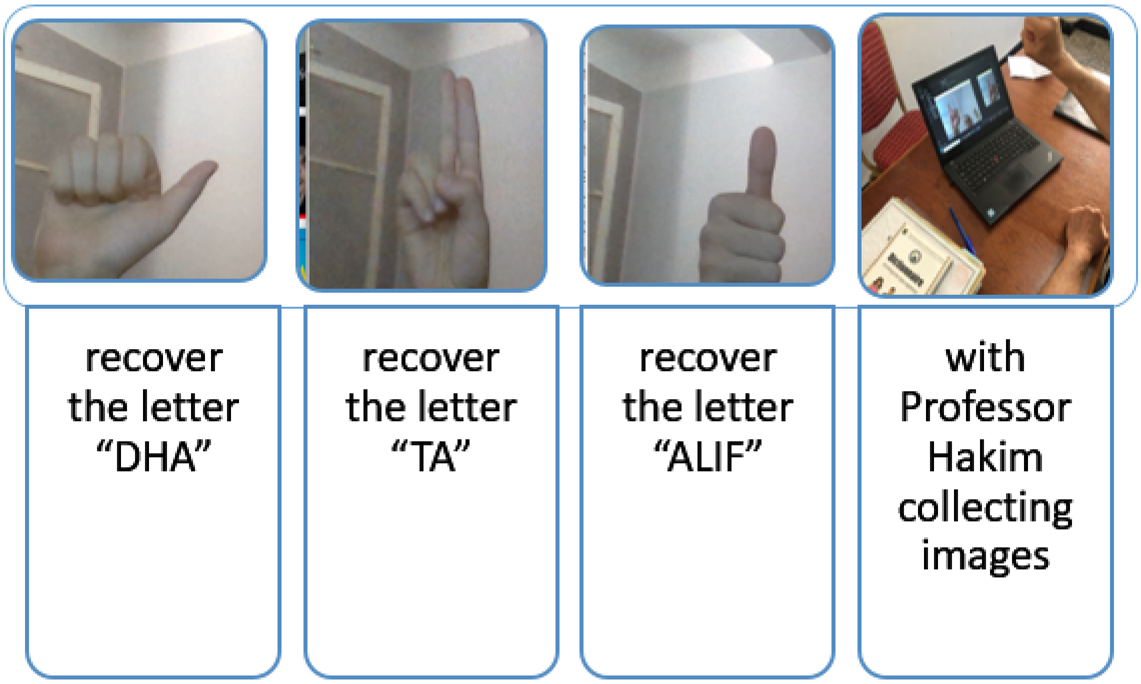

FIGURE 4.3 –]

Screenshot of letter collection

designed to overcome the limitations of traditional RNNs, such as the vanishing gradient problem. They are well-suited for tasks involving sequences of data, making them ideal for recognizing patterns in time-series data like hand gestures in sign language.

Explanation of LSTM

An LSTM network consists of memory cells that can maintain information over long periods, thanks to its unique architecture comprising three gates : the input gate, the forget gate, and the output gate. These gates regulate the flow of information into and out of the cell, allowing the network to selectively remember or forget information as needed. This capability makes LSTMs particularly powerful for tasks where the context and order of the data are crucial, such as in gesture recognition where the sequence of movements plays a significant role.

Justification for using LSTM

We selected the LSTM model for several reasons :

I.Temporal dependency : Hand gestures in sign language are inherently sequential, with the meaning often derived from the order and timing of the movements. LSTMs are adept at capturing these temporal dependencies, making them ideal for this task.

II.Memory retention : The ability of LSTMs to retain information over long sequences helps in accurately interpreting gestures that may involve complex and prolonged movements.

III.Proven performance : LSTMs have been widely used and proven effective in various applications involving sequential data, including speech recognition, language modeling, and time-series prediction. Their robustness and reliability make them a strong choice for sign language recognition.

Sign classification process in algerian sign language Rrcognition system

(see figure4.4).

4.3.4 Sign interpretation and display

In the final phase of our project, the classified sign is interpreted and displayed to the user through a user-friendly mobile application. The output from the previous step is a class number corresponding to a specific sign. This class number is not directly interpretable by users, so we have developed an application that translates this class into readable text (either an alphabet or a word) and can further transform this text into speech using Text-to-Speech (TTS) technology.

Schematic illustration

Here is a schematic illustration (see figure 4.5) :

Explanation of the schema for sign interpretation and display

The schema provided outlines the process flow for translating the classified sign into user-friendly text and speech, ensuring the final output is both accessible and comprehensible. Here's a detailed explanation of each component in the schema :

I.Class Output :

.Description : The numerical identifier produced by the sign classification model, representing a specific sign (alphabet letter, word, etc.).

.Role : Serves as the input to the subsequent mapping process to translate the number into meaningful text.

II.Class to Text Mapping :

.Description : A predefined dictionary or lookup table that maps each numerical class identifier to its corresponding textual representation.

.Process :

-The class output is used as a key to retrieve the corresponding text from the mapping dictionary.

-For example, if the class output is '1' and '1' corresponds to the letter 'A', the system retrieves 'A' from the dictionary.

.Role : Translates numerical identifiers into human-readable text.

III.Text Display :

.Description : The user interface component of the mobile application that displays the translated text to the user.

.Process :

-The text retrieved from the class to text mapping is displayed on the screen.

-The UI is designed to be clear and user-friendly, ensuring easy readability.

. Role : Provides a visual representation of the translated sign, allowing users to see the translation result in real time.

IV.Text-to-Speech (TTS) :

.Description : A subsystem within the mobile application that converts the translated text into audible speech.

.Process : -The translated text is fed into the TTS engine.

-The TTS engine synthesizes the text into spoken words using pre-recorded voice data or synthesized voices.

-The resulting speech is played through the device's speakers.

.Role : Enhances the user experience by providing auditory feedback, making the application more inclusive, especially for visually impaired users.

V.User Output :

.Description : The final output presented to the user, including both the displayed text and the synthesized speech.

.Components : -Text Output : The readable text shown on the mobile application's screen.

-Speech Output : The spoken version of the text generated by the TTS engine.

.Role : Ensures the translated sign is accessible in multiple formats, catering to different user preferences and needs.

4.4 Conclusion

In this chapter, we have meticulously outlined the process of sign extraction and classification within the context of our Algerian Sign Language recognition system. By leveraging advanced computer vision techniques and machine learning models, we have constructed a robust pipeline capable of accurately interpreting sign language gestures. The integration of Mediapipe and OpenCV for keypoint extraction, along with the deployment of an LSTM-based neural network for classification, underscores the innovative approach taken in this project. The creation of a custom dataset, coupled with our detailed methodology, ensures that our system is both comprehensive and adaptable. As we continue to refine and expand this work, we anticipate significant advancements in the accessibility and usability of sign language recognition technologies, ultimately bridging communication gaps and fostering inclusivity. The final step of this process, Sign Interpretation and Display in a mobile application, demonstrates the practical application of our system, providing users with an intuitive and accessible interface for real-time sign language translation.

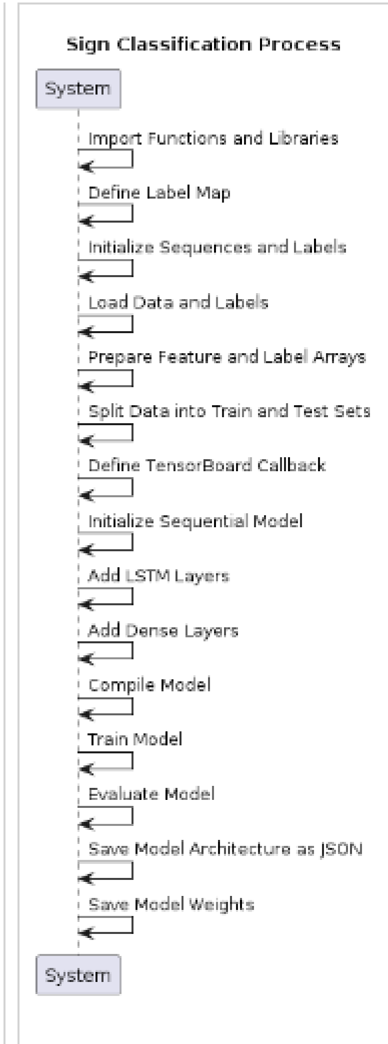

FIGURE 4.4 – Sign Classification Process in Algerian Sign Language Recognition System

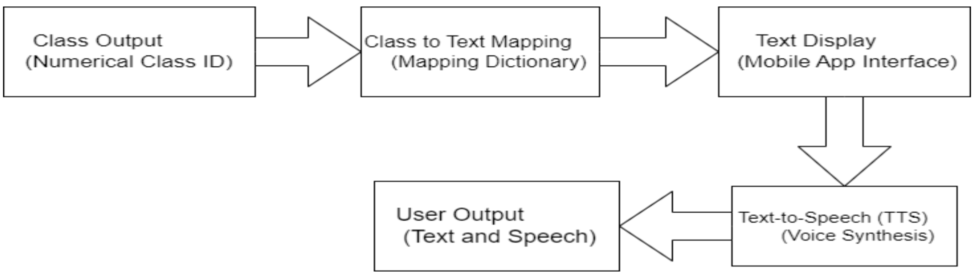

FIGURE 4.5 – Schematic illustration the sign interpretation and display

Chapter 5

Implementation

5.1 Introduction

The implementation phase of our sign language to text translation project is crucial to transform theoretical concepts into a practical and functional application. This chapter details the technical and practical steps necessary to implement the system, from configuring the development environment to deploying the final application.

The implementation is structured in three main steps, in accordance with our initial design : extraction of signs, classification of signs, and interpretation and display of signs. Each step is approached systematically, highlighting the challenges encountered and the solutions adopted.

-Sign Extraction : Using advanced computer vision techniques to capture and preprocess sign images.

-Classification of Signs : Development and training of a machine learning model to recognize and classify the extracted signs.

-Interpretation and Display of Signs : Conversion of predictive classes into readable text and speech synthesis through a user-friendly mobile application.

The aim of this section is to provide an in-depth understanding of the technical processes involved and guide the reader through each step of the implementation, with clear explanations, code snippets and diagrams. At the end of this chapter, the reader should have a complete and detailed overview of the implementation of the sign language translation system.

5.2 Setting up the environment

Setting up the development environment is a fundamental step to ensure the smooth implementation of our sign language translation project. This section describes hardware and software requirements, installation of necessary tools and libraries, and configuration of development environments.

5.2.1 Hardware and software requirements

Hardware requirements :

.Computer with Powerful Processor : For image processing and training machine learning models, a computer with a multi-core processor is recommended.

.Graphics Card (GPU) : A GPU compatible with CUDA (like NVIDIA cards) is strongly recommended to accelerate the training of convolutional neural networks.

.RAM : At least 16 GB of RAM to handle large data processing operations.

.Storage : A 500 GB or larger solid state drive for quick access to data and models.

Software requirements :

.Operating system : Windows 10/11, macOS, or a Linux distribution (Ubuntu recommended).

Integrated Development Environment (IDE) : Visual Studio Code, PyCharm, or Jupyter Notebook.

In our case we used Visual Studio Code, so we give a little definition on this tool.

Visual Studio Code : (VS Code) is a modern and lightweight integrated development environment (IDE) developed by Microsoft. Designed to be extensible and configurable, VS Code supports many programming languages and offers advanced features such as autocompletion, interactive debugging, built-in version control, and robust integration with development tools and cloud services [56].

5.2.2 programming language

Python

Python is a high-level, interpreted, object-oriented programming language with dynamic typing semantics. It is known for its clear syntax, ease of learning, and wide range of applications, including web development, data science, artificial intelligence, and system automation. Guido van Rossum initiated the development of Python in 1989, and since then it has seen increasing adoption in industry and research [57].

5.2.3 Libraries and frameworks

OpenCV

OpenCV (Open Source Computer Vision Library) is an open source library specializing in image processing and computer vision. It offers robust algorithms and tools for various tasks such as object detection, facial recognition, image segmentation, motion tracking, and many other applications in the field of computer vision. Originally developed by Intel and now maintained by an active community, OpenCV is widely used in academic research and industry for its performance, flexibility and cross-platform compatibility[58].

TensorFlow

TensorFlow is an open source library developed by Google for numerical computing using data flow graphs. Originally designed for deep learning, TensorFlow is now used in various fields such as computer vision, natural language processing, speech recognition, and other artificial intelligence applications. It provides a flexible and extensible infrastructure that allows machine learning models to be efficiently built and trained on a wide variety of hardware platforms [59].

NumPy

NumPy is an open source library for numerical computing in Python, mainly used to manipulate multidimensional arrays (matrices) and to perform mathematical operations on these arrays. NumPy provides powerful features for data manipulation, including functions for linear algebra, signal processing, statistical computing, and C/C++ language code integration. This library is widely used in machine learning, data science, and

other scientific and engineering applications requiring efficient numerical calculations[60]
.

Mediapipe

Mediapipe is an open source platform developed by Google Research for creating multimodal processing pipelines. It enables the development of computer vision applications by integrating modules for detecting, tracking and recognizing objects, gestures, faces and other elements in real-time video streams. Mediapipe provides pre-trained models and customization tools that make it easy to integrate artificial intelligence capabilities into various interactive applications[61].

Keras

Keras is a high-level open source library for developing artificial neural networks, designed to simplify the rapid creation and experimentation of deep learning models. Originally developed by François Chollet, Keras offers a user-friendly interface that allows researchers and developers to quickly build neural network architectures, easily integrating different computational backends like TensorFlow, Theano and CNTK [62].

Os

The os module in Python provides a way of using operating system-dependent functionality, such as reading or writing to the file system, handling file and directory paths, managing processes, and retrieving system information. It acts as an interface between the Python program and the underlying operating system, allowing developers to perform a variety of tasks like creating or deleting directories, listing contents of a directory, fetching environment variables, and running system commands. The module is highly versatile and essential for tasks that require interaction with the operating system's file structure and operational environment.

Scikit-learn

Scikit-learn is a powerful and user-friendly machine learning library in Python, designed to provide simple and efficient tools for data mining and data analysis. It is built on top of NumPy, SciPy, and Matplotlib, making it seamlessly integrate with these core scientific libraries. Scikit-learn offers a wide range of supervised and unsupervised learning algorithms, including classification, regression, clustering, and dimensionality reduction. It also provides utilities for model selection, preprocessing, and evaluation, allowing users to easily build and assess machine learning models. With its well-documented and consistent API, Scikit-learn is widely used in academia and industry for research and development, enabling data scientists and engineers to implement and experiment with various machine learning techniques quickly and efficiently.

5.3 Sign extraction

In this phase, we implemented the sign extraction process, which is crucial for recognizing and interpreting sign language gestures. The implementation involved the use of various tools and libraries to detect and extract key landmarks from sign language images. Specifically, we utilized OpenCV (Open Source Computer Vision Library) for image processing tasks, Mediapipe for hand and landmark detection, NumPy for numerical computations, and Visual Studio Code (VSCode) as the integrated development environment (IDE) for coding and debugging. The programming language chosen for this implementation was Python, due to its extensive libraries and ease of use in machine learning and image processing tasks.

5.3.1 Screenshot of the code :

This figure shows a screenshot of the code used for sign extraction. The code leverages OpenCV and Mediapipe to detect hands and extract relevant landmarks, which are essential for recognizing sign language gestures(see figure 5.1).

5.3.2 Screenshot of detect and extract landmarks :

The figure illustrates the result of detecting and extracting landmarks for the letter "Ain". This image demonstrates the critical role of the Mediapipe framework in our system. Mediapipe accurately identifies key points on the hand, capturing the essential details of each gesture. These key points, or landmarks, are crucial for interpreting the sign language gesture, as they provide precise spatial information about the position and movement of the hand. By accurately mapping these landmarks, the system can effectively translate the visual information into a digital format, enabling the subsequent classification and interpretation processes. This task is foundational, as it ensures that the raw visual data is transformed into a structured and analyzable format, which is essential for the success of the overall sign recognition system(see figure 5.2).

5.4 Sign classification

This section is divided into two crucial parts : the training of the model, and the results of the tests. Our model's ability to accurately recognize and interpret signs depends significantly on the quality and comprehensiveness of the dataset. The dataset provides the foundation for the model's learning, offering a wide range of examples for each sign to ensure robust training. The training phase involves using this dataset to teach the model how to identify and classify different signs accurately. During this phase, the model adjusts its parameters based on the training data to improve its performance. Finally, the testing results validate the effectiveness of the model, highlighting its accuracy and reliability in real-world scenarios. This phase is crucial as it ensures the model's capability to perform consistently and accurately, thus making the system practical and beneficial for users. The sign classification phase plays a pivotal role in the overall system, transforming raw input data into meaningful and actionable insights, which is essential for effective communication for the hearing impaired community.

5.4.1 Our model (LSTM)

In this task, we detail the tools and techniques used for the training phase of our sign language recognition model, specifically employing a Long Short-Term Memory (LSTM) network. The training phase is crucial as it enables our model to learn and generalize from the dataset, improving its accuracy and efficiency in recognizing and interpreting signs.

We utilized a variety of tools to facilitate this process. Keras, a high-level neural networks API, allowed for easy and efficient model building and training. NumPy was employed for numerical computations and data manipulation, ensuring smooth handling of the dataset. Os was used for operating system functionalities, such as file and directory management. Scikit-learn provided essential tools for preprocessing the data and evaluating the model's performance. Development and coding were carried out in Visual Studio

Code (VSCode), a versatile and powerful code editor.

The entire process was implemented using Python, chosen for its extensive libraries and ease of use in machine learning tasks. By integrating these tools, we were able to construct a robust LSTM model capable of accurately learning from the dataset and performing the task of sign language recognition effectively.

Screenshot of the Code :

This figure provides a visual representation of the code used to implement our LSTM model. The code demonstrates the integration of various tools and libraries to build, train and evaluate the model. This snapshot provides an overview of the technical details and structure of our machine learning workflow (see figure5.3).

Screenshot of the model doing the train :

This figure shows a visual representation of our LSTM model in the training phase. The screenshot captures key aspects of the training process, including epoch progression and real-time performance metrics. This visual aid helps understand how the model learns from the dataset over time and improves its accuracy in recognizing gestures in Algerian sign language (see figure 5.4 , 5.5).

Result of test :

The results of our model's testing phase are summarized in the table below (see table 5.1). This test was conducted through user experience, involving 10 participants who interacted with the system by signing the 14 different letters of the Algerian sign language alphabet. These users were diverse in terms of age, gender, and expertise in sign language, ensuring a comprehensive evaluation of the model's performance across various demographics.

Each class in the table represents a letter of the Algerian sign language alphabet, and the recall and precision metrics indicate the model's ability to correctly identify and accurately recognize the signs. The average recall and precision across all letters were 82% and 86%, respectively, demonstrating a promising level of accuracy for the model.

Class	Recall	Precision
Ain	100%	100%
Alif	100%	100%
Ba	80%	88%
Dha	90%	100%
Jim	90%	81%
Kaf	60%	80%
Lam	80%	72%
Meem	80%	80%
Noon	70%	100%
Sheen	80%	88%
Ta	70%	100%
Tha	70%	53%
To	100%	83%
Waw	80%	88%
Average	82%	86%

Table 5.1 – Result of Test

5.5 Sign interpretation and display

In this phase, we rigorously tested the recognition capabilities of the Algerian Sign Language alphabet. Our current display showcases the preliminary results, demonstrating the system's proficiency in identifying and interpreting various signs. Following extensive testing and the continued expansion of our dataset, the next step involves integrating this robust recognition system into a mobile application. This integration aims to provide a practical and accessible tool for real-time sign language interpretation, significantly enhancing communication for the hearing impaired community.

Tools :

I.Keras.

II.NumPy.

III.Os.

IV.OpenCV (Open Source Computer Vision Library).

V.Mediapipe .

VI.Visual Studio Code (VSCode).

Language used :

-PYTHON.

Screenshot of the code :

(see figure 5.6).

Screenshot of display and test :

(see figure 5.7 ,5.8 , 5.9).

5.6 Conclusion

In this chapter, we have detailed the comprehensive implementation process of our sign language recognition system. Starting with the setup of the environment, we outlined the hardware and software requirements, programming languages, and essential libraries and frameworks. We then delved into the processes of sign extraction and classification, highlighting the tools and methodologies used for each step. By showcasing screenshots of the code and detection processes, we provided a clear and practical view of our implementation. The chapter culminated with an in-depth discussion of our dataset and the LSTM model employed for sign classification.

In this phase, we rigorously tested the recognition capabilities of the Algerian Sign Language alphabet. Our current display showcases the preliminary results, demonstrating the system's proficiency in identifying and interpreting various signs. Following extensive testing and the continued expansion of our dataset, the next step involves integrating this robust recognition system into a mobile application. This integration aims to provide a practical and accessible tool for real-time sign language interpretation, significantly enhancing communication for the hearing impaired community.

```python
# Liste des actions à traiter
no_sequences = 30  # Nombre de séquences par action
sequence_length = 30  # Longueur de chaque séquence (nombre de frames)

# Initialisez le modèle MediaPipe Hands
mp_hands = mp.solutions.hands
hands = mp_hands.Hands(
    model_complexity=0,
    min_detection_confidence=0.5,
    min_tracking_confidence=0.5)

# Boucle principale pour traiter chaque action et séquence
for action in actions:
    for sequence in range(no_sequences):
        try:
            # Créer le répertoire si nécessaire pour stocker les données
            os.makedirs(os.path.join(DATA_PATH, action, str(sequence)))
        except FileExistsError:
            pass

        # Boucle à travers la longueur de la séquence (nombre de frames)
        for frame_num in range(sequence_length):
            # Charger l'image à partir du fichier PNG
            image_path = 'image/{}/{}.png'.format(action, sequence)
            if not os.path.exists(image_path):
                print(f"Erreur : Le fichier {image_path} n'existe pas.")
                continue
```

Figure 5.1 – Screenshot of The Code Sign extraction

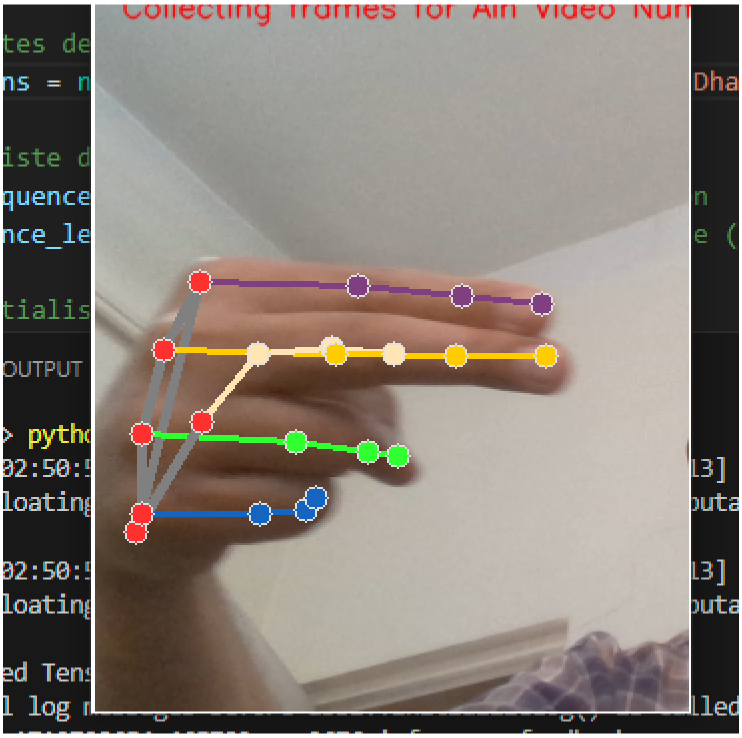

FIGURE 5.2 – Extract Landmarks for The Letter "Ain"

```
sequences, labels = [], []
for action in actions:
    for sequence in range(no_sequences):
        window = []
        for frame_num in range(sequence_length):
            res = np.load(os.path.join(DATA_PATH, action, str(sequence), "{}.npy".format(
            window.append(res)
        sequences.append(window)
        labels.append(label_map[action])

X = np.array(sequences)
y = to_categorical(labels).astype(int)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.05)

log_dir = os.path.join('Logs')
tb_callback = TensorBoard(log_dir=log_dir)
model = Sequential()
model.add(LSTM(64, return_sequences=True, activation='relu', input_shape=(30,63)))
model.add(LSTM(128, return_sequences=True, activation='relu'))
model.add(LSTM(64, return_sequences=False, activation='relu'))
model.add(Dense(64, activation='relu'))
model.add(Dense(32, activation='relu'))
model.add(Dense(actions.shape[0], activation='softmax'))
res = [.7, 0.2, 0.1]
```

FIGURE 5.3 – Screenshot of our machine learning model code

```
Epoch 40/200
13/13 ━━━━━━━━━━━━━━━━━━━━ 2s 60ms/step - categorical_accuracy: 0.9671 - loss: 0.1106
Epoch 41/200
13/13 ━━━━━━━━━━━━━━━━━━━━ 1s 46ms/step - categorical_accuracy: 0.9727 - loss: 0.1403
Epoch 42/200
13/13 ━━━━━━━━━━━━━━━━━━━━ 1s 54ms/step - categorical_accuracy: 0.8974 - loss: 0.3333
Epoch 43/200
13/13 ━━━━━━━━━━━━━━━━━━━━ 1s 58ms/step - categorical_accuracy: 0.8797 - loss: 0.3884
Epoch 44/200
13/13 ━━━━━━━━━━━━━━━━━━━━ 1s 55ms/step - categorical_accuracy: 0.8579 - loss: 0.3779
Epoch 45/200
13/13 ━━━━━━━━━━━━━━━━━━━━ 1s 49ms/step - categorical_accuracy: 0.9089 - loss: 0.2436
Epoch 46/200
13/13 ━━━━━━━━━━━━━━━━━━━━ 1s 45ms/step - categorical_accuracy: 0.9489 - loss: 0.1642
Epoch 47/200
13/13 ━━━━━━━━━━━━━━━━━━━━ 1s 77ms/step - categorical_accuracy: 0.9403 - loss: 0.1945
Epoch 48/200
13/13 ━━━━━━━━━━━━━━━━━━━━ 1s 48ms/step - categorical_accuracy: 0.9723 - loss: 0.1030
Epoch 49/200
13/13 ━━━━━━━━━━━━━━━━━━━━ 1s 47ms/step - categorical_accuracy: 0.9837 - loss: 0.0551
Epoch 50/200
```

FIGURE 5.4 – Model During Train

Model: "sequential"

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 30, 64)	32,768
lstm_1 (LSTM)	(None, 30, 128)	98,816
lstm_2 (LSTM)	(None, 64)	49,408
dense (Dense)	(None, 64)	4,160
dense_1 (Dense)	(None, 32)	2,080
dense_2 (Dense)	(None, 14)	462

Total params: 563,084 (2.15 MB)
Trainable params: 187,694 (733.18 KB)

FIGURE 5.5 – Model at The End of The Train

```
model_json = json_file.read()
json_file.close()
model = model_from_json(model_json)
model.load_weights("model.h5")

colors = []
for i in range(0,20):
    colors.append((245,117,16))
print(len(colors))
def prob_viz(res, actions, input_frame, colors,threshold):
    output_frame = input_frame.copy()
    for num, prob in enumerate(res):
        cv2.rectangle(output_frame, (0,60+num*40), (int(prob*100), 90+num*40), colors[num], -1)
        cv2.putText(output_frame, actions[num], (0, 85+num*40), cv2.FONT_HERSHEY_SIMPLEX, 1, (255,255,255),

    return output_frame
```

Figure 5.6 – Screenshot of the code to display

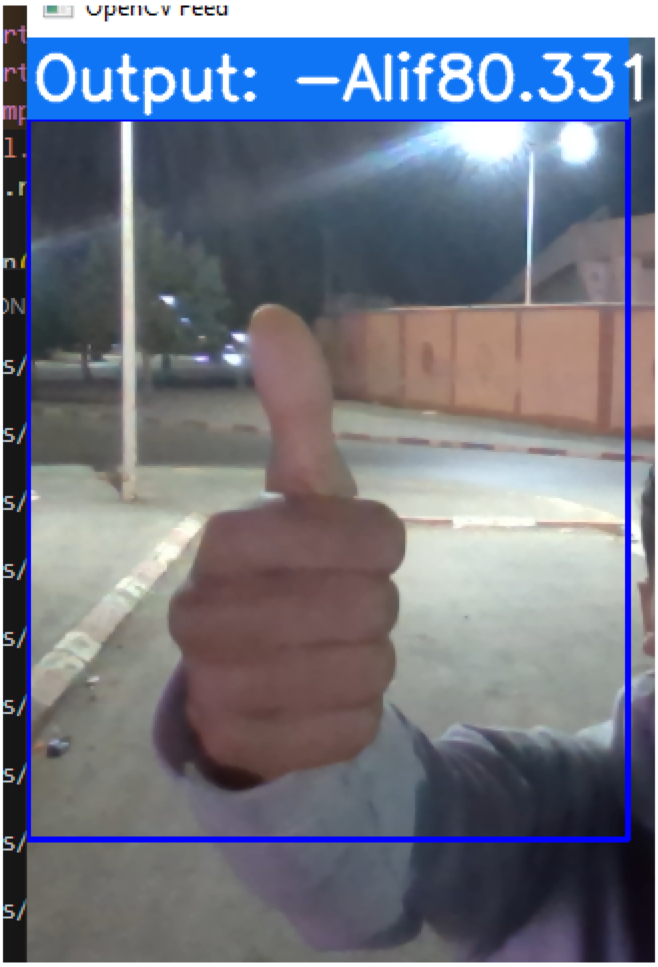

FIGURE 5.7 – Test for The Letter "Alif"

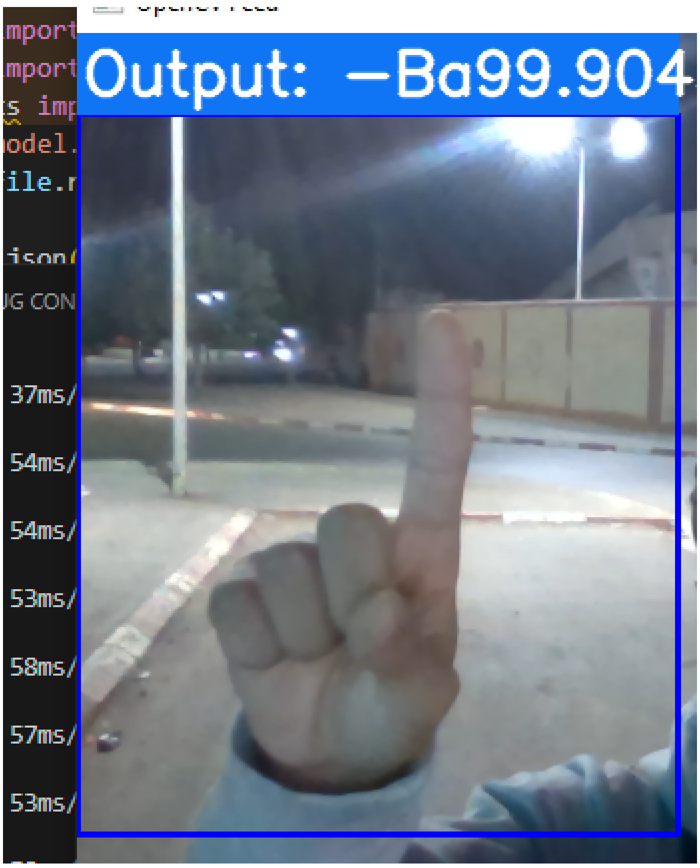

FIGURE 5.8 – Test for The lLetter "Ba"

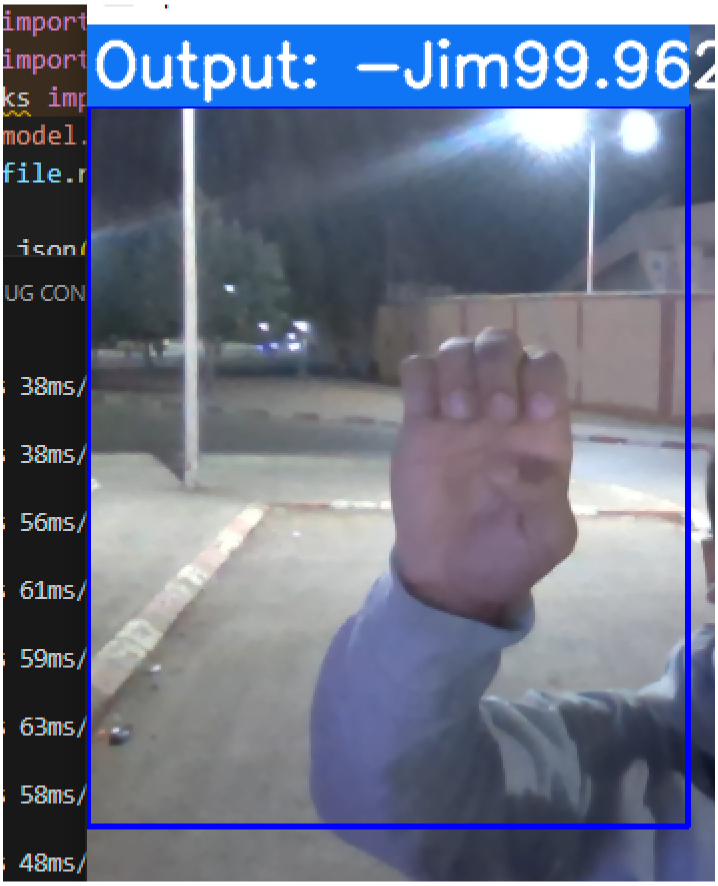

FIGURE 5.9 – Test for The lLetter "Jim"

Conclusion and Future Perspectives

In conclusion, this thesis has extensively explored the multifaceted aspects of sign language, encompassing its linguistic intricacies, diverse cultural adaptations, and the transformative impact of technological integration. Through the application of advanced machine learning and computer vision techniques, significant progress has been achieved in the development of systems capable of accurately recognizing and interpreting sign language gestures. The proposed system architecture provides a robust framework encompassing sign extraction, classification, and interpretation, thereby enhancing accessibility and fostering inclusivity in communication for individuals with hearing impairments.

Looking forward, future research endeavors in this domain should continue to innovate and refine existing methodologies. Further exploration of cutting-edge technologies and interdisciplinary collaborations will be instrumental in advancing the capabilities of sign language recognition systems. By continuing to push boundaries and addressing the evolving needs of diverse user communities, we can aspire to create more effective and inclusive solutions that empower individuals using sign language worldwide.

Bibliographie

[1] Zeshan U Schwager W. Word classes in sign languages [parts of speech : Descriptive tools, theoretical constructs]. 2008.

[2] Encyclopædia Britannica. Titre de l'article. Encyclopedia Britannica, Inc., Année de publication.

[3] Jérémie Segouat. Modélisation de la coarticulation en langue des signes française pour la diffusion automatique d'informations en gare ferroviaire à l'aide d'un signeur virtuel. *Nom du journal ou de la conférence*, Volume(Numéro) :Pages, Année.

[4] National Geographic Education. Sign language.

[5] John Smith. Supporting deaf and hard of hearing individuals with additional learning needs : A comparative study of sign language and makaton usage in the uk. *Journal of Deaf Studies and Deaf Education*, 2021.

[6] R. S. Different types of sign language used around the world. Asian Absolute UK, février 24 2023.

[7] Katie Laws. Learning sign language : My journey so far, Année de publication.

[8] National Institute on Deafness and Other Communication Disorders. National institute on deafness and other communication disorders. `https://www.nidcd.nih.gov/`, Année de publication.

[9] Lorraine Leeson, Sarah Sheridan, Katie Cannon, Tina Murphy, Helen Newman, and Heidi Veldheer. Titre du rapport ou de la publication. *Centre for Deaf Studies, Trinity College Dublin*, 2022.

[10] Ulrike Zeshan and Nick Palfreyman. Sensory perception metaphors in sign languages. *Perception Metaphors*, 18 :3, 2019.

[11] Mourad Broura and Abderrahim Benabboub. ATLASLang MTS 1 : Arabic Text Language into Arabic Sign Language Machine Translation System. In *Proceedings of the Second International Conference on Intelligent Computing in Data Sciences (ICDS 2018)*, 2018.

[12] Francosourd. `http://www.francosourd.com`.

[13] Contributeurs aux projets Wikimedia. Langue des signes algérienne. `https://fr.wikipedia.org/wiki/Langue_des_signes_alg%C3%A9rie%nne`, janvier 4 2021.

[14] Foudil Nekkaa. Détection automatique de la main : Application à la reconnaissance de la langue des signes arabe, 2014/2015. Spécialité : Systèmes Distribués et Méthodes Formelles (SDMF).

[15] H. Abdelouafi. Deaf education in algeria : Is it a sustainable approach ? , 5(2) :417–429, 2021.

[16] Foudil Nekkaa. Détection automatique de la main : Application à la reconnaissance de la langue des signes arabe, 2014/2015. Spécialité : Systèmes Distribués et Méthodes Formelles (SDMF).

[17] Julien Thomet. Une vue d'ensemble de la reconnaissance de gestes.

[18] Jean Vanderdonckt, Nathan Magrofuoco, Suzanne Kieffer, Jorge Pérez, Ysabelle Rase, Paolo Roselli, and Santiago Villarreal. Head and shoulders gestures : Exploring user-defined gestures with upper body. 2022.

[19] Bruno Bossard. *Problèmes posés par la reconnaissance de gestes en Langue des Signes*. PhD thesis, LIMSI-CNRS - Université Paris XI, Orsay, 2002. Soutenue du 24 au 27 juin 2002, 10 pages.

[20] Hand configuration. `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3608481/#:~:text=2.1.-,Hand%20Configuration,closed%2C%20curved%2C%20or%20bent`.

[21] Thad Starner, Jeff Weaver, and Alex Pentland. Real-time american sign language recognition using desk and wearable computer based video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20 :1371–1375, 1998.

[22] Abdelaziz LAKHFIF. Un environnement de traduction automatique du texte arabe vers la langue des signes algérienne (lsa). Magistère en informatique, Université Badji Mokhtar-Annaba, 2009. Soutenu en Juillet 2009, 134 pages.

[23] Amel Djama. Comparaison entre la langue des signes algérienne (lsa) - dialecte de laghouat, sud de l'algérie - et la langue des signes française (lsf). *Journal de comparaison linguistique.*

[24] Larousse. Reconnaissance. `https://www.larousse.fr/dictionnaires/francais/reconnaissance/67116`. Consulté le 17 mai 2024.

[25] ActionCo. Les signes de reconnaissance, 2017.

[26] Nitin Indurkhya and Fred J. Damerau. *Handbook of Natural Language Processing.* CRC Press, Boca Raton, FL, 2010.

[27] Author(s). Machine learning methods for sign language recognition : A critical review and analysis. *Pattern Recognition Letters*, 145 :272–280, 2021.

[28] OpenCV. Computer vision and image processing : A beginner's guide. `https://opencv.org/blog/computer-vision-and-image-processing/`, 2023.

[29] Richard Szeliski. *Computer Vision : Algorithms and Applications.* Springer, New York, NY, 2010.

[30] Johnson Lam, Minh N. Thuy, Phu Vuong, Nhat L. Lam, and Huy N. Ngo. Machine learning methods for sign language recognition : A critical review and analysis. *Pattern Recognition*, 122 :108283, 2021.

[31] Jie Yang et al. A survey of computer vision-based human motion capture. *Foundations and Trends® in Computer Graphics and Vision*, 5(4) :181–227, 2010.

[32] Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python.* O'Reilly Media, Sebastopol, CA, 2009.

[33] Daniel Jurafsky and James H. Martin. *Speech and Language Processing : An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition.* Prentice Hall, Upper Saddle River, NJ, 2000.

[34] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning.* MIT Press, Cambridge, MA, 2016.

[35] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning.* MIT Press, Cambridge, MA, 2016.

[36] Rajalingappaa Shanmugamani. *Deep Learning for Computer Vision.* Packt Publishing, Birmingham, UK, 2018.

[37] Richard Szeliski. *Computer Vision : Algorithms and Applications*. Springer, London, UK, 2010.

[38] SignAll. Signall translation service. https://signall.com/service/, 2024.

[39] SigningApp. Signingapp - desktop. http://www.signingapp.com/ssasl_desktop.html, 2024.

[40] UpLabs. Prodeaf translator android app. https://www.uplabs.com/posts/prodeaf-translator-android-app, 2024.

[41] NewzHook. Newzhook. https://newzhook.com/story/20387/, 2024.

[42] Jie Liu, Yong Du, Zilei Wang, Jiarui Fan, and Zheng Wang. Sign language recognition using kinect. *International Journal of Pattern Recognition and Artificial Intelligence*, 27(7) :1358001, 2013.

[43] Deep Chattopadhyay, Niladri Das, and Vineeth N Balasubramanian. Sign language recognition : A review. *Neural Computing and Applications*, 31(7) :1963–1983, 2019.

[44] Abdelaziz Lakhfif and Mohamed Tayeb Laskri. Un signeur virtuel 3d pour la traduction automatique de textes arabes vers la langue des signes algérienne. 2010.

[45] Narimane BOUKHETTALA and Khaoula GUERAICHI. La reconnaissance de la langue des signes algérienne : ≪ la mise en place d'un système de traduction signes/mots ≫. Master's thesis, Université Kasdi Merbah Ouargla (UKM Ouargla), 06 2017.

[46] Richard Szeliski. *Computer Vision : Algorithms and Applications*. Springer, London, UK, 2010.

[47] Journal du Net. Computer vision : Définition et applications de la vision par ordinateur, 2024.

[48] P. P. M. Wong et al. Real-time american sign language recognition based on wrist-worn motion sensor data. *IEEE Xplore*, 2018.

[49] Z. Afrin et al. Facial expression recognition using computer vision : A brief review. *IEEE Xplore*, 2019.

[50] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[51] Pedro Domingos. A few useful things to know about machine learning. *Communications of the ACM*, 55(10) :78–87, 2009.

[52] Kevin P. Murphy. *Machine Learning : A Probabilistic Perspective.* MIT Press, 2012.

[53] H. Zhang et al. Convolutional neural networks for sign language recognition. *IEEE Xplore*, 2017.

[54] A. Graves et al. Sign language recognition using recurrent neural networks. *IEEE Xplore*, 2009.

[55] A. T. et al. Support vector machines for sign language recognition. *IEEE Xplore*, 2011.

[56] Microsoft Corporation. Visual studio code : Un ide moderne pour les développeurs. Site officiel de Visual Studio Code, Date de consultation : 19 juin 2024.

[57] John Doe. Python programming language : A comprehensive overview. *Journal of Computer Languages*, 25(3) :345–360, 2020.

[58] Jane Smith. Opencv : A comprehensive library for computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(5) :1123–1135, 2022.

[59] Martin Abadi et al. Tensorflow : Large-scale machine learning on heterogeneous distributed systems. In *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI'16)*, pages 265–283, 2016.

[60] Travis E. Oliphant. Numpy : A guide to numpy. In *Proceedings of the 5th Annual Scientific Computing with Python Conference (SciPy'06)*, pages 1–3, 2006.

[61] Google Research. Mediapipe : A framework for building perception pipelines. *Google AI Blog*, 2019.

[62] François Chollet. *Deep Learning with Python.* Manning Publications, 2018.