People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

Akli Mohand Oulhadj University – Bouira

Faculty of Exact Sciences

Department of Computer Science

Master's Thesis

in Computer Science

Specialty: *ISIL*

Theme

Building a 3D Avatar-Based Dictionary of Arabic-Algerian Sign Language: Text to Sign Translation System

Proposed and directed by:

- Dr. Taha Zerrouki

Prepared by:

- Amara Ouail Said
- Kherfi Lyes

Academic Year 2025 – 2026

Acknowledgements

First and foremost, we thank Allah for granting us the strength, patience, and determination to carry this work through to its completion.

We express our deepest gratitude to our supervisor, Dr. Taha Zerrouki, for his unwavering guidance, his generosity with his time, and his constant encouragement throughout this project. His expertise in natural language processing and sign-language technology shaped this work in ways that go far beyond what these pages can reflect.

We sincerely thank the members of the jury for agreeing to evaluate this thesis and for the time and attention they devoted to reading it.

Our gratitude goes to all the Algerian Sign Language experts who participated in the evaluation campaign. Their involvement was the most essential part of this project, and their patience and willingness to share their knowledge made the dataset possible.

We thank the deaf associations and community members who welcomed us, answered our questions, and gave us a closer understanding of Algerian Sign Language and the daily realities of the Deaf community.

Finally, we thank our families, our friends, and our colleagues in the ISIL programme at Akli Mohand Oulhadj University - Bouira, for their support throughout these two years.

Dedication

To my parents, whose sacrifices made every page of this thesis possible.

To my sister and brother, for their constant encouragement.

To all my friends and fellow students who walked this path alongside me.

To every deaf person who deserves better access to education, communication, and the world around them.

Amara Ouail Said

Dedication

To my father, above all,
in the hope that this work makes him a little prouder of me.

To my mother and my brother,
who supported me at every step of the way.

To my friends, for the moments of doubt we turned into motivation together.

To the Algerian Deaf community,
whose language this work is a small attempt to honour.

Kherfi Lyes

ملخص

تُعدّ لغة الإشارة الجزائرية اللغةَ الطبيعية للمجتمع الصمّ في الجزائر، غير أنّها تفتقر إلى الدعم الرقمي الكافي مقارنةً بلغات الإشارة الأخرى. تهدف هذه الرسالة إلى ردم هذه الفجوة من خلال بناء قاموس رقمي موثّق لإشارات اللغة الجزائرية، مُرمَّز بنظام التوصيف الصوتي HamNoSys وصيغة SiGML القابلة للقراءة آلياً، ثم استخدامه لترجمة النصوص العربية إلى إشارات متحركة عبر شخصية ثلاثية الأبعاد.

اعتمدنا في بناء المعجم على مصدرَين تكامليَّين: مدوّنة DictaSign للغة الإشارة الفرنسية التي تضمّ نحو ألف مدخل مُرفَق بترميزات HamNoSys جاهزة، وتسجيلات فيديو محلية أجراها خبراء في لغة الإشارة الجزائرية لتغطية المفردات الخاصة بالسياق الجزائري. وللتحقق من صحة الإشارات، صممنا منهجية تحكيم تعتمد على الإجماع بين مُحكّمين مستقلين، ودعمناها بمنصة ويب تتيح للخبراء تقييم الإشارات عن بُعد وفق أربع درجات حكم.

أسفرت حملة التقييم عن 889 حكماً على 410 إشارة مرشّحة، صادرةً عن أربعة خبراء، أُثبتت صحة 310 منها. ومع ضمّ الإشارات الـ417 الواردة في عمل 3DZSignDB السابق، بلغ المعجم الجاهز للعرض بالصورة المتحركة 727 إشارة، أي ما يُقارب نصف القاموس الوطني الرسمي. يُشكّل هذا العمل أساساً قابلاً للتوسيع لدعم تعليم الصمّ وتسهيل وصولهم إلى الخدمات العامة.

الكلمات المفتاحية: لغة الإشارة الجزائرية، HamNoSys، SiGML، شخصية ثلاثية الأبعاد، ترجمة النص إلى إشارة، قاموس رقمي.

Abstract

Algerian Sign Language (ALSL) is the natural language of the Algerian Deaf community, yet it remains poorly supported by digital resources in comparison with contemporary sign languages. This thesis addresses that gap by constructing a validated digital dictionary of ALSL signs, encoded in the Hamburg Notation System (HamNoSys) and serialised in the machine-readable SiGML format, and using it to translate written Arabic into animated signs through a three-dimensional avatar.

Candidate signs were drawn from two complementary sources: the DictaSign corpus of French Sign Language, which provided approximately one thousand entries with existing HamNoSys transcriptions, and locally recorded videos performed by ALSL experts, covering vocabulary specific to the Algerian context. A consensus-based validation methodology, supported by a purpose-built web platform, allowed a panel of independent experts to evaluate each candidate sign across four judgement levels.

The evaluation campaign produced 889 judgements from four experts over 410 candidate signs, yielding 310 validated entries. Combined with the 417 signs of the prior 3DZSignDB work, the avatar-ready lexicon reaches 727 signs, covering close to half of the official national dictionary. The result is an extensible, standards-based resource designed to support deaf education and access to public services.

Keywords: Algerian Sign Language, HamNoSys, SiGML, 3D avatar, text-to-sign translation, digital dictionary.

Résumé

La langue des signes algérienne (ALSL) est la langue naturelle de la communauté sourde d'Algérie, mais elle reste insuffisamment soutenue par les ressources numériques par rapport aux langues des signes mieux documentées. Cette thèse comble cette lacune en construisant un dictionnaire numérique validé de signes ALSL, encodé dans le système de notation Hamburg (HamNoSys) et sérialisé dans le format lisible par machine SiGML, puis en l'utilisant pour traduire du texte arabe en signes animés à travers un avatar tridimensionnel.

Les signes candidats ont été tirés de deux sources complémentaires : le corpus DictaSign de la langue des signes française, qui a fourni environ un millier d'entrées avec des transcriptions HamNoSys existantes, et des vidéos enregistrées localement par des experts en ALSL, couvrant le vocabulaire propre au contexte algérien. Une méthodologie de validation basée sur le consensus, soutenue par une plateforme web dédiée, a permis à un panel d'experts indépendants d'évaluer chaque signe candidat selon quatre niveaux de jugement.

La campagne d'évaluation a produit 889 jugements de quatre experts sur 410 signes candidats, aboutissant à 310 entrées validées. Combiné aux 417 signes du travail antérieur 3DZSignDB, le lexique prêt pour l'avatar atteint 727 signes, couvrant près de la moitié du dictionnaire national officiel. Le résultat est une ressource extensible et normalisée, conçue pour soutenir l'éducation des sourds et l'accès aux services publics.

Mots-clés : langue des signes algérienne, HamNoSys, SiGML, avatar 3D, traduction texte vers signe, dictionnaire numérique.

Contents

List of Figures

Abbreviation	**Definition**
ALSL	Algerian Sign Language
ASL	American Sign Language
BSL	British Sign Language
LSF	Langue des Signes Française (French Sign Language)
HamNoSys	Hamburg Notation System
SiGML	Signing Gesture Markup Language
CWASA	Client Web Avatar Signing Architecture
FACS	Facial Action Coding System
CNN	Convolutional Neural Network
1D-CNN	One-Dimensional Convolutional Neural Network
XML	Extensible Markup Language
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
JS	JavaScript
3D	Three-Dimensional
ISIL	Ingénierie des Systèmes d'Information et de Logiciels
IPA	International Phonetic Alphabet
SMPL	Skinned Multi-Person Linear Model
TOC	Table of Contents
ViSiCAST	Virtual Signing: Capture, Animation, Storage and Transmission
eSIGN	Easy Sign (European research project)
WHO	World Health Organization

General Introduction

Language is the primary instrument through which people exchange knowledge, take part in society, and obtain essential services. For the majority who can hear, this exchange happens through speech. For deaf people, it happens through sign language: a complete natural language expressed with the hands, the face, and the body rather than with sound. When the hearing majority does not understand sign language, and when few tools exist to bridge the two, deaf people are left at a disadvantage in school, in hospitals, in public offices, and in everyday life. Reducing this disadvantage is the social motivation of the present work.

The scale of the issue is considerable. Sign languages are used by Deaf communities worldwide, and they are not a single universal system: more than three hundred distinct sign languages have been documented, each with its own vocabulary and grammar [3]. Among them, Algerian Sign Language (ALSL) serves the Algerian Deaf community. Yet ALSL has received far less documentation and technological support than languages such as American Sign Language or British Sign Language, leaving its users with comparatively few digital resources [7, 9].

Modern information technology has transformed how the hearing population communicates and learns, but much of this progress has bypassed sign-language users [8]. The consequences are felt across daily life. In education, deaf pupils struggle to follow lessons and to access learning materials prepared for hearing students. In healthcare, the absence of interpreting tools can compromise the accuracy of communication between a deaf patient and a clinician [16]. In public administration, official information is rarely available in an accessible signed form. Each of these gaps isolates deaf individuals from services that hearing citizens take for granted.

Sign languages are full natural languages in every linguistic sense. They possess their own phonology, morphology, syntax, and lexicon, organised independently of any spoken language [4, 5]. The recognition of this fact is comparatively recent: it was Stokoe who first showed that signs are not holistic gestures but are built from smaller, systematically combined units, demonstrating that sign languages have a structured linguistic system comparable to spoken languages [1]. This insight is the foundation on which any computational treatment of sign language must rest, because it means a sign can be analysed, encoded, and reproduced from its component parts.

This thesis proposes a contribution to closing the technological gap for Algerian Sign Language. It presents the construction of a structured, expert-validated dictionary of ALSL signs, encoded in a standard notation and used to drive a three-dimensional virtual signer that converts written Arabic into animated signs. The aim is not only to build a working system, but to produce a reusable linguistic resource that can support the education of deaf children, assist communication in public services, and serve as a foundation for future research in Algerian sign-language technology.

Three obstacles make this gap difficult to close. First, Although an official, standardised ALSL dictionary was published by the Ministry of National Solidarity in 2017 and introduced in schools [45], it exists only in printed form: ALSL still lacks a machine-readable digital dictionary, it lacks machine-translation systems able to convert text into signs, and it lacks tools that can present signs through animated three-dimensional figures. Second, the human interpreters who could otherwise bridge the gap are few in number and cannot be present in every classroom, clinic, or office, particularly in rural areas. Third, building a digital sign-language resource requires expertise that spans linguistics, deaf-community knowledge, and computer graphics simultaneously, and such combined expertise is scarce.

The central problem this thesis addresses is therefore the following: *how can a reliable, reusable digital dictionary of Algerian Sign Language be built and used to translate written Arabic into accurate signed output, given the scarcity of existing resources and the need for genuine linguistic validation by the Deaf community?* Answering this question requires both a sound methodology for collecting and validating signs and a concrete software system that can store, evaluate, and render them.

The work pursues the following objectives:

- To validate every sign through the independent judgement of qualified ALSL experts, so that the resulting dictionary reflects authentic usage rather than borrowed or invented forms.
- To encode each validated sign in a standard, machine-readable notation (HamNoSys, serialised as SiGML), so that the dictionary is portable and can drive any compatible avatar engine [34, 39].
- To develop a web-based platform that allows experts to evaluate signs efficiently and that records their judgements in a structured, recoverable form.
- To render the validated signs through a three-dimensional avatar, producing a text-to-sign translation usable in education and public communication.

- To deliver the result as a reusable resource that future researchers can extend, and that practitioners can apply in the teaching of deaf learners.

Chapter 1

Sign Language

1.1 Introduction

This chapter establishes the linguistic foundations on which the rest of the thesis is built. It explains what sign languages are, how they are structured, and why that structure makes computational processing possible. It then introduces Algerian Sign Language and its particular situation, and it closes by explaining how technology can support the education and inclusion of deaf people . Understanding this material is essential, because the quality of any sign-language system depends ultimately on respecting the linguistic properties of the language it represents.

1.2 Deafness and the Deaf Community

Hearing loss affects a large number of people across the world, and a significant proportion of those affected rely on a sign language as their main means of communication. For many of these individuals, deafness is not experienced primarily as a medical condition but as the basis of a cultural and linguistic community with its own language, history, and social life. This distinction matters for technology design: a tool built for the Deaf community must treat sign language as a genuine language and must be shaped with the participation of its users, rather than imposed from outside [15].

The communication barrier between deaf and hearing people has concrete effects. A deaf child in a mainstream classroom may receive instruction in a spoken language they cannot fully access; a deaf patient may be unable to describe symptoms accurately to a hearing doctor; a deaf citizen may be excluded from public announcements delivered only in speech or text [16]. These are not marginal inconveniences but barriers to education, health, and civic participation. Technology that can translate between written or spoken language and sign language therefore addresses a real social need, and this need is the motivation for the present work.

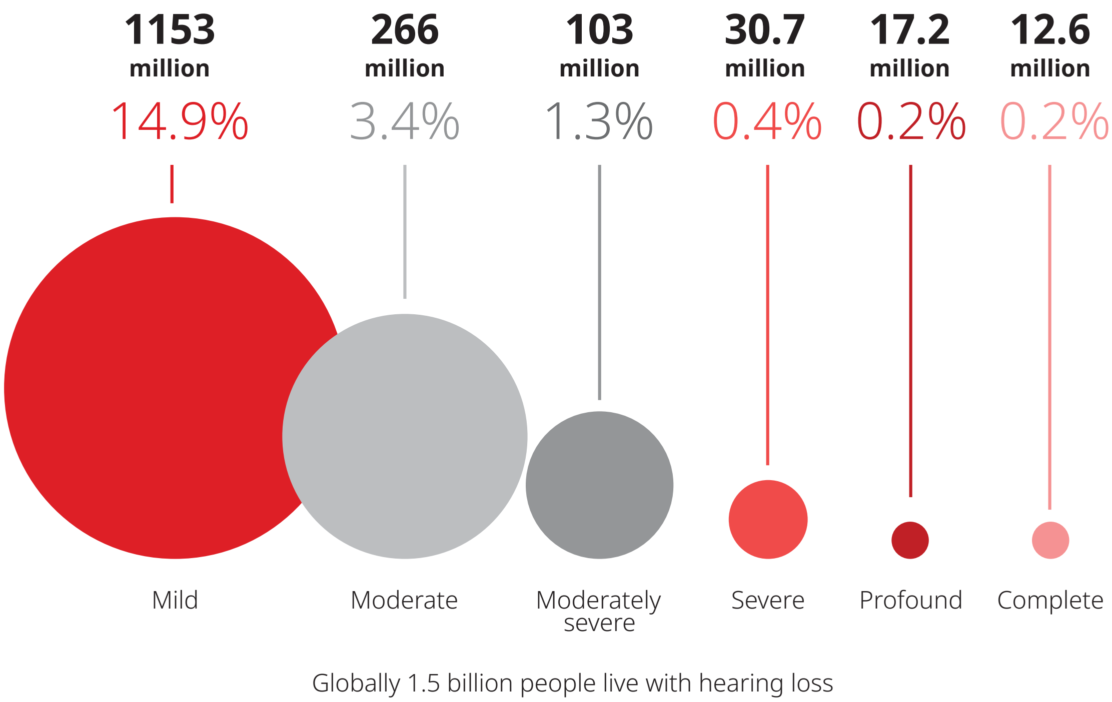

Figure 1.1: Number of people and percentage prevalence according to grades of hearing loss, source: the World Health Organization "World Report on Hearing" (2021)

1.3 Sign Languages as Natural Languages

A sign language is a natural language that arises spontaneously within a Deaf community and is passed from one generation to the next. Decades of linguistic research have shown that sign languages display the same kinds of structural organisation as spoken languages: a phonology that combines meaningless units into meaningful ones, a morphology that builds and modifies words, a syntax that arranges them into sentences, and a semantics that assigns meaning [4, 5]. They are not simplified codes, nor are they manual versions of the surrounding spoken language. A sign language such as ALSL is a language in its own right: its grammar is not borrowed from the spoken Arabic of the surrounding society, even though its vocabulary has been strongly shaped by contact with French Sign Language [7, 10].

A common misconception is that sign language is universal. In reality, different Deaf communities have developed different sign languages, and these are mutually unintelligible in the same way that spoken languages are [3]. The grammar of a sign language is not borrowed from the local spoken language; it follows principles specific to the visual medium in which it is expressed.

1.4 The Visual–Gestural Modality

The single most important difference between signed and spoken languages is the channel through which they are produced and perceived. Spoken languages use the vocal–auditory channel: sounds are produced by the vocal tract and received by the ear. Sign languages use the visual–gestural channel: meaning is produced by the hands, face, and body and received by the eye [5].

This difference is not merely superficial. The visual channel allows several pieces of information to be expressed at the same time, because the eye can take in the configuration of both hands, their movement, and a facial expression simultaneously. Spoken language, by contrast, is largely sequential, since sounds must follow one another in time. This capacity for simultaneous expression shapes the grammar of sign languages, allowing them to use space and parallel articulation in ways that have no direct equivalent in speech [4, 5]. Any system that aims to reproduce sign language faithfully must therefore be able to represent several channels of information at once, a requirement that directly influences the notation and rendering choices discussed in Chapter 2.

1.5 Phonological Structure of Signs

Phonology is the level of language concerned with the smallest contrastive units, the building blocks that have no meaning of their own but that combine to form meaningful words. Stokoe's foundational insight was that signs, like spoken words, can be broken down into such units, and that changing one of them can change the meaning of a sign [1]. Later research refined and extended his analysis, establishing a set of parameters that together describe how any sign is articulated [5]. These parameters are handshape, location, movement, orientation, and the non-manual components.

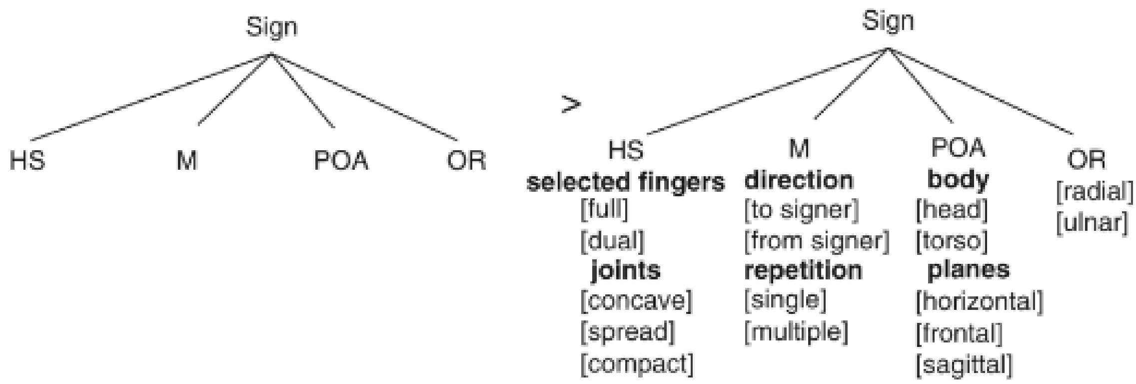

Figure 1.2: The phonological parameters of a sign in early and contemporary models. Adapted from [5].

Figure 1.2 shows the four parameters traditionally identified in early ASL phonological analyses: handshape, movement, place of articulation, and orientation [5]. More recent models of sign-language phonology generally recognise non-manual markers, including facial expressions and head movements, as an additional phonological parameter that contributes to linguistic meaning and grammatical structure [6].

1.5.1 Handshape

Handshape refers to the configuration of the fingers and thumb while a sign is produced. Each sign language has an inventory of permitted handshapes, and not every configuration the hand can physically adopt is used; the set is language-specific and subject to systematic constraints, much as each spoken language uses only a subset of the sounds the vocal tract can produce [5]. A change of handshape alone can distinguish one sign from another, which is why accurate handshape reproduction is critical in any avatar that renders signs.

1.5.2 Location

Location is the place, relative to the signer's body or in the space in front of them, where a sign is made. A sign may be articulated at the head, the chest, the non-dominant hand, or in neutral signing space. Location carries both lexical information, distinguishing one sign from another, and grammatical information, since positions in space can be used to track the participants in a sentence [5].

1.5.3 Movement

Movement describes the path the hands trace and the manner in which they move: a sign may move in a straight line, an arc, or a circle, and it may be quick, slow, repeated, or tense. Movement can distinguish signs from one another, and it can also carry grammatical meaning, for example marking the way an action unfolds over time [5]. Because movement is dynamic, it is one of the more demanding parameters to capture and reproduce accurately.

1.5.4 Orientation

Orientation refers to the direction in which the palm and fingers point during a sign. Two signs that share the same handshape, location, and movement may still differ only in orientation, and that difference can change their meaning. Orientation can also interact with grammar, contributing to the way signs are modified in context [5].

1.5.5 Non-Manual Markers

Not all of a sign language is produced by the hands. Facial expressions, the position of the eyebrows, movements of the head, the direction of eye gaze, and the posture of the upper body all carry linguistic information. These non-manual markers are not decorative; they are grammatically required components that can signal questions, negation, the marking of a topic, or conditional clauses [6]. A raised eyebrow or a head shake can change the grammatical function of an utterance. This has a direct consequence for technology: an avatar that moves only its hands and leaves its face still will fail to convey a large part of the grammatical meaning, and studies confirm that omitting non-manual features substantially reduces how well signed output is understood [15].

1.6 Morphology in Sign Languages

Morphology concerns the way languages build and modify words. Sign languages possess rich morphological systems that frequently exploit the spatial nature of the visual channel. A sign can be modified to indicate who is performing an action and who receives it by directing its movement through space; it can be repeated to indicate that an action happens habitually or repeatedly; and special constructions known as classifiers use particular handshapes to represent categories of objects and their movement [4, 6]. Many of these processes rely on spatial modulation that has no parallel in spoken languages, which underlines again that a sign language must be modelled on its own terms rather than as a manual copy of speech.

1.7 Syntax of Sign Languages

Syntax governs how signs are combined into sentences. Sign languages organise their sentences according to systematic, rule-based principles. They commonly use the space in front of the signer to encode grammatical relationships, they frequently arrange information using a topic–comment structure, and they mark the type of a clause, such as a question or a negation, using non-manual signals applied across a stretch of signing [4, 6]. These properties show that sign languages conform to the same general grammatical principles that govern human language as a whole, while adapting them to the possibilities of the visual medium.

1.8 Iconicity in Sign Languages

Sign languages often display iconicity, a resemblance between the form of a sign and its meaning: a sign for a cup may trace the shape of holding one. For a long time iconicity was thought to undermine the status of sign languages as true languages, but modern linguistics treats it differently. Iconicity coexists with full grammatical structure and does not reduce a sign language to mere pantomime; signs remain conventional, language-specific, and combinable according to grammatical rules even when they are partly iconic [5]. Recognising iconicity is useful for technology, because iconic relationships can aid both learning and the design of comprehensible animations.

1.9 Written and Notational Representations of Sign Language

Spoken languages have long-established writing systems; most sign languages do not have a writing system in everyday use. This absence is one of the central difficulties for sign-language technology, because a computer needs some written or symbolic form of a sign in order to store, search, and reproduce it [15]. Over time, several notation systems have been devised to fill this role, beginning with Stokoe's pioneering notation for American Sign Language [1]. These systems record the phonological parameters of a sign in symbolic form, so that a sign can be written down precisely and later reconstructed. The notation systems most relevant to avatar-based synthesis, in particular the Hamburg Notation System and its machine-readable serialisation, are examined in detail in Chapter 2; here it is enough to note that the lack of a natural written form is exactly the problem that such notations are designed to solve.

1.10 Algerian Sign Language (ALSL)

Algerian Sign Language is the natural language of the Algerian Deaf community. Like every sign language, it exhibits the phonological parameters, spatial morphology, and rule-governed syntax described above, and it functions as a complete means of communication for its users [7]. It is a language in its own right, separate from spoken or written Arabic, even though its users live within an Arabic-speaking culture.

ALSL developed within the Algerian Deaf community and has been shaped over time by contact with French Sign Language, a result of the educational and cultural connections through which deaf education developed in Algeria [10,11]. Much of the early schooling of deaf children in Algeria drew on French-based methods and materials, so a substantial part of the ALSL lexicon was inherited from, or shaped by, French Sign Language. Comparative studies of the two languages find substantial overlap in their vocabularies while confirming that ALSL is structurally independent, with its own grammatical organisation [10]. This relationship is significant for the present work: because so many signs are shared, a French Sign Language sign is a reasonable *starting hypothesis* for the corresponding Algerian one. It is only a hypothesis, however, since the two languages have diverged and a shared concept may be signed differently in Algeria. Every candidate drawn from French Sign Language is therefore treated as provisional and checked against genuine Algerian usage by the expert validation described in Chapter 3, which is what turns a borrowed candidate into a confirmed ALSL entry.

A further defining characteristic of ALSL is its regional variation. Studies of community sign systems across Algeria document noticeable differences between regions, reflecting the way sign languages evolve within local social networks and schools [12]. This variation is both a richness and a practical challenge: it means that for a given concept there may be more than one signed form in use, and that fixing a single canonical sign requires care and the involvement of experts who know the community. Despite this variation, the Ministry of National Solidarity published a unified Algerian Sign Language dictionary in 2017, which has since been adopted in schools as a national reference [45]. This standard does not eliminate regional differences, but it provides an authoritative baseline against which signs can be compared. The methodology in Chapter 3 builds on the same principle, requiring agreement among independent experts before a sign is accepted.

Nonetheless, this standardisation remains partial and, crucially, exists only in printed form: ALSL is still far less supported by machine-readable digital resources than many other sign languages, and challenges of documentation, education policy, and accessibility continue to affect its development [9]. The limited availability of standardised *digital* resources is precisely the gap that this thesis sets out to help fill.

1.11 Computational Modeling of Sign Languages

Computational modelling of sign language means representing its linguistic parameters in a form that a computer can process, for the purpose of recognition, synthesis, or translation. Because signs are composed of identifiable parameters, they can be encoded systematically; but a faithful model must capture not only the manual components but also the non-manual ones, must handle the timing of dynamic movement, and must represent the use of space [13]. These requirements make sign-language modelling more demanding than a naive view might suggest, and they explain why standardised notations and carefully built datasets are so important.

1.12 Automatic Sign Language Translation

Automatic sign-language translation refers to computer systems that convert between a spoken or written language and a sign language. Two complementary directions exist. *Recognition* takes signed input, typically video, and converts it into text or speech; it draws heavily on computer vision and machine learning [8, 13]. *Synthesis*, the direction pursued in this thesis, takes written or spoken language as input and produces signed output, usually through an animated avatar. Synthesis is essential because, in the absence of a widely used written form for sign languages, generating clear animated signs is the natural way to make textual information accessible to deaf users [15]. The system developed in this work is a synthesis system: it converts written Arabic into animated Algerian Sign Language.

1.13 The Role of Technology in Deaf Education and Inclusion

The ultimate purpose of the technology described in this thesis is social. A reliable text-to-sign system, built on a validated dictionary, can support deaf education in several ways. It can act as a tireless tutor that demonstrates vocabulary and can repeat a sign as many times as a learner needs, at an adjustable pace, in a way that printed materials cannot [31]. It can make written educational content accessible by presenting it in signed form, supporting deaf pupils who are still developing literacy in the written language. Beyond the classroom, such systems can improve access to public information and services, and can reduce dependence on the limited number of human interpreters [32].

A dictionary of validated signs has value even beyond the system that uses it. It is a documented record of part of the ALSL lexicon, which contributes to the preservation and standardisation of the language, and it provides a foundation that other researchers can extend and that educators can draw upon. In this sense the work is intended not as an end in itself but as a building block for the wider effort to support the Algerian Deaf community.

1.14 Conclusion

This chapter has established that sign languages, including Algerian Sign Language, are full natural languages with their own phonology, morphology, and syntax, expressed through the visual–gestural channel. It has shown that signs are built from identifiable parameters, which is what makes their computational treatment possible, and that non-manual features are grammatically essential and cannot be ignored by any faithful system. It has introduced the particular situation of ALSL, including its relationship with French Sign Language, its regional variation, and its limited digital resources, and it has explained why a text-to-sign system matters for the education and inclusion of deaf people. The next chapter turns to the technology used to represent and render signs: the notation systems that give signs a written form, and the three-dimensional avatars that bring them to life.

Chapter 2

Three-Dimensional Avatars and Sign Representation

2.1 Introduction

The previous chapter showed that signs can be analysed into well-defined parameters and that any faithful reproduction must capture both manual and non-manual components. This chapter examines the technology that makes such reproduction possible. It reviews the two main approaches to presenting signs by computer, explains how three-dimensional avatars are constructed and animated, and presents the notation systems that allow a sign to be written down and then rendered. It pays particular attention to the Hamburg Notation System and its machine-readable form, SiGML, because the dataset that forms the core contribution of this thesis is built upon them. The chapter closes by reviewing related research and positioning the present work within it.

2.2 Approaches to Presenting Sign Language by Computer

There are two broad ways to present sign language through a computer. The first is to record video of human signers and play back the appropriate clips. This approach produces completely natural signing, but it is rigid: a video library cannot generate new sentences, it is costly to extend, it consumes large amounts of storage and bandwidth, and modifying a sign means filming it again [15]. The second approach is to use a three-dimensional avatar, a virtual human that produces signs from a symbolic description. An avatar is far more flexible: once the system can render signs from descriptions, new vocabulary can be added by writing new descriptions rather than filming, sentences can be assembled dynamically, and the data needed to drive the avatar is compact [15, 30]. These advantages are why avatar-based synthesis was chosen for this work, and why the dictionary at its core is stored as symbolic descriptions rather than as video.

2.3 3D Avatar Technology

A three-dimensional avatar is a virtual human created with computer-graphics techniques and animated to perform gestures, including the gestures of sign language. In a sign-language system the avatar is the output stage: it takes a processed description of a sign and turns it into a visible performance, reproducing the handshapes, movements, orientations, and facial expressions that make up the sign [14]. To do this convincingly, an avatar must combine an accurate model of the human body with the ability to articulate the fingers precisely and to display facial expression, since both are essential to sign language.

Avatar technology has advanced considerably. Early systems, produced between roughly 2000 and 2010, relied on hand-crafted animations and on symbolic descriptions such as HamNoSys and SiGML to drive the figure; their output was usable but sometimes stiff [39, 41]. More recent research has explored data-driven methods that learn signing motion from large collections of recorded movement, improving the naturalness and smoothness of the result [14, 24]. The system in this thesis follows the established notation-driven approach, because it offers precise control over each sign and rests on mature, well-documented standards.

Figure 2.1: An example of a signing avatar.

2.4 Construction of 3D Human Avatars

A three-dimensional human avatar is built from several layers that work together. The visible surface is a mesh, a network of many small polygons that forms the shape of the body and face. Beneath the mesh is a skeleton, a hierarchy of connected joints that can be rotated to pose the figure. The mesh is attached to the skeleton through a process called skinning, which determines how the surface deforms as the joints move; good skinning produces smooth, natural bending at the joints, while poor skinning produces visible artefacts [25, 27]. On top of this, material and lighting models give the surface a realistic appearance.

For sign language, two capabilities matter more than photorealism. The first is detailed articulation of the hands, since handshape is a core parameter of every sign and the fingers must adopt precise configurations. The second is facial animation, because non-manual markers are grammatically essential. Facial movement is often described using the Facial Action Coding System, which breaks expression into a set of independent component movements that can be combined to produce the brow raises, head positions, and other expressions that sign language requires [29]. Research on capturing the body, hands, and face together has produced detailed models that represent all three within a single framework, reflecting how important the combination is for expressive signing [23].

Figure 2.2: An example of avatar construction.

2.5 Uses of 3D Avatars

Three-dimensional signing avatars have been applied across many domains, which demonstrates both their maturity and their relevance to the present work. In telecommunication, avatars can provide signed interpretation of spoken input in close to real time, reducing reliance on scarce human interpreters [30]. In education, they serve as patient virtual tutors that can demonstrate vocabulary and grammar repeatedly and at adjustable speed, with measurable benefits for learners [31]. In broadcasting, avatars can add signed interpretation to television and emergency announcements far more cheaply than video [32]. In healthcare, they can help deaf patients understand medical information [15]. They have also been integrated into virtual-reality and social platforms to allow deaf users to participate on equal terms [33]. These applications confirm that avatar-based sign presentation is a practical technology, and several of them, especially the educational uses, align directly with the social purpose of this thesis.

2.6 Representing Sign Language for Avatars

To make an avatar sign, the sign must first be described in a form the avatar can execute. Such a description must capture the five phonological parameters introduced in Chapter 1, namely handshape, location, movement, orientation, and non-manual features, together with the timing of the movement and the use of space [34]. The fidelity of this description directly determines how well the signed output is understood: when key parameters, and especially non-manual features, are represented inaccurately or omitted, comprehension drops sharply [15]. For this reason, sign-language synthesis relies on standardised notation systems that describe signs precisely and unambiguously, and that can be converted into the animation parameters an avatar consumes.

2.7 Notation Systems for Sign Language

Because sign languages generally lack an everyday writing system, researchers have developed notation systems to record signs in symbolic form. The earliest was the system devised by Stokoe for American Sign Language, which introduced the idea of writing a sign in terms of its parameters [1]. Later systems aimed at broader coverage and finer detail. Among these, the Hamburg Notation System has become the most widely used in avatar-based synthesis, because it is detailed, it applies across different sign languages, and it can be processed by computer [34]. The remainder of this chapter focuses on HamNoSys and on SiGML, the markup language that turns HamNoSys into a form an avatar engine can execute, because together they are the representation on which this thesis is built.

2.8 The Hamburg Notation System (HamNoSys)

The Hamburg Notation System, known as HamNoSys, is a notation for sign languages developed at the University of Hamburg in 1984 and refined over four successive versions, the current one dating from 2004 [34, 35]. It grew out of the tradition begun by Stokoe but, unlike Stokoe's notation, it was deliberately designed not to depend on any single national sign language or fingerspelling alphabet, so that it can be applied internationally to any sign language [34]. In this respect HamNoSys plays a role for sign languages comparable to the role the International Phonetic Alphabet plays for spoken languages: it is a phonetic description of how a sign is articulated rather than a practical everyday writing system [34].

The notation provides a large inventory of roughly two hundred and ten iconically motivated symbols, each standing for one articulatory aspect of a sign such as a handshape, an orientation, a place of articulation, or a movement [34, 36]. Because the symbols are visual and systematic rather than arbitrary, a transcriber can learn to read a sign description directly from its symbols, and a computer can parse the same description into the parameters it needs to drive an avatar.

Two features make HamNoSys especially suitable for this work. First, it is phonetically based: it records how a sign is articulated, in terms of the same parameters described in Chapter 1, rather than tying the description to any spoken-language word. This makes it equally applicable to ALSL and to the French Sign Language signs from which many ALSL candidates are drawn. Second, it is designed to be processed by computer, and it can be expressed in a structured electronic form, which is the bridge to avatar animation [34,35].

2.8.1 Overall Structure of a Transcription

A HamNoSys transcription is not a free arrangement of symbols but an ordered sequence whose parts always appear in the same order. The description is built from six blocks: an optional *symmetry operator*, optional *non-manual features*, and then the four mandatory manual blocks, namely *handshape*, *hand orientation*, *hand location*, and *movement* [34,36, 37]. Reading a transcription therefore means reading these blocks in turn: first whether and how the two hands are related, then the configuration of the dominant hand, then where it points, then where it is placed, and finally how it moves.

Figure 2.3 shows this organisation for a two-handed sign. The leftmost symbol is the symmetry operator, which states that the sign is two-handed and how the non-dominant hand mirrors the dominant one; the blocks that follow describe the dominant hand (shown in blue) and, where the hands differ, the non-dominant hand (shown in green); the parameters that apply only to non-symmetric signs are marked separately (shown in red). Encoding the relationship between the hands once, at the start, avoids repeating the full description for both hands and keeps the transcription compact.

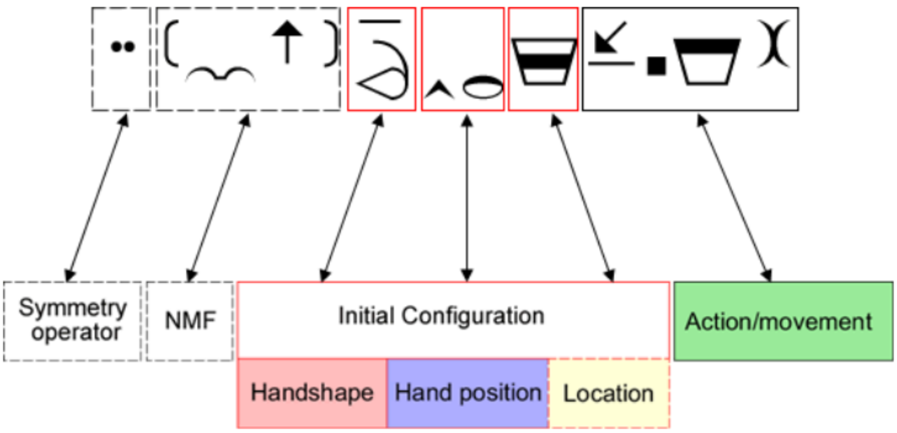

Figure 2.3: Structure of a HamNoSys transcription. [26]

The four manual blocks correspond exactly to the phonological parameters introduced in Chapter 1, which is why a HamNoSys string can be regarded as a written form of the sign's phonology. The subsections below describe each manual block in turn and illustrate it with the symbols actually used.

2.8.2 Handshape

The handshape block specifies the configuration of the fingers and thumb at the start of the sign. HamNoSys builds handshapes from a small set of base forms and then modifies them, rather than giving every possible hand configuration its own symbol. The base forms range from a closed fist through partly open hands to a fully spread hand, and each is denoted by a distinct symbol [34]. These base forms are then refined in two ways: diacritical symbols specify the position of the thumb, and further symbols indicate bending of the fingers, so that a large number of attested handshapes can be written from a compact and systematic set of building blocks [36].

Figure 2.4 shows this construction. The top row gives several base forms with their symbols; the middle row shows how the same base form is combined with thumb-position diacritics; and the bottom row shows how bending is added. This compositional approach is well suited to computational use, because a handshape can be generated by combining a base symbol with its modifiers rather than stored as an indivisible unit, and it matches the way an avatar articulates the hand from joint positions.

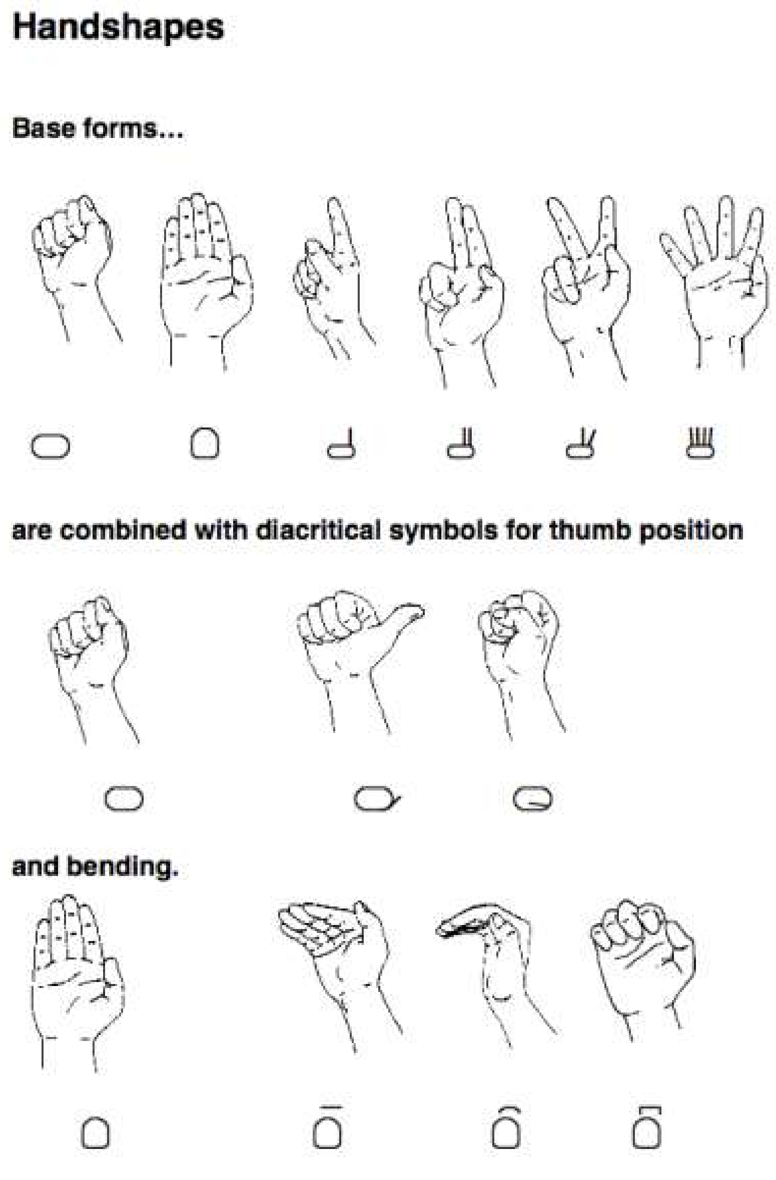

Figure 2.4: Handshapes: base forms, thumb position, and bending.

2.8.3 Hand Orientation

Once the handshape is fixed, the orientation block records the direction in which the hand points. HamNoSys describes orientation in terms of the direction of the extended finger, that is, the way the hand is aimed in space, choosing from a set of directions defined relative to the signer's body [34]. These directions are organised in three planes, so that the same notation can express pointing up, down, left, or right, pointing toward or away from the signer, and the intermediate diagonals.

Figure 2.5 illustrates the available directions in the three reference planes. Because orientation can change the meaning of a sign that is otherwise identical in handshape, location, and movement, the notation treats it as an independent block, and an avatar must reproduce it faithfully for the sign to be understood correctly.

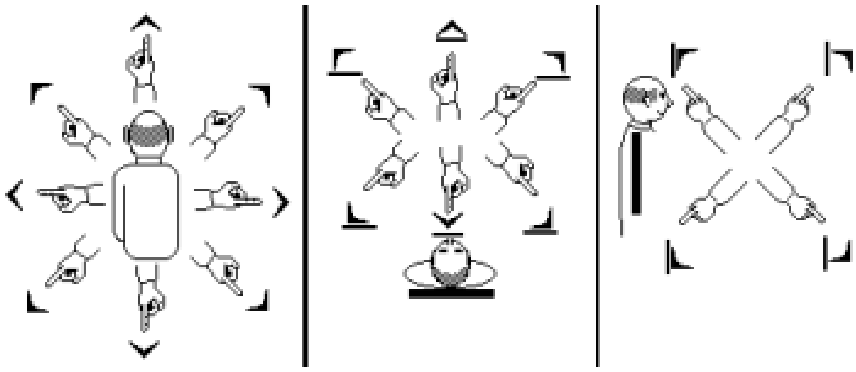

Figure 2.5: Hand orientation directions in three planes.

2.8.4 Hand Location

The location block states where, relative to the signer's body or in the signing space in front of them, the sign is articulated. HamNoSys offers a set of location symbols, conventionally divided into places on or near the body, distances from the body, and positions in neutral signing space [37]. A sign may be placed at the head, the chest, the non-dominant hand, or in the space in front of the signer, and the same symbol set covers all of these. Location carries lexical information, since two otherwise identical signs made at different places are different signs, and it also supports the grammatical use of space described in Chapter 1.

2.8.5 Movement

The movement block is the most elaborate part of a transcription, because movement is dynamic and can be composed of several elements. HamNoSys distinguishes *path movements*, in which the hand travels through space along a straight line, an arc, or a circle, from *local movements*, such as a change of handshape or a rotation of the wrist performed without moving the hand through space [34,36]. Movements can also be modified, for example by repetition or by being performed by both hands at once, and several movement symbols can be combined in sequence to describe a complex motion.

Figure 2.6 shows how a straight upward path movement combines with different curved or wavy modifiers, each producing a recognisably different motion of the same hand. Circular movements form a particularly rich subgroup: Figure 2.7 presents the circular movement symbols arranged by the plane in which the circle is traced and by its direction, illustrating how systematically the notation covers the space of possible movements.

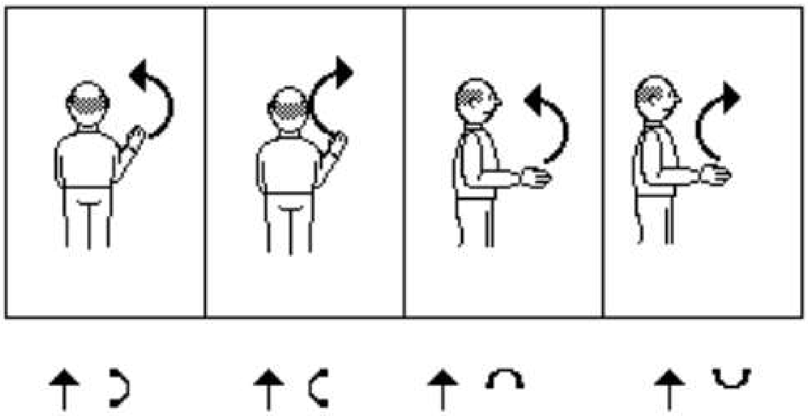

Figure 2.6: Straight path movement with curved modifiers.

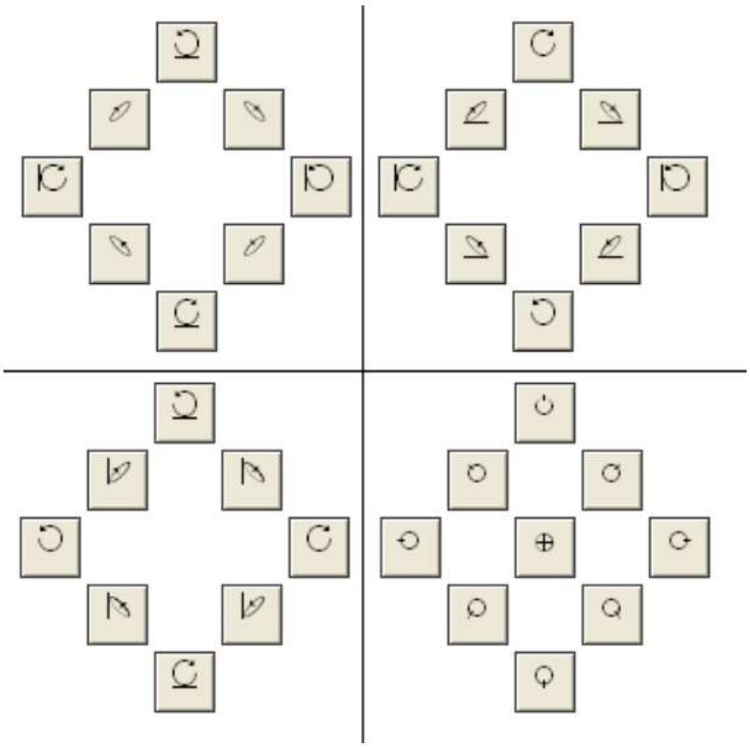

Figure 2.7: Circular movement symbols by plane and direction.

2.8.6 A Worked Example

Bringing the blocks together clarifies how a complete sign is written. Figure 2.8 shows the sign for HOUSE together with its HamNoSys transcription. Reading the symbols in order gives the handshape of the two flat hands, their orientation, their location in the signing space, and the movement that traces the outline of a roof, so that the written string fully specifies the sign that the photograph on the right shows being performed. This example also illustrates the iconicity discussed in Chapter 1: the movement of the sign resembles the shape of the object it denotes, yet the description remains a precise, parameter-by-parameter encoding rather than a vague gesture.

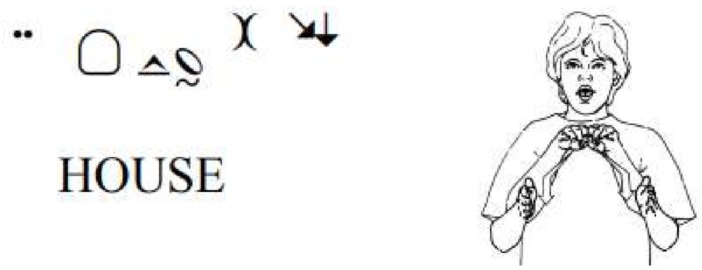

Figure 2.8: The sign HOUSE and its HamNoSys transcription.

2.8.7 Why HamNoSys for This Work

The properties surveyed above explain why HamNoSys was adopted as the encoding for the lexicon built in this thesis. It is phonetic and language-independent, so it applies to ALSL exactly as it applies to the French Sign Language signs from which many ALSL candidates are drawn. It is compositional, so a few hundred symbols suffice to describe an open-ended vocabulary, and entries can be written by combining building blocks rather than by recording each sign whole. It is fixed in order, so every transcription can be parsed unambiguously by a program. And it is mature and standardised, with stable tool support and a defined path to a machine-readable form [34, 35]. That machine-readable form, SiGML, is the subject of the next section.

2.9 Signing Gesture Markup Language (SiGML)

While HamNoSys provides the human-readable notation, an avatar engine needs the same information in a structured, machine-readable form. This is the role of the Signing Gesture Markup Language, SiGML, an XML-based format developed within the ViSiCAST and eSIGN European projects to serialise HamNoSys descriptions so that avatar systems can execute them [39, 40]. A SiGML document encodes the manual components of a sign, namely the handshape, orientation, location, and movement, alongside the non-manual components, in a set of structured elements that an animation engine can read directly.

SiGML is important to this thesis for the same reason HamNoSys is: it is a standard. Because the dictionary is stored as SiGML, it is not tied to one particular piece of software. The same files could be rendered by any compatible engine, and the rendering engine could in principle be replaced without rewriting the dictionary [39]. This independence between the linguistic content and the rendering technology is a deliberate design choice, and it is what makes the dataset a durable, reusable resource rather than a component locked inside one application.

2.10 From Notation to Animation: The Pipeline

The representation used in this work forms a clear pipeline. A sign is first transcribed in HamNoSys, capturing its parameters in symbolic form. The HamNoSys transcription is then serialised as SiGML, producing a structured electronic description. Finally, the SiGML description is passed to an avatar engine, which interprets it and animates the virtual human to perform the sign [34,39]. Each stage has a distinct purpose: HamNoSys provides a precise, linguistically grounded description; SiGML provides a machine-readable interchange format; and the avatar engine provides the visible output. This separation of concerns, in which the linguistic description is kept independent of the graphics, follows the design established by the projects that created SiGML and is central to the architecture of the system presented in Chapter 3.

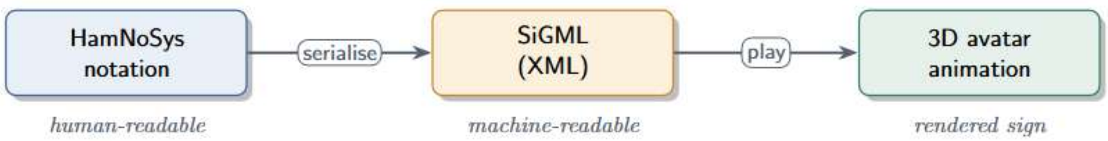

Figure 2.9: A simple pipeline diagram: HamNoSys notation → SiGML serialisation → avatar rendering..

2.11 Avatar Animation Engines

The component that turns a SiGML description into a visible performance is the avatar animation engine. The engines most associated with HamNoSys and SiGML originate from the ViSiCAST and eSIGN projects and their successors, most notably the JASigning system developed at the University of East Anglia, which reads SiGML and animates a virtual signer in real time [38,41]. These engines were created specifically for sign language, and they understand the structure of HamNoSys-derived descriptions, which makes them a natural choice for a notation-driven system such as the one in this thesis. Modern web-based rendering allows such avatars to run inside an ordinary web browser without any software installation, which is an important practical advantage for reaching deaf users on everyday devices.

2.12 The ViSiCAST and eSIGN Projects

Much of the technology described in this chapter originates from the ViSiCAST project and its successor eSIGN, European initiatives that established the technical foundations for avatar-based sign-language access to digital content [41]. These projects produced several enduring results: an avatar specification capable of detailed hand and facial articulation, the SiGML notation for representing signs in machine-readable form, an editor for composing HamNoSys, and an animation engine able to render signing in real time, all evaluated with Deaf users [39, 42]. Their work shifted sign-language interpretation from sole reliance on human signers towards scalable, technology-based solutions, and the standards they created remain in use today. The present thesis builds directly on this legacy by adopting HamNoSys and SiGML as its representation.

Figure 2.10: An example of avatar construction.

2.13 Related Works in Avatar-Based Sign Synthesis

Recent research has concentrated on making avatar signing more natural and more acceptable to Deaf communities. One line of work models signing as continuous motion learned from recorded data: a representative example reconstructs and generates three-dimensional signing motion from skeletal data, producing smooth, natural animation rather than isolated, mechanical signs [14]. A second line emphasises that technical quality alone is not enough. Work on the representation of signing avatars argues that faithful modelling must include non-manual signals such as facial expression and upper-body movement, and that avatars should be designed with the participation of the Deaf community if they are to be accepted [15]. Large datasets of three-dimensional signing motion have also been assembled to support such research [24]. Together these works define the direction of the field: combining accurate, linguistically grounded representation with natural motion and community involvement. The present work contributes to this direction by producing a validated, standards-based dictionary for a language, ALSL, that has so far been largely absent from it.

2.14 Recent Works in Algerian Sign Language

A small but growing body of work addresses Algerian Sign Language specifically. This section reviews the most relevant contributions.

3D Avatar-Based Translation for Algerian Sign Language The work most closely related to this thesis is the development of an automated Arabic-to-ALSL translation system based on three-dimensional avatar technology [17]. Its purpose is to reduce the communication barrier between Arabic speakers and the Algerian Deaf community by translating Arabic text into animated signs. It combines two methods: a notation-based method that uses HamNoSys to encode signs and converts them to SiGML for avatar execution, ensuring linguistic consistency and compatibility with standard frameworks; and a motion-capture method that records real human movement and transfers it to the avatar to improve realism [17]. Its architecture comprises an input module for Arabic text, a processing module that performs linguistic analysis and rule-based translation, and an output module that renders the result through the avatar. The authors note important limitations, in particular the scarcity of large-scale ALSL datasets and the difficulty of modelling non-manual features accurately [17]. The dataset and methodology developed in the present thesis respond directly to the first of these limitations, by building and validating exactly the kind of resource whose scarcity is identified.

1D-CNN-Based Algerian Sign Language Recognition In the domain of recognition rather than synthesis, one study presents a system for recognising Algerian Sign Language vocabulary used in healthcare [16]. It extracts landmark points for the hands and body from video, combines them with engineered features, and feeds them to a one-dimensional convolutional neural network, achieving high accuracy on a set of medically relevant signs and demonstrating the value of such a tool in clinical settings where interpreters are scarce [16]. More broadly, reviews of Arabic sign-language recognition report that deep convolutional networks consistently outperform older machine-learning methods, particularly when combined with sequence models that capture how signs unfold over time [18, 19]. This body of work addresses the complementary, recognition direction; the present thesis addresses synthesis.

ALGSL89: An Algerian Sign Language Dataset The development of structured datasets is essential to progress, and one contribution provides a dataset of Algerian Sign Language gestures together with a real-time recognition system based on a convolutional neural network [20]. The system learns features directly from images of hand signs and classifies them, achieving strong accuracy and demonstrating suitability for real-time use [20]. Its focus is on static signs, and the authors note that capturing the dynamics of continuous signing remains a challenge, a limitation also observed in other early recognition systems [19]. The availability of such annotated datasets allows different methods to be compared on a common basis, which underlines the broader value of dataset-building work of the kind this thesis undertakes for synthesis.

A Deep Learning Framework for Bilingual Sign Language Interpretation A further contribution is a master's thesis that proposes a real-time bilingual sign-language interpretation system [21]. It combines several techniques: convolutional networks to extract features from hand gestures, a hybrid convolutional-recurrent architecture to capture both spatial and temporal information, and an object-detection method to locate the hands efficiently [21]. The system relies mainly on alphabet, or fingerspelling, datasets rather than whole-word datasets, which simplifies vocabulary management but demands very precise handshape detection [22]. Notably, it supports communication in both directions: after recognising signed input it produces text, and it also includes an avatar module that generates signs from text, enabling text-to-sign translation [21]. Its reliance on alphabet-level data limits its scalability to large vocabularies, but it illustrates the value of combining recognition and synthesis, and its avatar component shares the text-to-sign goal pursued in this thesis.

2.15 Positioning of the Present Work

The review above shows a clear gap. The technology for representing and rendering signs through avatars is mature and standardised, resting on HamNoSys, SiGML, and engines descended from the ViSiCAST and eSIGN projects [34, 39, 41]. Recent research has advanced the naturalness of avatar signing and emphasised community involvement [14, 15]. Work specific to Algerian Sign Language exists, but it concentrates largely on recognition, and the most closely related synthesis work identifies the scarcity of validated ALSL datasets as a key obstacle [16, 17, 20]. The present thesis is positioned exactly in this gap. It contributes a validated, standards-based dictionary of Algerian Sign Language signs, encoded in HamNoSys and SiGML and confirmed by independent experts, and it uses this dictionary to drive a three-dimensional avatar. In doing so it addresses the very limitation that related work has highlighted, and it provides a reusable foundation for the education of deaf people and for future research in Algerian sign-language technology.

2.16 Conclusion

This chapter has reviewed the technology on which the thesis is built. It explained why avatar-based synthesis is preferable to video for a flexible, extensible system, and described how three-dimensional avatars are constructed and animated, stressing the importance of hand articulation and facial expression for sign language. It presented the notation systems that give signs a written form, focusing on HamNoSys and its machine-readable serialisation SiGML, and it described the pipeline from notation to animation and the engines that perform it. Finally, it reviewed related research, including work specific to Algerian Sign Language, and showed that the present thesis fills a clear gap by providing a validated, standards-based ALSL dictionary for avatar-based synthesis. The following chapter presents the conception and methodology through which this dictionary was designed and built.

Chapter 3

Design

3.1 Introduction

The problem this thesis addresses is the scarcity of a validated, reusable digital lexicon for Algerian Sign Language. Compared with better-supported languages, ALSL has almost no machine-readable datasets or text-to-sign tools, and without a structured dictionary of signs encoded in a form an avatar can read. Building such a dictionary raises a further difficulty: each candidate sign must be confirmed as correct by qualified ALSL experts.

The principal objective of this thesis is to build a validated, reusable dictionary of Algerian Sign Language, encoded in SiGML, that can drive a three-dimensional avatar to translate written Arabic into signs, and so support the education and inclusion of the Algerian deaf community. The dictionary, not the avatar, is the contribution: the rendering of SiGML into a moving avatar is carried out by the existing 3DZSignDB system [17], which our dataset feeds.

This chapter is concerned with the design of the system that meets that objective. Its role is to fix the design before the implementation reported in Chapter 4. It answers four design questions in turn: where the signs come from and how a candidate sign is decided to be correct (the validation methodology); how each one is turned into an avatar-ready description (the HamNoSys-to-SiGML pipeline); how the platform that gathers those decisions is structured (its architecture and data model); and how the practical difficulty of working with sign-language experts shaped every one of these choices. Because the chapter is about design, it is organised around the schemas of the system: the overall architecture, the conversion pipeline, the validation logic, the platform architecture, and the data model.

3.2 Overall Design of the System

The system is designed as a pipeline that turns raw sign material into a validated, avatar-ready dataset. Figure 3.1 shows its overall structure. Candidate signs are first collected as videos from two sources and added to the platform, where experts judge them under a consensus rule. Only the signs that the experts validate go forward. For each validated sign its HamNoSys transcription is then obtained / retrieved from the DictaSign site for signs taken from that corpus, or written by hand for local signs that have no transcription — and serialised to SiGML through the eSIGN editor. The resulting SiGML entries form the validated dataset, which is what the existing JASigning avatar reads to produce signed output.

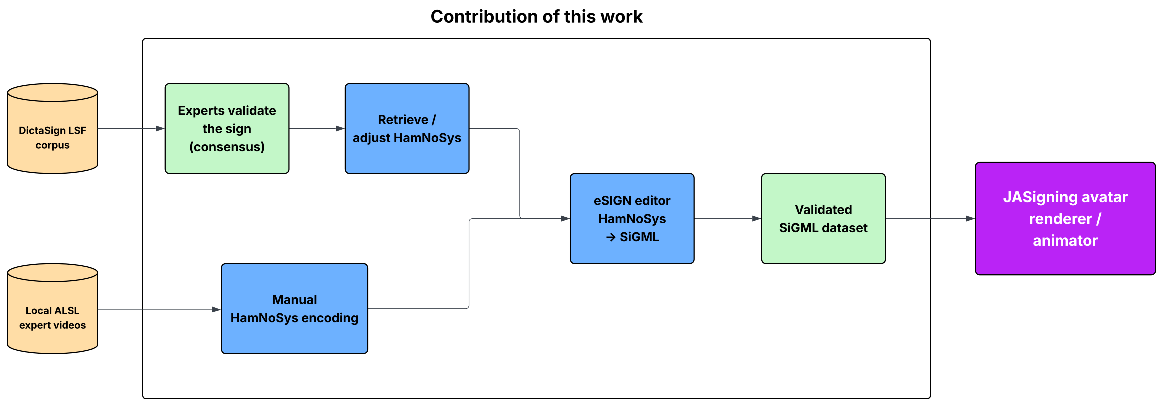

Figure 3.1: Overall design: from the sign sources to the avatar-ready dataset. The dashed region is the contribution of this work; the avatar engine is the existing rendering system that the dataset feeds.

Three components are therefore designed in this chapter, and they correspond to the three boxes inside the dashed region of Figure 3.1: the validation methodology that decides which candidate signs are authentic, the evaluation platform that puts that methodology into practice, and the encoding pipeline that turns each validated sign into a SiGML entry. The avatar engine lies outside this boundary and is treated as a fixed consumer of the dataset, which is why the design keeps the linguistic content (SiGML) strictly independent of the rendering technology: any engine that reads SiGML could replace JASigning without touching the dataset.

3.3 Design of the Sign Acquisition and Encoding Pipeline

The core of the design is the route by which a sign becomes a SiGML entry. This section sets out the choice of sources, the role of HamNoSys as the common representation, and the conversion pipeline itself.

3.3.1 Choice of Data Sources

The design draws candidate signs from two complementary sources, each answering a different need. The first and main source is the DictaSign corpus of French Sign Language [44], which provides on the order of a thousand entries. It was chosen for two reasons that follow directly from Chapter 1 and Chapter 2: Algerian Sign Language is historically and lexically close to French Sign Language, so a large share of its signs are plausible Algerian candidates once an expert confirms them; and, decisively for the workload, the DictaSign entries already carry HamNoSys transcriptions, so these signs do not have to be encoded from scratch. The second source covers what the corpus cannot: signs specific to Algerian usage, including religious and cultural vocabulary absent from a French corpus. These are obtained as reference videos performed by local experts and are encoded by hand. The two sources are deliberately funnelled into a single uniform format downstream, so that the platform, the validation rule, and the avatar treat every entry identically regardless of where it came from.

3.3.2 The Common Encoding: HamNoSys

HamNoSys and SiGML were defined in Chapter 2 and are not redefined here; in this design they play a specific structural role. HamNoSys is the common encoding/format: it is the one representation in which both kinds of source meet. A corpus sign arrives already in HamNoSys; a locally recorded sign is brought into HamNoSys by manual encoding. Once a sign is in HamNoSys, the rest of the pipeline is identical for both, because HamNoSys is phonetic and language-independent and can be serialised mechanically to SiGML. Choosing a single common format is what lets two very different acquisition routes share one conversion and one storage format.

3.3.3 Designing the Conversion Pipeline

Figure 3.2 shows the design of the conversion pipeline from a source sign to an accepted SiGML entry. The two entry routes converge on the eSIGN editor, which holds the HamNoSys transcription against its Arabic headword; from the editor the entry is exported as SiGML; the SiGML is rendered by the avatar; and the rendered movement is checked against the reference. A correct rendering is added to the validated dataset, while a mismatch sends the transcription back to the editor for revision. This verify-and-revise loop is the central design decision of the pipeline: it makes the avatar itself the test of the encoding, so that no entry joins the dataset until it has been seen to render correctly.

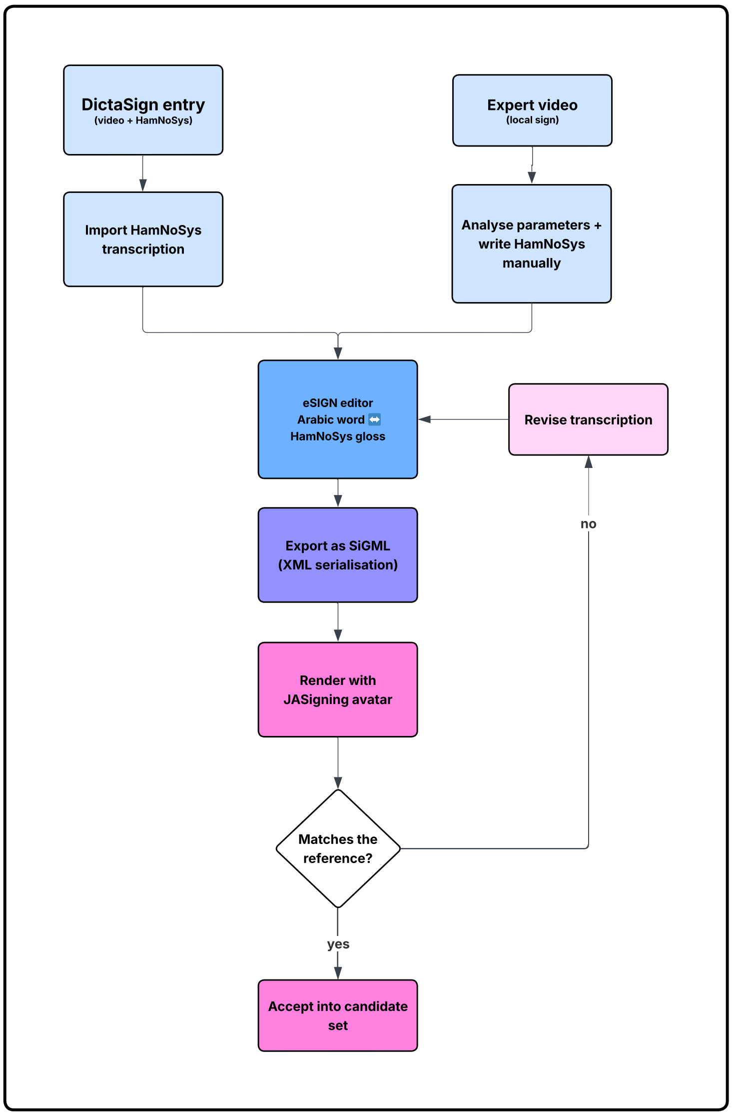

Figure 3.2: Design of the HamNoSys-to-SiGML conversion pipeline. Both acquisition routes converge on the eSIGN editor; every entry is verified by rendering it through the avatar before acceptance.

The two routes differ only in how the HamNoSys is obtained. On the import route, the transcription already exists and the work is to retrieve it, attach it to the Arabic headword, and adjust it where an expert reports that the Algerian sign departs from the French one. On the manual route, the transcription must be written from the video: the sign is analysed into the phonological parameters of Chapter 1 and expressed, parameter by parameter, in HamNoSys. The manual route is far more demanding, which is precisely why the design leans on the corpus for the bulk of the vocabulary and reserves manual encoding for the signs that have no other source. From the editor onward the two routes are indistinguishable, which is what keeps the dataset uniform.

3.4 Design of the Validation Methodology

A dictionary is only as trustworthy as the procedure that admits its entries. A single expert can make a mistake or apply a personal variant, and relying on one opinion would let such errors slip into the dictionary. The validation methodology is designed to prevent this.

For each candidate sign an expert records exactly one of four judgements: *correct*, *incorrect*, *almost* (right but needing a small modification), or *flagged* (a technical fault in the entry rather than a verdict on the sign). The four-way choice is itself a design decision: a plain accept/reject would force experts to discard a sign that is nearly right or to accept one with a technical defect, and the *almost* and *flag* options capture exactly those nuances. The independent judgements on a sign are then combined into a single status by the rule shown in Figure 3.3: three or more *incorrect* judgements reject it, three or more *almost* send it for revision, two or more flags mark it as flagged, and two or more *correct* validate it; when more than one of these conditions holds, the strictest outcome wins, so a contested sign is never validated on a thin margin. Requiring two independent *correct* judgements rather than one prevents a single expert's error from validating a sign, while remaining achievable across several hundred signs.

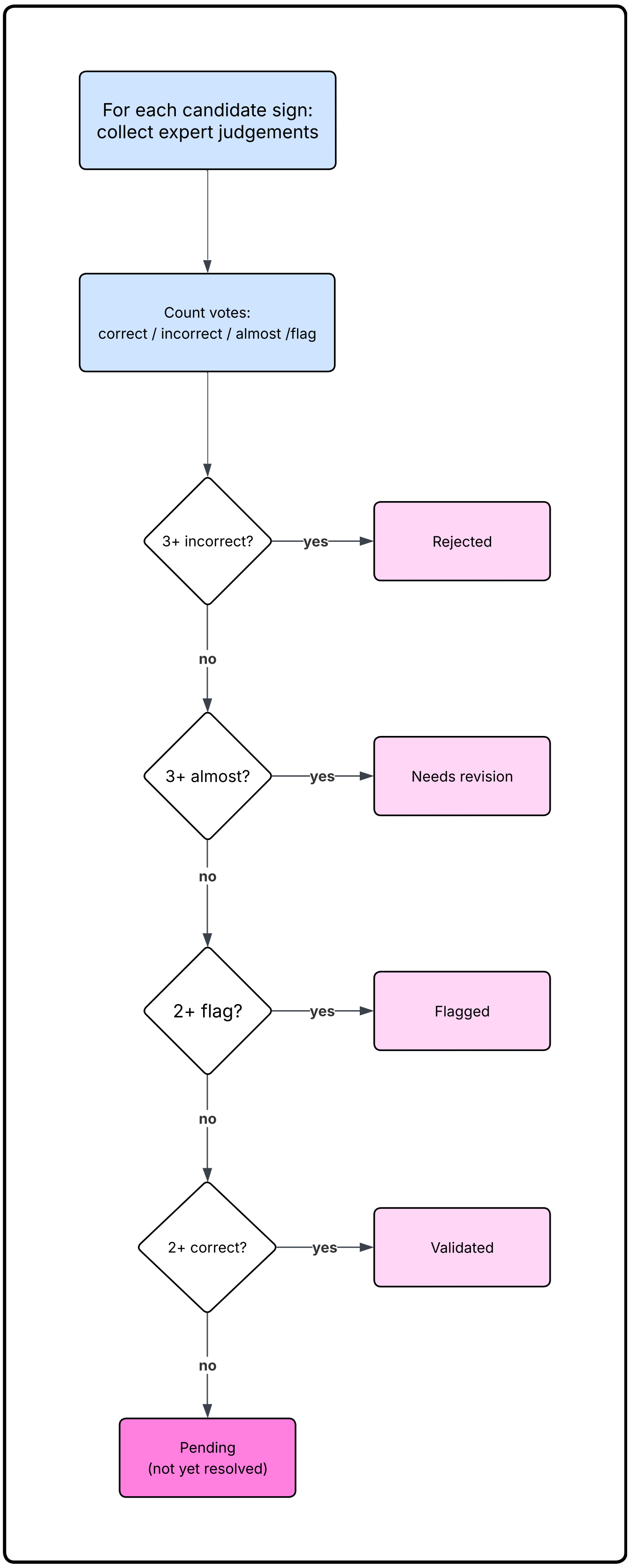

Figure 3.3: Design of the consensus validation logic. The checks are applied in order of strictness, and the strictest applicable status is the one assigned to the sign.

A sign sent for revision . If an expert's comment notes that it stands for a meaning other than the one displayed, the sign goes back to the experts for a second look and joins the lexicon once its proper meaning is agreed.

3.5 Design of the Evaluation Platform

The platform is built to apply the validation methodology on a scale of several hundred signs and a panel of experts. Its design has two parts an architecture that separates the interface, the logic and the data, and a data model that records experts, judgements, and administrative decisions.

3.5.1 Architecture

The presentation tier is the interface through which experts evaluate signs and the administrator monitors the campaign. The application tier holds all the logic: authentication, recording judgements, computing each sign's consensus status by the rule of Figure 3.3, the administrative views, and the data export. The data tier provides persistent storage for the experts, their judgements, and any administrative decisions. Separating the three tiers keeps the evaluation logic in one place and allows the interface or the storage to change without disturbing the rest.

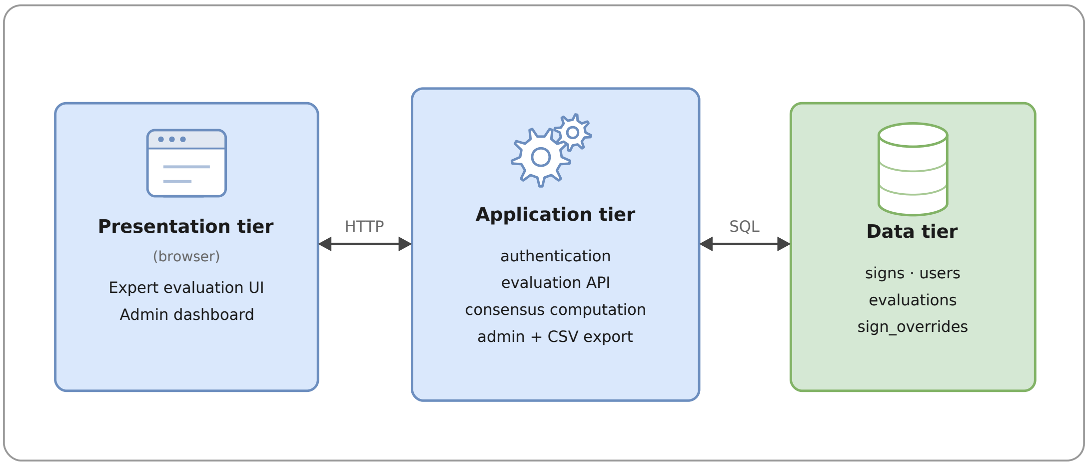

Figure 3.4: Architecture of the evaluation platform: a presentation tier, an application tier holding the logic, and a data tier.

3.5.2 Data Model

The data model is organised around four tables (Figure 3.5). The *signs* table holds one record per candidate sign, together with its concept and reference video. The *users* table holds one record per expert. The *evaluations* table holds one row per judgement, each linked to an expert and a sign, with a uniqueness constraint so that an expert has at most one judgement per sign, which remains revisable. The *sign_overrides* table records any manual administrative decision on a sign. A sign's consensus status is not kept as a fixed field but derived from the evaluations whenever it is needed, so it always reflects the latest judgements; where the rule cannot settle a sign automatically, the administrator can record an explicit status for it in the *sign_overrides* table, which is stored and can be edited at any time.

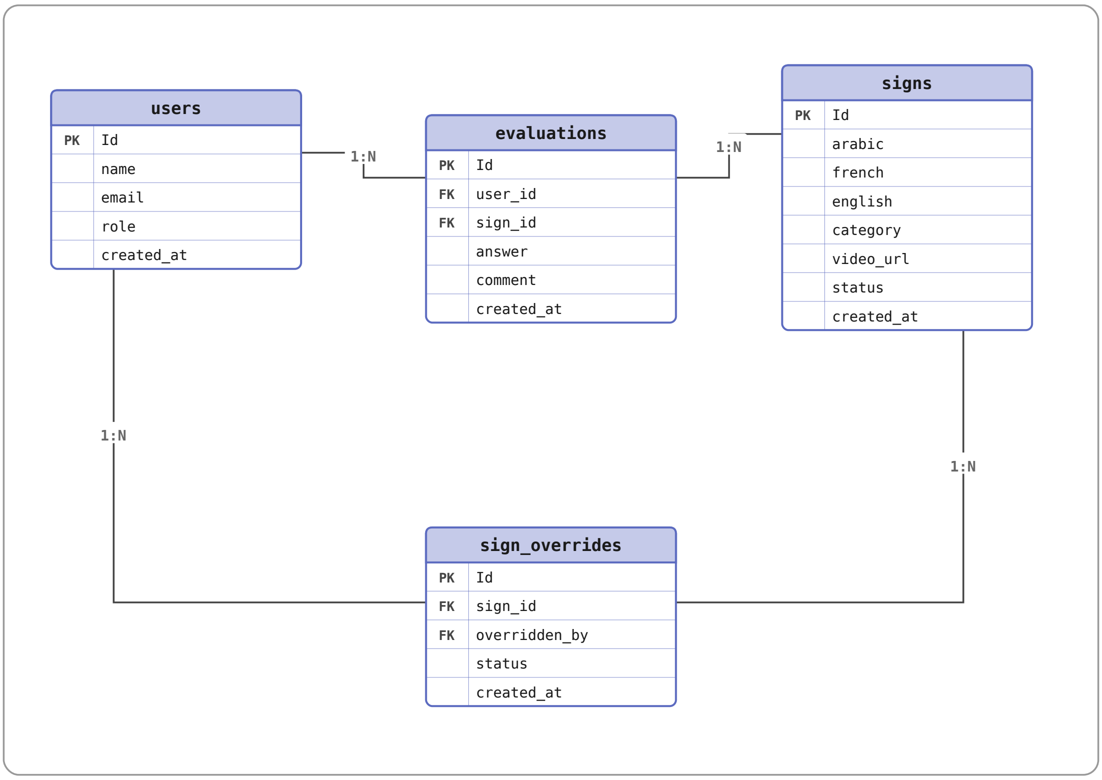

Figure 3.5: Data model of the platform: experts, their evaluations, and administrative overrides.

3.5.3 The Evaluation Workflow

The workflow is designed to fit the way the experts actually work rather than to impose a fixed order. After signing in, an expert selects a grammatical category, or all signs at once, and is led through the remaining signs one at a time. Each sign is shown as a concept in Arabic, French, and English together with its reference recording, and is judged with one of the four actions and an optional comment. A progress indicator, overall and per category, lets an expert stop after a few signs and resume later without losing their place. The trilingual presentation, the per-category progress, and the resumable session are all responses to the constraint discussed in the next section: expert time is scarce, so the design must let it be spent in short, interruptible sessions.

3.6 Designing for the Expert Community

This work was carried out together with qualified Algerian Sign Language experts, reached mainly through deaf associations who helped put us in contact with them. The experts took part in the evaluation campaign, each handling a portion of the candidate signs and working through them at their own pace, until a substantial number of signs had been reviewed and validated.

The design responds to this at two points. First, the validation methodology shares the work: each expert reviews a portion of the vocabulary, and when two or more of them judge the same sign, their answers are compared to decide its status. Second, the platform is built for low-friction, asynchronous use: nothing to install, any browser, any order, in short sessions that can be paused and resumed.

3.7 Conclusion

This chapter has set out the design of the system. It fixed an overall pipeline from two sign sources to an avatar-ready SiGML dataset; designed the HamNoSys-to-SiGML conversion around a single common format and an avatar-based verify-and-revise loop; designed a consensus validation methodology with four judgements, ordered thresholds, and a computed status; and designed a three-tier platform, with its data model, to apply that methodology. Throughout, the difficulty of engaging sign-language experts was the constraint that shaped the choices. The next chapter reports how this design was implemented and used in practice, and presents the results of the evaluation campaign.

Chapter 4

Implementation and Results

4.1 Introduction

The problem this thesis addresses is the scarcity of a validated, reusable digital lexicon for Algerian Sign Language. ALSL is not fully standardised, it has almost none of the digital resources, annotated datasets, or text-to-sign tools available for better-supported languages. Without a structured dictionary of signs encoded in a machine-readable form, no avatar can reliably translate written Arabic into Algerian Sign Language. This chapter reports how we built such a resource and confirmed it with the Deaf community.

The general idea of the implementation is simple: to produce a validated dataset of Algerian Sign Language signs in SiGML form. Such a dataset gives each sign a precise, machine-readable description, so that avatar generation becomes a matter of reading an entry rather than animating a sign by hand. Two routes produced these entries: most signs were imported from the DictaSign corpus, while signs absent from it, in particular Algerian-specific vocabulary, were encoded manually in HamNoSys from expert reference videos, the most demanding part of the work. The sections below present the development tools, the platform as built, the work with the expert community, the evaluation campaign and its results, and the construction and rendering of the lexicon.

4.2 Development Tools

The system was built with a small set of established tools, chosen for reliability and ease of maintenance:

- **eSIGN editor:** composition and editing of HamNoSys transcriptions and their export to SiGML.

- **JASigning:** the existing avatar engine that interprets SiGML and renders the signs in the browser [17].
- **Node.js / Express:** the server side of the evaluation platform, handling authentication, judgement recording, and status computation.
- **SQLite:** a lightweight, file-based database storing experts, judgements, and overrides; sufficient for the scale of the campaign and trivial to back up.
- **HTML / CSS / JavaScript:** the browser interface used by the experts, requiring no installation on their side.
- **Python:** data preparation, cleaning, and statistical analysis of the collected judgements.

The encoding choices (HamNoSys and SiGML) were justified in Chapter 2 and are not restated here.

4.3 Implementation Steps

The construction of the dataset followed a sequence of concrete steps, presented in the subsections below, from the collection of source material to the final validated lexicon.

4.3.1 Data Sources

The candidate signs were drawn from two complementary sources.

The DictaSign LSF corpus. The first and main source is the DictaSign corpus of French Sign Language [44], which provides around 1,000 sign entries. This corpus was chosen for two reasons. First, ALSL is historically and lexically close to LSF (Chapter 1), so a large proportion of LSF signs are plausible candidates for ALSL, subject to expert confirmation. Second, the DictaSign entries already include HamNoSys transcriptions, which removes the need to encode these signs from scratch and reduces the manual workload considerably.

Figure 4.1: Example of a DictaSign corpus entry with its HamNoSys transcription.

Local ALSL expert videos. The second source covers the signs that are specific to Algerian usage or absent from the corpus. These were collected as video recordings performed by local ALSL experts, and each recorded sign was then encoded manually in HamNoSys.

Figure 4.2: Example frame from a locally recorded ALSL expert video.

4.3.2 The Validation Platform

Once the candidate signs were collected, they had to be judged by the experts. To make this possible at the scale of several hundred signs, we built a web-based validation platform whose goal is to ease the experts' work: an expert can evaluate signs from any browser, with no installation, in short sessions and in any order.

After signing in, the expert chooses a grammatical category (nouns, verbs, adjectives, adverbs, prepositions, pronouns, or all signs at once) and is guided through the remaining signs one by one. For each sign, the platform shows the concept in Arabic, French, and English, plays the reference recording, and offers the four judgement actions with an optional comment. A progress bar shows how much work remains overall and per category, so an expert can stop at any point and resume later without losing track (Figure 4.3).

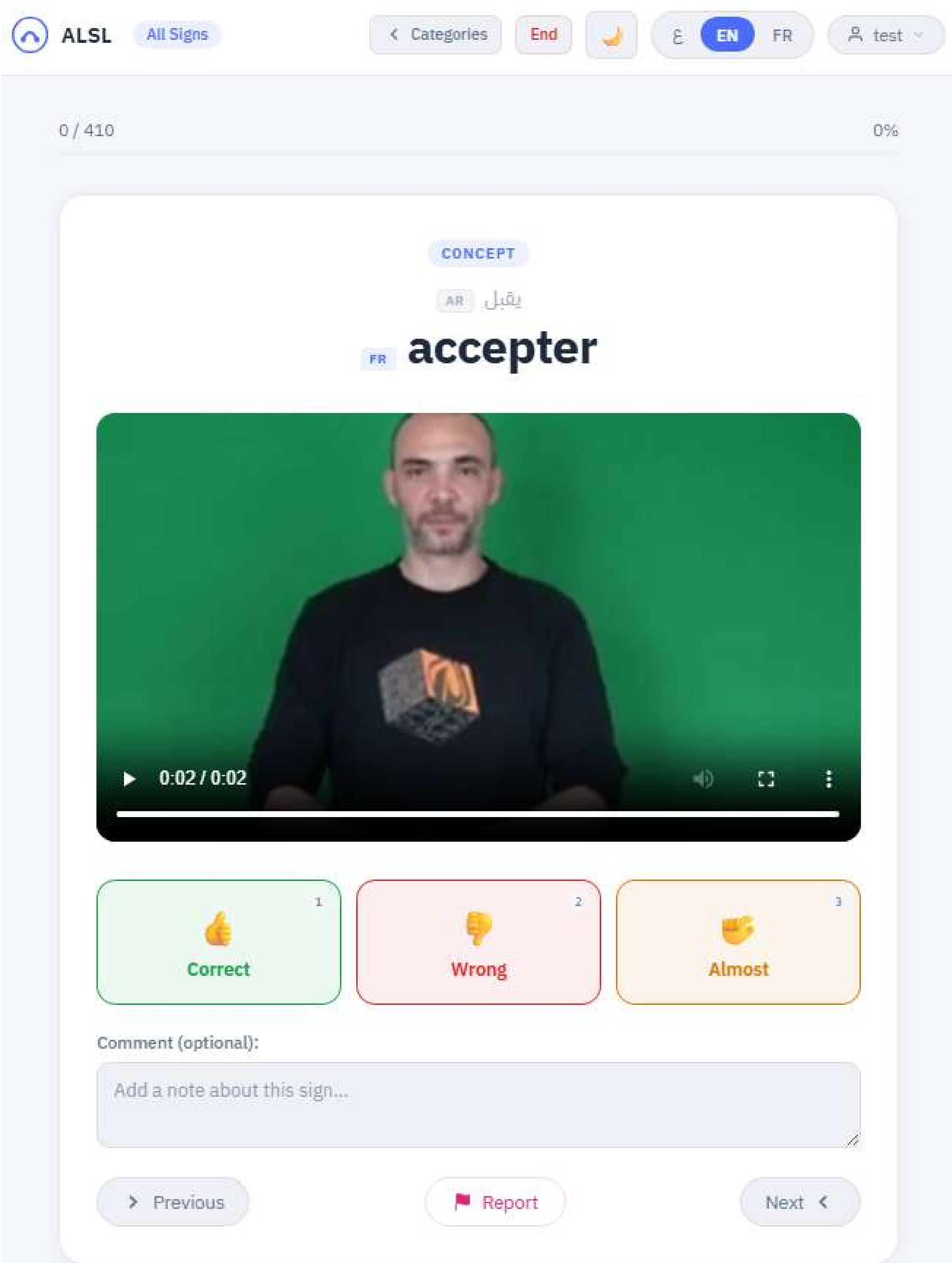

Figure 4.3: The expert interface: category selection with per-category progress tracking.

The administrator has a separate dashboard (Figure 4.4) that summarises the state of the campaign: the number of registered experts, the total evaluations submitted, and the count of signs in each consensus status (validated, rejected, needs revision, flagged), computed automatically by the rule defined in Chapter 3. The dashboard also exports the data as CSV files for analysis, encoded so that Arabic text displays correctly in standard spreadsheet tools.

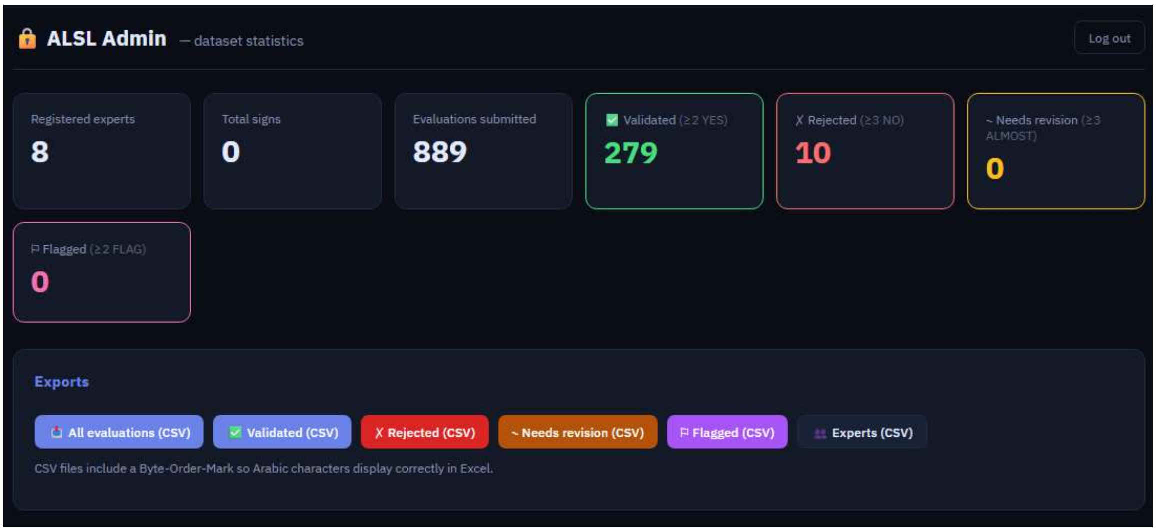

Figure 4.4: The administrative dashboard: aggregate statistics and CSV exports.

4.3.3 Working with the Expert Community

The most demanding part of the implementation was not the software but the human side: finding qualified ALSL experts and securing their participation. Such specialists are few, they are not concentrated in one institution, and their time is limited. Reaching them required going through deaf associations and personal introductions, explaining the purpose of the project, and building trust before any evaluation could begin. Sustaining their participation across a campaign of hundreds of signs then required repeated contact, reminders, and patience.

To make this dialogue possible, we also had to learn elements of Algerian Sign Language ourselves. Without some command of the signs, we could not understand an expert's objection to a candidate sign, describe a disputed sign precisely, or judge whether a proposed correction had been applied.

Each expert made a concrete and complementary difference. Rather than every expert reviewing every sign, the workload was distributed: each expert took on a portion of the vocabulary and worked through it at their own pace, and overlapping judgements on the same signs are what allowed the consensus rule to be applied. Sign by sign, these individual contributions accumulated until the full set of 410 candidate signs had been covered by the panel. Figure 4.5 illustrates the expert panel and their contributions to the campaign.

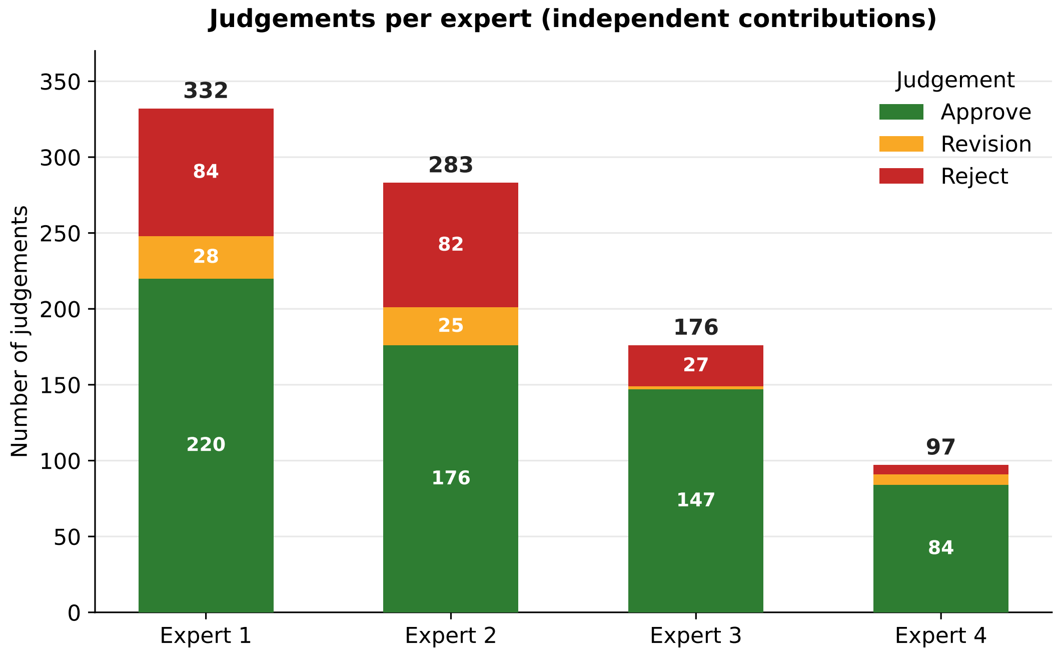

Figure 4.5: The expert panel and the contribution of each expert to the evaluation campaign.

Beyond evaluating, the experts shaped the platform itself. Their feedback led to the trilingual presentation of concepts, the addition of the ”almost” and ”flag” actions for nuanced judgements, the comment field for recording the reason behind a decision, and administrative tools such as password resets and returning a resolved sign to the queue. Each feature answered a real difficulty raised during the campaign.

4.3.4 Dataset Creation

The heart of the implementation is the creation of the dataset itself: producing, for each Arabic word, a HamNoSys transcription of its sign and the SiGML serialisation that the avatar engine consumes. The HamNoSys transcriptions were obtained by two methods: importing existing transcriptions from the DictaSign corpus, and encoding signs manually. This subsection covers the first method; manual encoding is described next.

Importing HamNoSys from DictaSign

For every concept present in the DictaSign corpus, the entry already provides a reference video of the sign and its HamNoSys transcription, as shown in Figure 4.6 for the concept *confortable*. The work in this case is not to encode the sign but to retrieve its transcription and link it to the equivalent Arabic word, so that each lexicon entry pairs an Arabic headword with the notation of its sign.

Figure 4.6: A DictaSign entry: reference video of the sign and its HamNoSys transcription.

The retrieved transcription is then taken into the eSIGN editor (Figure 4.7). The editor associates the Arabic word, entered as the spoken-language text, with a gloss row whose HamNoSys field holds the transcription; a dedicated dialog allows the HamNoSys symbols to be inserted and corrected. At this stage the imported transcription can also be adjusted when the experts have indicated that the Algerian sign differs slightly from the French one.

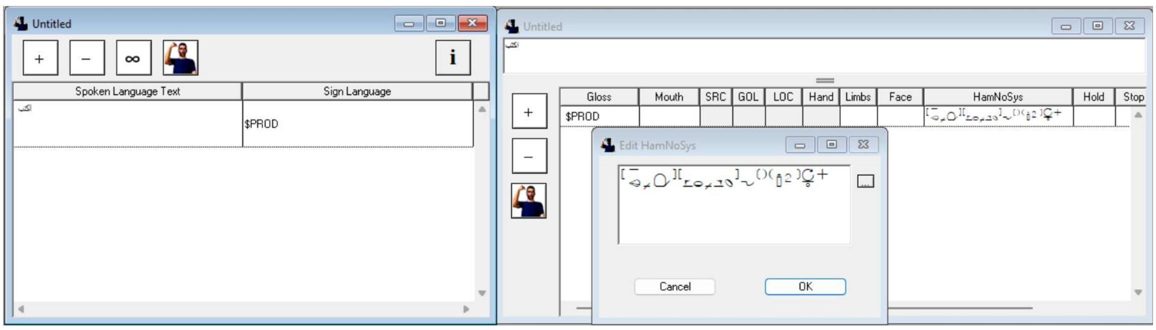

Figure 4.7: The eSIGN editor: the Arabic word linked to its gloss and HamNoSys transcription, with the HamNoSys editing dialog open.

From the editor, the entry is exported as SiGML (Figure 4.8). The exported document wraps the sign in a `hns_sign` element whose `hamnosys_manual` block lists, tag by tag, the handshape, orientation, location, and movement symbols of the transcription, while a `hamnosys_nonmanual` block is reserved for non-manual features. This XML form is exactly what the lexicon stores.

```
<sigml>

        <hns_sign gloss="$PROD">
                <hamnosys_nonmanual>
                </hamnosys_nonmanual>
                <hamnosys_manual>
                        <hamparbegin/>
                        <hampinch12/>
                        <hamfingerstraightmod/>
                        <hamthumbacrossmod/>
                        <hamplus/>
                        <hamflathand/>
                        <hamthumboutmod/>
                        <hamparend/>
                        <hamparbegin/>
                        <hamextfingerol/>
                        <hampalmd/>
                        <hamplus/>
                        <hamextfingeror/>
                        <hampalmur/>
                        <hamparend/>
                        <hampalm/>
                        <hamseqbegin/>
                        <hamclose/>
                        <hamfingertip/>
                        <hamindexfinger/>
                        <hamseqend/>
                        <hamcircled/>
                        <hamsmallmod/>
                        <hamrepeatfromstart/>
                </hamnosys_manual>
        </hns_sign>

</sigml>
```

Figure 4.8: The SiGML export of the sign: the HamNoSys symbols serialised as XML elements.

Finally, the SiGML is verified by playing it through the JASigning avatar (Figure 4.9). If the rendered movement matches the reference video, the entry is added to the validated dataset; otherwise the transcription is revised in the editor and re-exported. This closes the loop from corpus entry to avatar-ready description.

Figure 4.9: The JASigning avatar rendering the exported SiGML, used to verify the encoding.

Manual HamNoSys Encoding

The second method applies to the signs drawn from the local data source, those with no entry in DictaSign. Many of these are concepts specific to Algerian usage, including religious and cultural vocabulary that a French corpus does not cover. For each such sign, we watched the expert's reference video, analysed it into its phonological parameters (Chapter 1), and wrote the HamNoSys transcription ourselves, symbol by symbol, in the eSIGN editor. This is the most labour-intensive part of dataset creation, since every parameter of the sign must be identified by eye and expressed in notation.

The editor supports this task with a symbol palette (Figure 4.10) that organises the full HamNoSys inventory into tabs mirroring the structure of a transcription: handshapes, orientations, locations, two tabs of movements, and the two-handed operators. Encoding a sign means selecting, in order, the symbols that describe what the video shows.

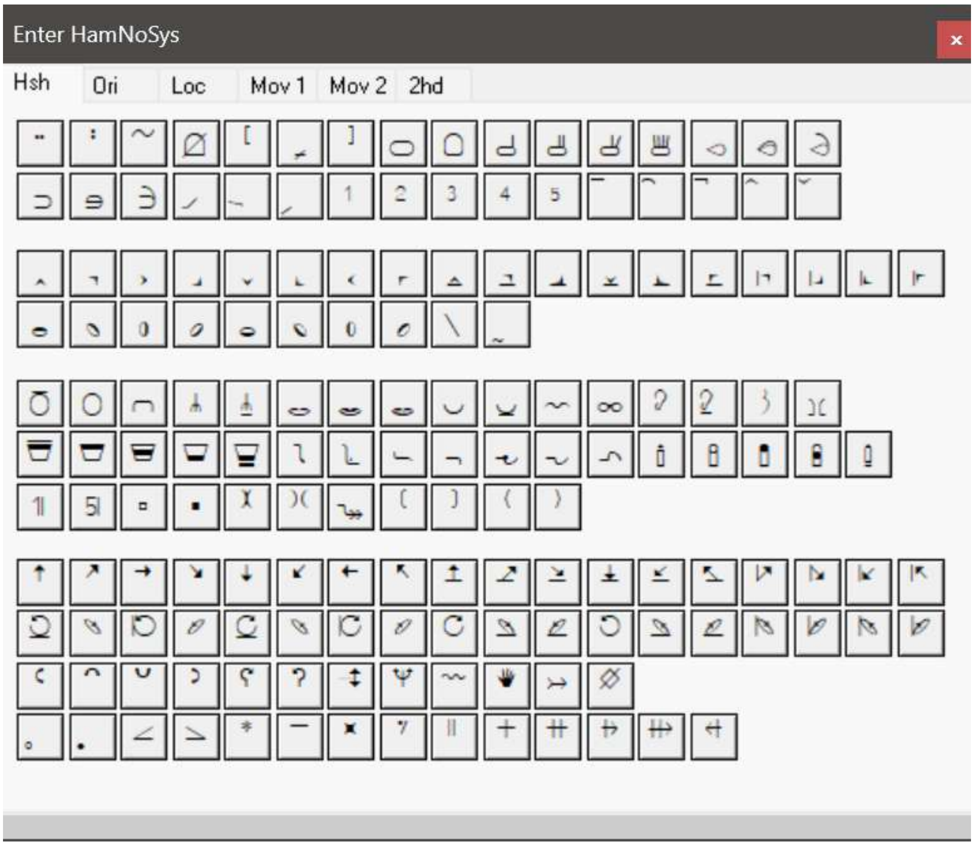

Figure 4.10: The HamNoSys symbol palette of the eSIGN editor, organised by parameter: handshape, orientation, location, movement, and two-handed operators.

Figures 4.11 to 4.13 follow the encoding of one local word from start to finish. The first step (Figure 4.11) creates the entry: the Arabic word is entered as the spoken-language text and a production gloss is attached to it, still without any notation.

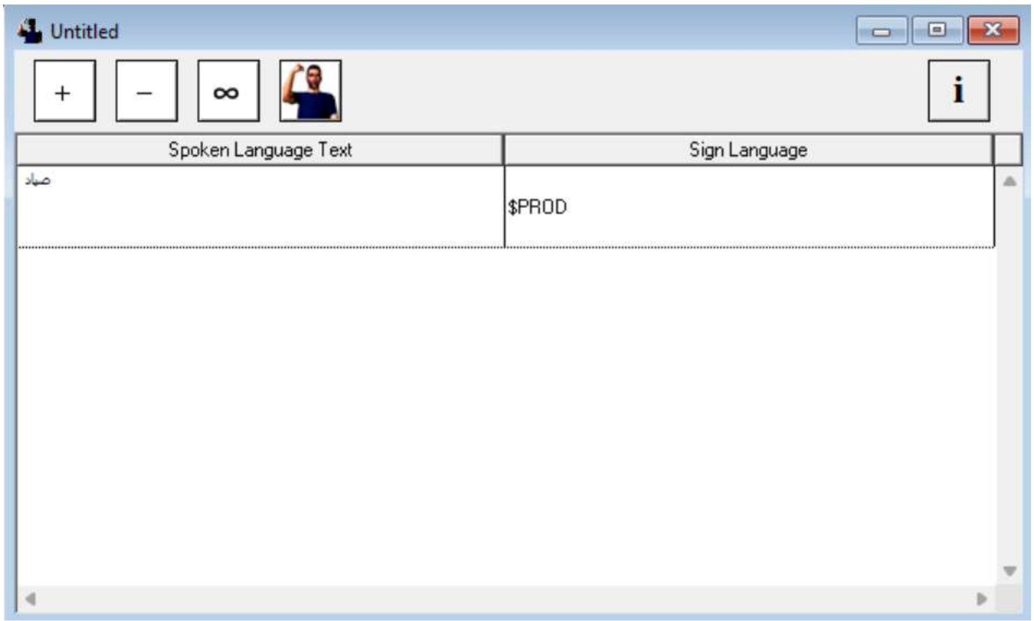

Figure 4.11: Manual encoding, first step: the Arabic word entered and linked to an empty gloss.

In the second step (Figure 4.12), the transcription is built up in the gloss row: the HamNoSys field is filled progressively with the symbols chosen from the palette, alongside the fields reserved for non-manual components such as the mouth and face.

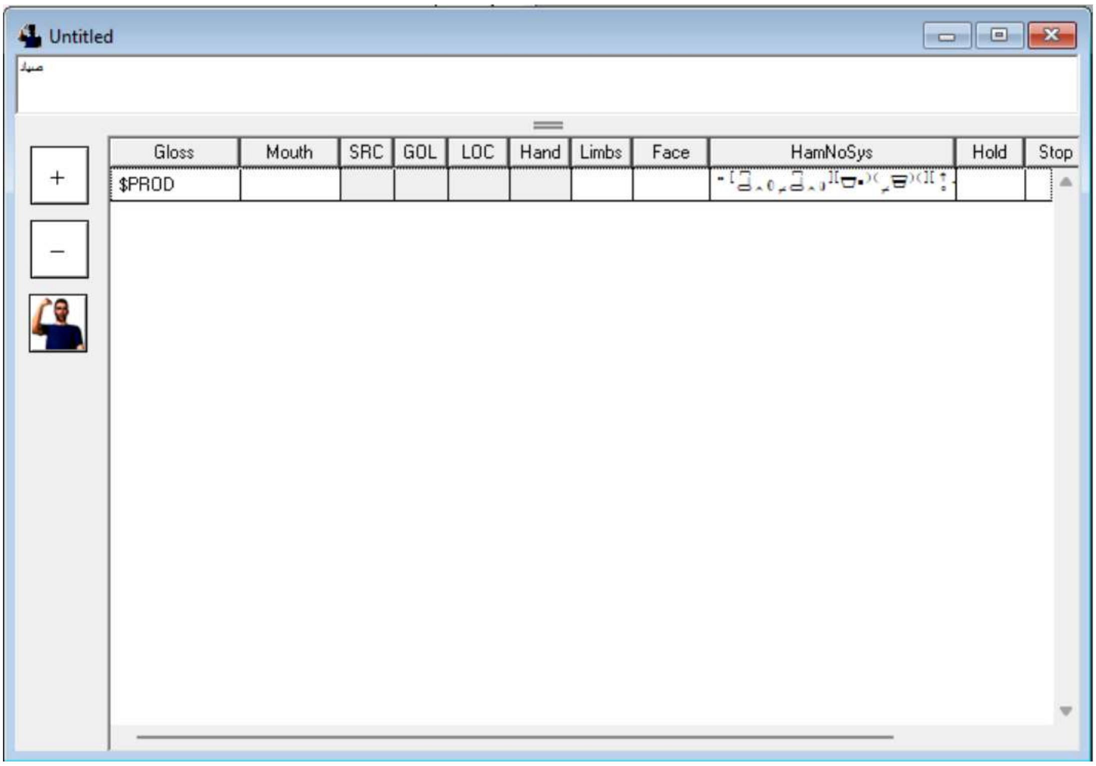

Figure 4.12: Manual encoding, second step: the HamNoSys field of the gloss filled in symbol by symbol.

The completed transcription is shown in the editing dialog (Figure 4.13): the pure HamNoSys string of the sign, reading in the fixed order of the notation from the symmetry and handshape symbols through orientation and location to the movement. From this point the entry follows the same path as the imported ones: it is exported as SiGML and verified through the avatar.

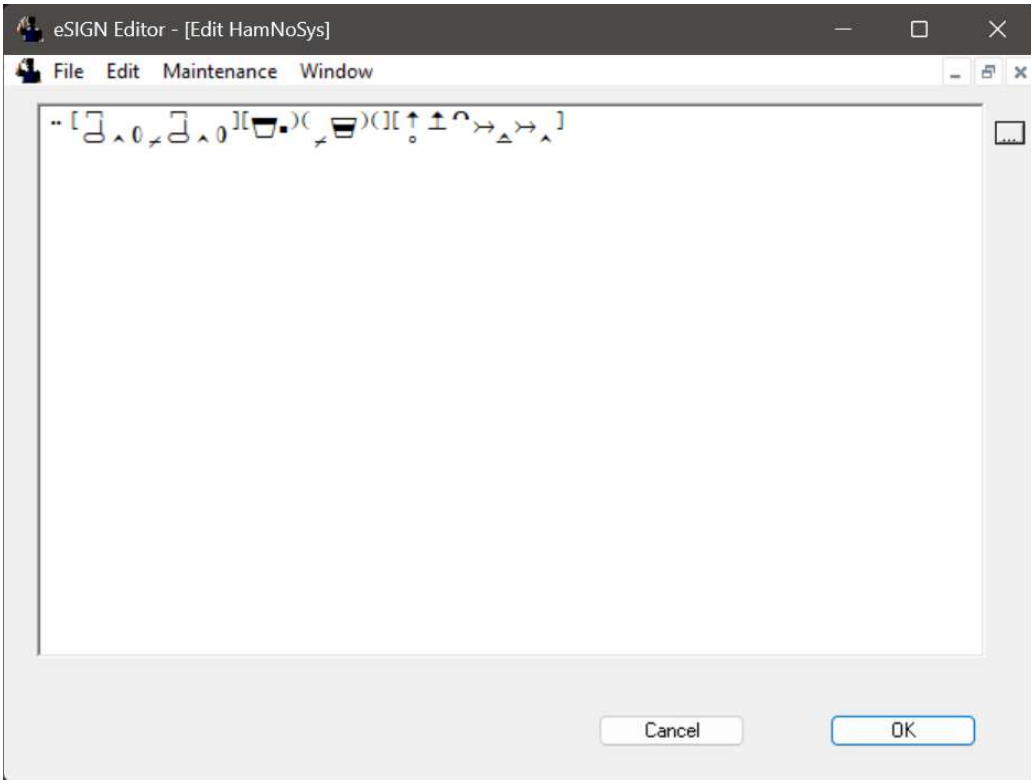

Figure 4.13: The complete HamNoSys transcription of the sign الصياد.

The same procedure was applied to the rest of the local vocabulary. Figures 4.14–4.16 show three further examples of manually encoded signs, each with its Arabic headword and its finished transcription in the editor.

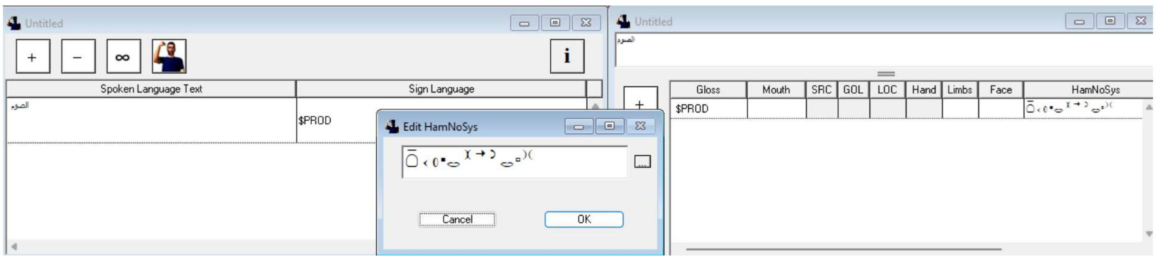

Figure 4.14: Manually encoded sign of the word الصوم.

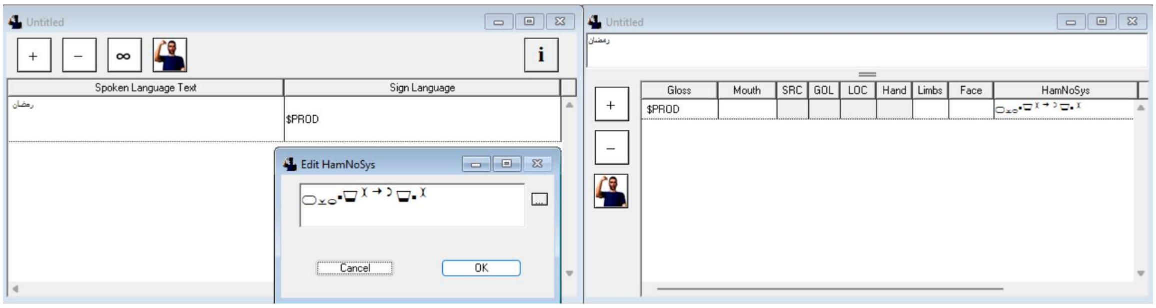

Figure 4.15: Manually encoded sign رمضان.

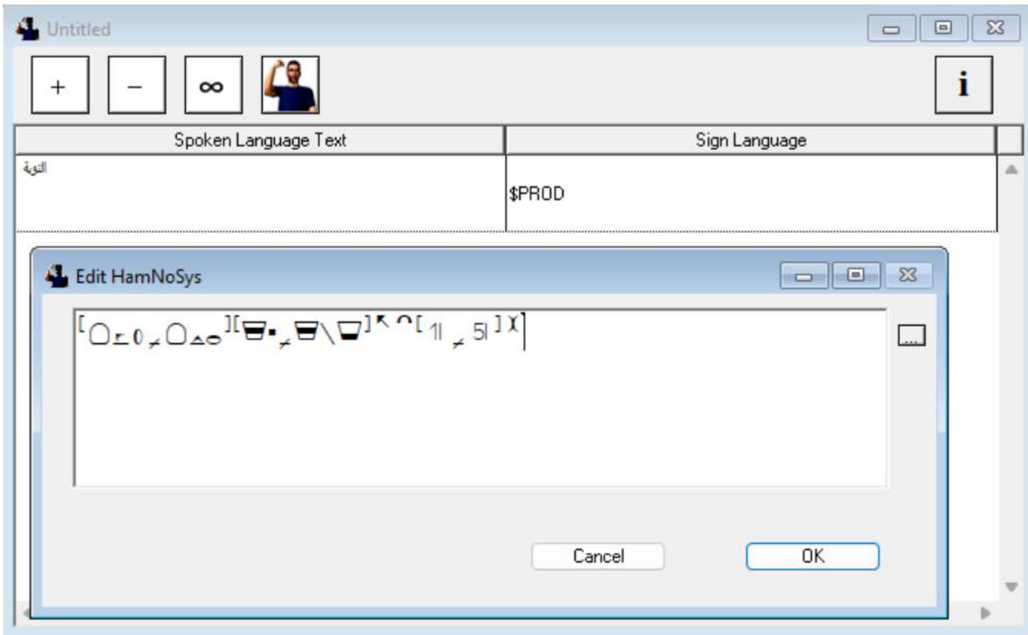

Figure 4.16: Manually encoded sign التوبة.

Conversion from HamNoSys to SiGML

Whichever method produced the HamNoSys transcription, the entry must finally be converted into SiGML, since the notation itself cannot be animated: only SiGML can be executed by the CWASA SiGML player that drives the JASigning avatar. The conversion proceeds in two steps.

SiGML generation. Once the transcription is complete and saved in the eSIGN editor, the document is exported through the editor's *Export SiGML* command (Figure 4.17). The export serialises every gloss of the document automatically: no symbol has to be converted by hand. Figure 4.18 shows the result for one word: a `sigml` document containing one `hns_sign` element per sign, whose `hamnosys_manual` block lists the transcription's symbols, in order, as XML tags.

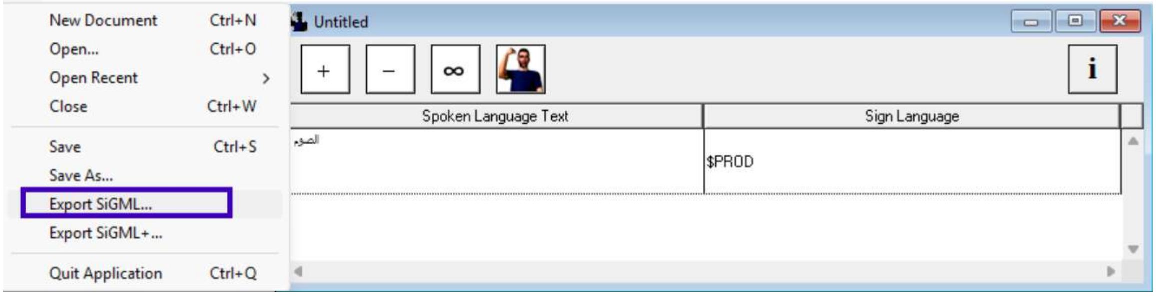

Figure 4.17: Generating the SiGML: the Export SiGML command of the eSIGN editor.

```
<sigml>

        <hns_sign gloss="$PROD">
                <hamnosys_nonmanual>
                </hamnosys_nonmanual>
                <hamnosys_manual>
                        <hamflathand/>
                        <hamfingerstraightmod/>
                        <hamextfingerl/>
                        <hampalml/>
                        <hamlrat/>
                        <hamlips/>
                        <hamtouch/>
                        <hammover/>
                        <hamarcr/>
                        <hamlips/>
                        <hamlrbeside/>
                        <hamclose/>
                </hamnosys_manual>
        </hns_sign>

</sigml>
```

Figure 4.18: The exported SiGML of the sign: each HamNoSys symbol serialised as an XML element.

SiGML playing. The exported file is then loaded into the CWASA SiGML player, which interprets the elements and animates the JASigning avatar accordingly, reproducing the handshape, orientation, location, and movement encoded in the notation. The player runs in an ordinary web browser, so each new entry can be checked immediately after export. Playing the SiGML serves as the final verification of the encoding:if the rendered sign matches the reference video, the entry is added to the validated dataset; if not, the transcription is corrected in the editor and re-exported.

The CWASA SiGML Player

CWASA (Client Web Avatar Signing Architecture) is the browser-based packaging of the JASigning avatar engine developed at the University of East Anglia within the ViSiCAST/eSIGN line of work [38, 39]. It loads a SiGML document, from a URL or pasted as text, and animates a virtual signer in real time (Figure 4.19).

Its installation and configuration are light:

- **Distribution:** a package of JavaScript libraries and avatar resources, embedded in an ordinary HTML page; no installation of dedicated software [38].
- **Requirements:** a modern browser with WebGL support, and an HTTP server to serve the package files, since browsers do not load the avatar resources from the local file system.
- **Settings:** choice of avatar character, playback speed control, and frame-by-frame stepping, useful for comparing a rendered sign against its reference video.
- **Our setup:** the package is served by the same Node.js server as the platform, so signs render on any device with a browser.

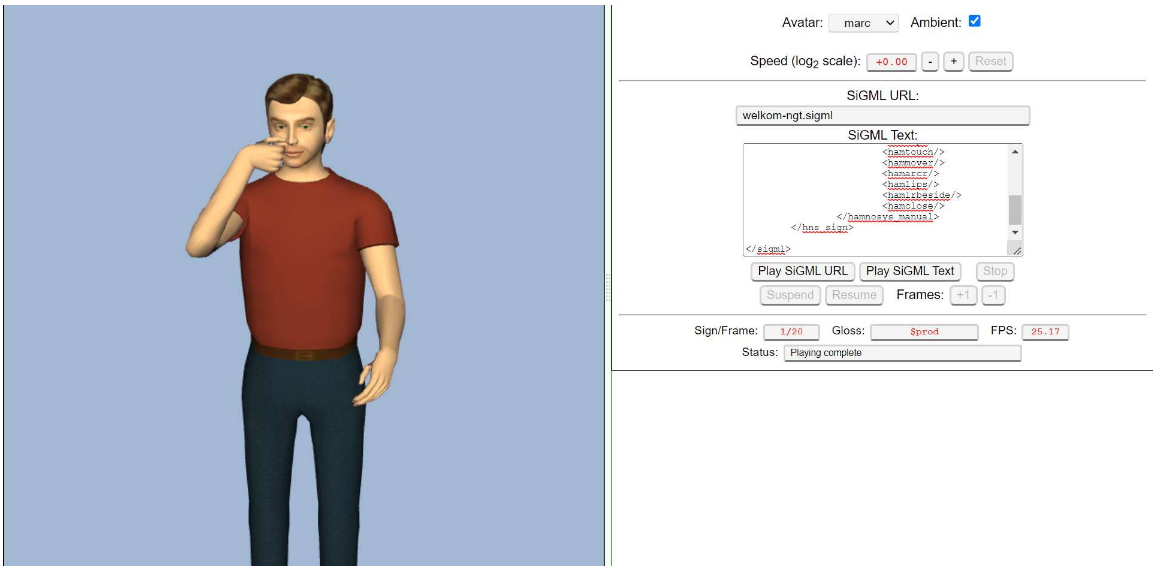

Figure 4.19: The CWASA SiGML player rendering a sign from a SiGML document.

4.4 The Evaluation Campaign and Results

The campaign covered 410 unique candidate signs. Five experts contributed judgements through the platform, submitting 889 independent evaluations in total. Participation was deliberately uneven, reflecting each expert's availability: the two most active experts contributed 332 and 283 judgements respectively, the others 176, 97, and 1, and it is this overlap of independent judgements on the same signs that allowed the consensus rule of Chapter 3 to operate. Across all judgements, 628 were "correct", 199 "incorrect", and 62 "almost".

Applying the consensus rule produced the outcome shown in Figure 4.20. Of the 410 candidates, 279 signs (68.0%) were validated directly by two or more "correct" votes, and 10 (2.4%) were rejected. A further 30 signs (7.3%) received two positive judgements of which at least one was "almost": for these, the corrections recorded in the experts' comments were applied to the transcriptions, and the corrected entries were incorporated into the lexicon, bringing the total of validated entries to 309 (75.4%). The remaining 91 signs (22.2%) were still pending further judgement at the close of the campaign.

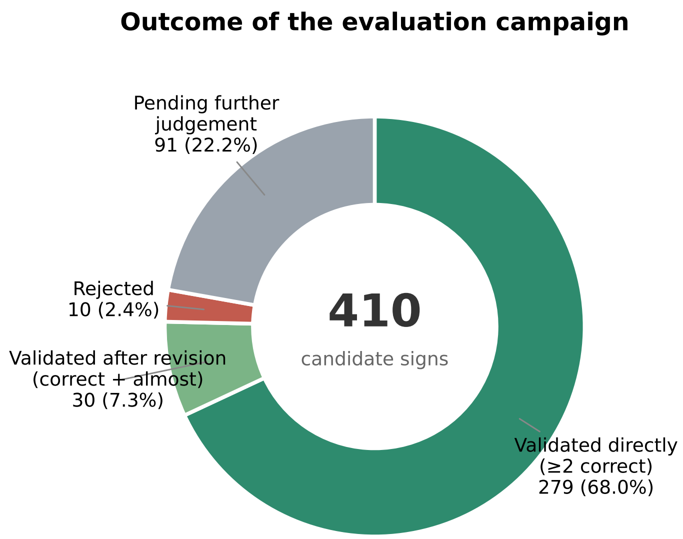

Figure 4.20: Outcome of the evaluation campaign over the 410 candidate signs.

The "almost" comments proved their worth in practice: in most cases the expert indicated that the sign was correctly formed but mapped to a slightly different concept (for example, a sign proposed for *truck* identified as the sign for *bus*, or one proposed for *road* identified as *river*), which is precisely the nuance the judgement was designed to capture and which a blunt accept-or-reject vote would have lost.

Figure 4.21 shows the distribution of the validated lexicon across grammatical categories, confirming that the dictionary covers everyday vocabulary across word classes rather than a narrow domain.

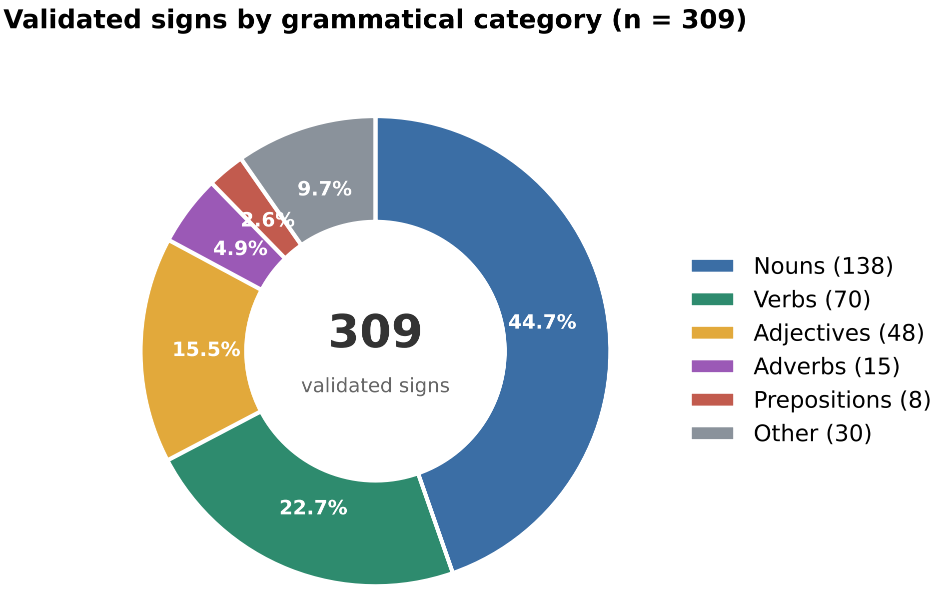

Figure 4.21: Distribution of the validated lexicon across grammatical categories.

4.4.1 Coverage of the National Dictionary

The dataset built here does not stand alone: it extends the prior 3DZSignDB work, which encoded 418 ALSL signs in HamNoSys/SiGML [17]. Together with the 309 signs validated in this thesis, the combined avatar-ready lexicon reaches 727 signs. Against the roughly 1,500 signs of the official national dictionary , this covers close to half of the standardised vocabulary (Figure 4.22) — a substantial step for a language that began with almost no digital resources, and one the additive dataset design (Chapter 3) lets future work extend without disturbing existing entries.

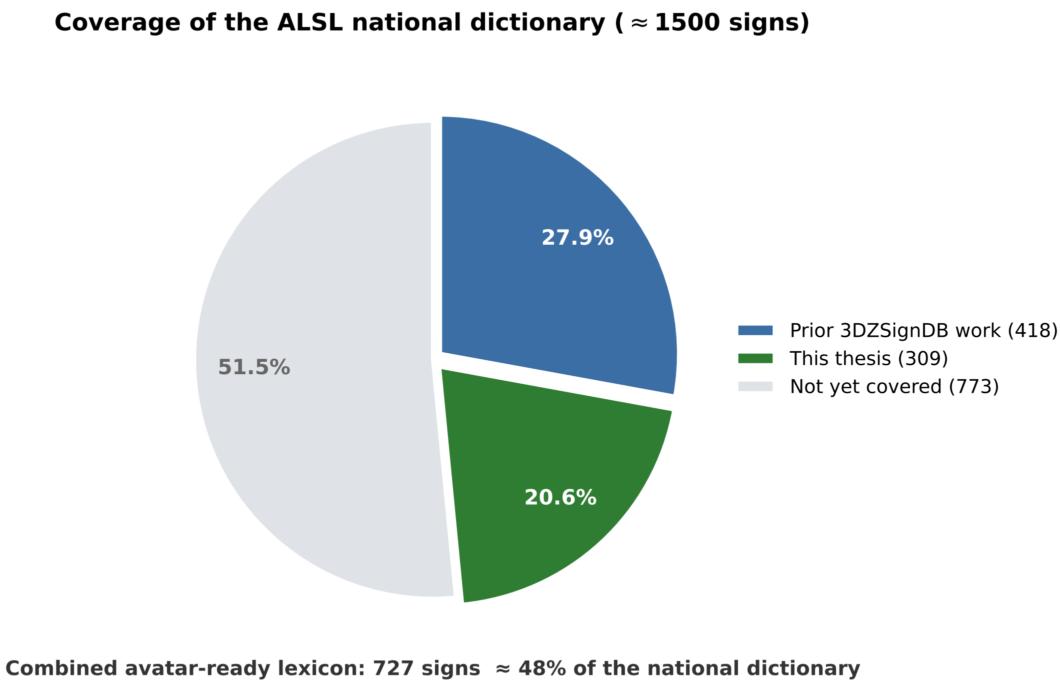

Figure 4.22: Coverage of the ALSL national dictionary: the combined contribution of the prior 3DZSignDB work (418) and this thesis (309).

4.5 Future Work

The contribution of this thesis is a foundation, and several directions would extend it naturally:

- **Expansion of the ALSL dataset and annotation quality.** The lexicon should grow beyond its current vocabulary toward broader everyday and domain-specific coverage (education, healthcare, administration), starting with the completion of the 91 signs left pending. Annotation quality can be raised in parallel: enriching entries with the non-manual components that current transcriptions only partially capture, and involving more experts so that each sign accumulates more independent judgements [15].
- **Automatic video-to-HamNoSys conversion.** Manual encoding proved to be the most labour-intensive step of dataset creation. A tool that analyses a sign video and proposes a HamNoSys transcription, even a draft one to be corrected by hand, would greatly simplify the transition from recorded video datasets to avatar-ready notation, and would make scaling the lexicon to thousands of signs realistic.
- **A modern 3D animation model.** The JASigning avatar is reliable but visually dated. Rebuilding the rendering stage on current technologies, such as modern web-

based 3D engines and data-driven motion synthesis learned from recorded signing, would improve both the visual quality and the smoothness of the animation while keeping the SiGML lexicon unchanged as input [14].

Each of these directions builds directly on the validated dataset this thesis contributes, which was designed from the start to be extended.

4.6 Conclusion

This chapter reported the realisation of the design set out in Chapter 3. Candidate signs were gathered from two sources, the DictaSign LSF corpus and locally recorded ALSL expert videos, and each was encoded in HamNoSys, either by importing an existing transcription or by writing it manually in the eSIGN editor, before being exported as SiGML and verified through the CWASA player. A web-based evaluation platform put the consensus methodology into practice, and a panel of experts, whose recruitment and sustained involvement were the hardest part of the work, submitted 889 judgements over 410 candidate signs. The campaign yielded 309 validated entries, 279 directly and 30 after applying the corrections recorded by the experts, with only 10 rejections. The result is the resource this thesis set out to build: a validated, avatar-ready SiGML lexicon of Algerian Sign Language, together with the platform and methodology to extend it, and the future directions identified above all build upon it.

General Conclusion

This thesis set out to address the scarcity of digital resources for Arabic–Algerian Sign Language by building a standardised, expert-validated lexicon and using it to drive a three-dimensional signing avatar. The theoretical chapters established that signs decompose into well-defined phonological parameters and that these parameters can be written in HamNoSys, serialised as SiGML, and rendered by an avatar engine. The conception chapter turned this into a design: a consensus-based validation methodology, a web platform to apply it, and a uniformly encoded SiGML lexicon feeding the existing rendering system. The implementation chapter carried that design through in practice.

The principal contribution is the dataset itself. Candidate signs were drawn from two complementary sources, the DictaSign LSF corpus and locally recorded ALSL expert videos, and encoded in HamNoSys either by import or by hand, before being serialised as SiGML and verified through the avatar. A panel of experts, recruited and sustained through considerable effort, submitted 889 judgements over 410 candidate signs, yielding 309 validated entries. Combined with the 418 signs of the prior 3DZSignDB work, this brings the avatar-ready ALSL lexicon to 727 signs, close to half of the roughly 1,500 signs in the national dictionary. Alongside the dataset, the work delivered a reusable validation methodology and a deployed evaluation platform, together demonstrating a practical path from a related sign-language corpus to an authentic, avatar-ready Algerian Sign Language resource.

Several directions would extend this contribution naturally:

- **Dataset expansion and annotation quality.** Grow the lexicon toward broader everyday and domain-specific vocabulary, starting with the pending signs, and enrich non-manual components with a wider expert panel [15].

- **Automatic video-to-HamNoSys conversion.** A tool that drafts a HamNoSys transcription from a sign video would remove the main bottleneck of manual encoding and make scaling to thousands of signs realistic.

- **A modern 3D animation model.** Rebuilding the rendering stage on current web-based 3D engines and data-driven motion synthesis would improve visual quality and naturalness, while keeping the SiGML lexicon unchanged as input [14].

Each of these builds directly on the validated dataset this thesis contributes, which was designed from the outset to be extended.

Bibliography

[1] Stokoe, W. C. (1960). *Sign Language Structure*. University of Buffalo.

[2] J. Forster, C. Schmidt, T. Hoyoux, O. Koller, U. Zelle, J. H. Piater, and H. Ney, "RWTH-PHOENIX-weather: A large vocabulary sign language recognition and translation corpus," in *Proc. 8th Int. Conf. Language Resources and Evaluation (LREC)*, Istanbul, Turkey, May 2012, pp. 3785–3789.

[3] G. NC, M. Ladi, S. Negi, P. Selvaraj, P. Kumar, and M. Khapra, "Addressing resource scarcity across sign languages with multilingual pretraining and unified-vocabulary datasets," *Advances in Neural Information Processing Systems*, vol. 35, pp. 36202–36215, 2022.

[4] Sandler, W., & Lillo-Martin, D. (2006). *Sign Language and Linguistic Universals*. Cambridge University Press.

[5] Brentari, D. (2010). *Sign Languages*. Cambridge University Press.

[6] Pfau, R., Steinbach, M., & Woll, B. (2012). *Sign Language: An International Handbook*. De Gruyter Mouton.

[7] Lakhfif, A. (2017). Études linguistiques sur la Langue des Signes Algérienne. Université d'Alger.

[8] Camgöz, N. C., et al. (2018). Neural sign language translation. *Proceedings of CVPR.*

[9] M. A. Abdel-Fattah, "Arabic Sign Language: A perspective," *Journal of Deaf Studies and Deaf Education*, 2005.

[10] A. Djama, "Comparaison entre ALSL et LSF," Master thesis, Aix-Marseille Univ., 2016.

[11] M. S. Mansour, "Langage et surdité," Univ. of Oran, 2007.

[12] F. Nekkaa, "Détection automatique de la main," Univ. Constantine 2, 2015.

[13] Koller, O. (2019). Quantitative survey of sign language recognition. *Computer Vision and Image Understanding.*

[14] L. Dong, L. Chaudhary, F. Xu, X. Wang, M. Lary, and I. Nwogu, "SignAvatar: Sign Language 3D Motion Reconstruction and Generation," in *2024 IEEE 18th International Conference on Automatic Face and Gesture Recognition (FG)*, Istanbul, Turkiye, 2024, pp. 1–10, doi: 10.1109/FG59268.2024.10581934.

[15] R. Wolfe, J. C. McDonald, and J. C. Schnepp, "Sign Language Avatars: A Question of Representation," *Information*, vol. 13, no. 4, p. 206, 2022, doi: 10.3390/info13040206.

[16] L. Bellil, M. G. Ghiri, and A. Boulesnane, "Empowering Deaf Community in Healthcare Communication: 1D-CNN-Based Algerian Sign Language Recognition System," in *Proc. 2024 6th Int. Conf. Pattern Analysis and Intelligent Systems (PAIS)*, El Oued, Algeria, 2024, pp. 1–7, doi: 10.1109/PAIS62114.2024.10541233.

[17] T. Zerrouki, M. F. Slimani, A. Mami, and R. Mazari, "3DZSignDB: 3D avatar SigML data for Algerian sign language," *Data in Brief*, vol. 60, p. 111568, 2025.

[18] M. Alayed, "Arabic Sign Language Recognition Using Deep Learning: A Systematic Review," *Sensors*, vol. 24, no. 23, Art. no. 7798, 2024, doi: 10.3390/s24237798.

[19] A. Boulesnane, L. Bellil, and M. G. Ghiri, "ALGSL89: An Algerian Sign Language Dataset," in *Proc. Int. Conf. Artificial Intelligence and Data Analytics*, 2023, pp. 1–6.

[20] K. Ahmed, I. Kouar, and E. Kouar, "Real Time System Based Deep Learning for Recognizing Algerian Sign Language," *Int. J. Comput. Digit. Syst.*, vol. 15, no. 1, pp. 713–722, Feb. 2024, doi: 10.12785/ijcds/150152.

[21] A. Bouchama and A. Chihaoui, "A Deep Learning Framework for Bilingual Sign Language Interpretation," Master's thesis, University 8 May 1945 – Guelma, 2025. [Online]. Available: https://dspace.univ-guelma.dz/jspui/bitstream/123456789/18254/1/F5_8_BOUCHAMA_AMEL.pdf

[22] C. Padden and D. C. Gunsauls, "How the alphabet came to be used in a sign language," *Sign Language Studies*, vol. 4, no. 1, pp. 10–33, 2003.

[23] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black, "Expressive Body Capture: 3D Hands, Face, and Body from a Single Image," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, June 2019, pp. 10975–10985, doi: 10.1109/CVPR.2019.01125.

[24] Z. Yu, J. Guo, S. Wang, Y. Li, S. Huang, M. Xu, and D. Tao, "SignAvatars: A Large-scale 3D Sign Language Holistic Motion Dataset and Benchmark," arXiv:2310.20436v3 [cs.CV], Oct. 2023. (ECCV 2024.)

[25] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "SMPL: A Skinned Multi-Person Linear Model," *ACM Transactions on Graphics*, vol. 34, no. 6, Art. no. 248, Oct. 2015, doi: 10.1145/2816795.2818013.

[26] Kang, Z., 2019. Spoken language to sign language translation system based on HamNoSys. In: *Proceedings of the 2019 International Symposium on Signal Processing Systems*, September 2019, pp. 159–164.

[27] L. Kavan, S. Collins, J. Žára, and C. O'Sullivan, "Geometric Skinning with Approximate Dual Quaternion Blending," *ACM Transactions on Graphics*, vol. 27, no. 4, Art. no. 105, 2008, doi: 10.1145/1409060.1409067.

[28] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis," in *Proc. European Conf. Computer Vision (ECCV)*, Glasgow, UK, Aug. 2020, pp. 405–421, Springer, doi: 10.1007/978-3-030-58552-8_24.

[29] P. Ekman and W. V. Friesen, *Facial Action Coding System (FACS): Investigator's Guide*, University of California Press, 2002. ISBN: 978-0931535027.

[30] N. Zhou, Y. Wang, and J. Li, "Real-Time Speech-to-Sign Language Translation Using 3D Avatars with End-to-End Optimization," *IEEE Access*, vol. 11, pp. 109875–109889, Oct. 2023, doi: 10.1109/ACCESS.2023.3321478.

[31] J. A. Smith, R. M. Johnson, and T. L. Brown, "Adaptive 3D Sign Language Avatar Tutors: A Randomized Controlled Trial with 387 Learners," *Journal of Sign Language and Technology*, vol. 7, no. 2, pp. 45–67, Apr. 2024, doi: 10.1007/s10587-024-00123-4.

[32] C. Boularaki, G. Margetis, and C. Stephanidis, "3D Avatar Technologies for Automated Sign Language Interpretation in Television Broadcasting," in M. Antona and C. Stephanidis (Eds.), *Universal Access in Human-Computer Interaction: Applications and Technologies*, HCII 2021, Lecture Notes in Computer Science, vol. 12769, Springer, 2021, pp. 234–249, doi: 10.1007/978-3-030-78095-1_16.

[33] VIVERSE Platform, "Social Virtual Reality Avatars for Cross-Language Communication Including Sign Language Support," VIVERSE Technical Report TR-VR-2023-001, Seoul, South Korea, Jun. 2023.

[34] T. Hanke, "HamNoSys — Representing Sign Languages in Language Resources and Language Processing Contexts," in *Proc. Workshop Representation and Processing of Sign Languages at LREC*, Lisbon, Portugal, May 2004, pp. 1–6, ELDA.

[35] R. Prillwitz, T. Hanke, D. Leven, R. Schwarz, and C. Sikora, *HamNoSys Version 3.0: Hamburg Notation System for Sign Languages — User's Guide*, Signum Press, Seedorf, Germany, 1996. ISBN: 978-3927734700.

[36] P. Jedlička, Z. Krňoul, and M. Železný, "Handling Sign Language Transcription System with the Computer-Friendly Numerical Multilabels," *arXiv preprint* arXiv:2204.06924, 2022.

[37] P. Jedlička, Z. Krňoul, J. Kanis, and M. Železný, "On the Importance of Sign Labeling: The Hamburg Sign Language Notation System Case Study," *arXiv preprint* arXiv:2302.10768, 2023.

[38] S. Elliott, J. Glauert, and R. Wales, "The JASigning Avatar System: Development and Applications," in *Proc. 4th Workshop on the Representation and Processing of Sign Languages*, Valletta, Malta, May 2010, pp. 75–82, ELRA.

[39] J. Colville, J. R. Glauert, and S. A. Elliott, "SiGML: Signing Gesture Markup Language for Avatar Animation," ViSiCAST Project Deliverable D4.2, University of East Anglia, Norwich, UK, 2002.

[40] S. Cheshire, J. R. Glauert, and R. Wales, "SiGML: Design Issues for the Representation of Sign Language Notations for Avatar Animation," ViSiCAST Project Technical Report, University of East Anglia, 2001.

[41] ViSiCAST Consortium, *ViSiCAST Final Project Report: Virtual Signing for Digital Television*, European Commission IST Project IST-1999-20520, Brussels, Belgium, Dec. 2004.

[42] M. Safoleas, A. Arfaras, and G. Vardakas, "ViSiCAST Evaluation Framework: User Trials and Comprehension Studies," ViSiCAST Project Deliverable D5.3, Athens Information Technology, 2005.

[43] L. Zhao, K. Kipper, W. Schuler, C. Vogler, M. Palmer, and D. Metaxas, "A Machine Translation System from English to American Sign Language," in *Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP)*, Philadelphia, PA, USA, Jul. 2000, pp. 224–231, ACL.

[44] S. Matthes, T. Hanke, A. Regen, J. Storz, S. Worseck, E. Efthimiou, A.-L. Dimou, A. Braffort, J. Glauert, and E. Safar, "Dicta-Sign – Building a Multilingual Sign Language Corpus," in *Proc. 5th Workshop on the Representation and Processing of Sign Languages (LREC)*, Istanbul, Turkey, 2012.

[45] Ministry of National Solidarity, Family and Women's Affairs (Algeria), *Dictionary of Algerian Sign Language*, Algiers, 2017.