Desenvolupament d'un Sistema de Reconeixement de Llengua de Signes

Javier Comes Tello

Resum— Aquest projecte té com a objectiu desenvolupar un sistema de reconeixement de llengua de signes en temps real mitjançant visió artificial i tècniques d'aprenentatge profund. El sistema és capaç d'identificar i traduir gestos de la *Llengua de Signes Americana (ASL)* a text amb l'objectiu de millorar la comunicació entre persones sordes i oients. La solució utilitza una arquitectura *Inflated 3D ConvNet (I3D)* preentrenada a *ImageNet* i optimitzada per classificar els signes utilitzant el conjunt de dades WLASL. El model processa entrades de vídeo RGB i combina prediccions a nivell de vídeo i de fotograma per millorar la precisió del reconeixement. Tot i que encara limitat per la variabilitat del conjunt de dades i el desequilibri de classes, el model va aconseguir ser capaç de detectar lletres, paraules i frases curtes en temps real.

Paraules clau—Visió artificial, aprenentatge profund, traduir gestos, Inflated 3D, WLASL

Abstract—This project aims to develop a real-time sign language recognition system using computer vision and deep learning techniques. The system is capable of identifying and translating *American Sign Language (ASL)* gestures into text, with the intention of improving communication between deaf and hearing people. The solution employs an *Inflated 3D ConvNet (I3D)* architecture pre-trained on *ImageNet* and optimized for sign classification using the *WLASL* dataset. The model processes RGB video inputs and combines video and frame-level predictions to improve recognition accuracy. Although still limited by dataset variability and class imbalance, the model was able to detect letters, words, and short phrases in real time.

Index Terms— Computer vision, deep learning, gesture recognition, Inception I3D, WLASL

1 Introducció - Context del treball

En el món existeixen nombroses llengües, algunes més parlades que d'altres. No obstant això, **la comunicació no es limita únicament a l'expressió oral**, sinó també a la comprensió mutua. Hi ha persones sordes que no poden escoltar, però que poden comunicar-se perfectament a través de la llengua de signes. Alhora, hi ha persones que no coneixen aquesta llengua però desitgen comunicar-se amb qui la utilitza. Fins a quin punt resulta dificil moure les mans? **Aprendre gestos no difereix tant d'aprendre una nova llengua**.

Una llengua de signes és una llengua natural que utilitza moviments del cos per expressar-se i es compren a través de la vista, gràcies a la qual les persones sordes poden establir un canal de comunicació amb el seu entorn social, sigui aquest conformat per altres persones sordes o per qualsevol persona que conegui la llengua de signes[1]. Contrariament al que es podria pensar, **no hi ha una llengua de signes universal**, sinó que cada país té la seva pròpia, igual que passa amb les llengües orals.

Aquesta forma de comunicació és, en si mateixa, un idioma complet, amb la seva pròpia gramàtica, estructura i regles sintàctiques. Les persones que la dominen com a llengua materna s'anomenen **signants**, és a dir, persones que utilitzen la llengua de signes com a mitjà principal de comunicació. D'altra banda, hi ha persones que aprenen aquesta llengua per necessitat o per interès, ja sigui per la seva interacció amb la comunitat signant o per a fins professionals. A més, existeixen intèrprets que faciliten la comunicació entre persones signants i no signants. Pero, encara hi ha barreres d'accessibilitat.

Tot i els avenços en l'àmbit de la traducció automàtica per a idiomes parlats, encara no existeixen sistemes equivalents que interpretin de manera fiable i precisa la llengua de signes. Aquesta falta de recursos tecnològics crea una dificultat d'accessibilitat important per a les persones signants, especialment en entorns on no es disposa d'intèrprets. Davant aquesta situació, es fa evident la necessitat de desenvolupar solucions capaces de reconèixer i traduir automàticament els gestos de la llengua de signes. Per això, aquest projecte sorgeix amb l'objectiu d'analitzar les contribucions existents en aquest camp, identificar les àrees de millora i dissenyar un prototip funcional que pugi mostrar el potencial real d'aquest tipus de tecnologia.

En aquest context, el meu projecte té com a objectiu principal desenvolupar un sistema capaç de **reconèixer i traduir en temps real la llengua de signes**, inicialment de **l'American Sign Language (ASL)** a text. L'enfocament prioritza el **reconeixement de gestos captats mitjançant una càmera**

- *E-mail de contacte: 1631153@uab.cat*
- *Menció realitzada: Enginyeria de Computació*
- *Treball tutoritzat per: Gemma Sanchez Albaladejo (Departament de Ciencies de la Computació)*
- *Curs 2024/25*

en directe, permetent una **traducció automàtica immediata** a través de tècniques avançades de visió per computador. Per assolir aquest objectiu, s'han utilitzat inicialment tècniques com **MediaPipe** per a la detecció de mans, **CNN** per al reconeixement d'imatges estàtiques i **xarxes LSTM** per identificar gestos amb moviment. Tot i obtenir resultats inicials utils, les limitacions en temps real han portat a evolucionar cap a models més potents com **Inception I3D,** capaç de captar la dinàmica espaciotemporal dels gestos mitjançant vídeos RGB del dataset WLASL[12].

Encara que sovint es pensa que una sola imatge estàtica pot ser suficient per reconèixer un signe, aquesta idea és errònia. La llengua de signes no és només una col·lecció de posicions fixes de les mans, sinó que inclou també el moviment i, en molts casos, l'expressió facial. Per exemple, certes lletres com "J" o "Z" en l'ASL requereixen un moviment específic que no es pot captar amb una sola imatge (vegeu la **Figura 1**). Per això, els sistemes de reconeixement han de tenir en compte tant els gestos estàtics com els dinàmics, i analitzar seqüències de vídeo per entendre correctament el significat complet del signe.

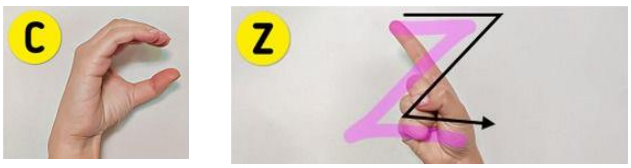

Fig 1. Exemples de signes estaics i dinàmics en ASL

En els següents apartats es detalla els objectius que es volen assolir en aquest projecte, la metodologia utilitzada, la planificació, l'estat de l'art on es recopila informació d'altres treballs relacionats amb el camp i el desenvolupament tècnic sobre la detecció de les mans i les proves realitzades.

2 OBJECTIUS

L'objectiu principal d'aquest projecte és **desenvolupar un sistema capaç de reconèixer automàticament la llengua de signes**, amb la finalitat d'explorar la seva viabilitat tecnològica i establir una base per a futures millores.

Aquest objectiu general es desglossa en diversos **subobjectius específics**, que permeten avançar de manera progressiva cap a un sistema funcional i cada vegada més complet. Aquests subobjectius són els següents:

1. **Cerca de conjunts de dades**: Investigació i selecció de bases de dades públiques (com *Sign Language MNIST*[6], *WLASL*[12] o *How2Sign*[27]) que ofereixin una varietat suficient de gestos i situacions per garantir la qualitat i la generalització del model entrenat.
2. **Reconeixement de mans:** Identificació de la posició i el moviment de les mans mitjançant l'extracció de punts clau tridimensionals, utilitzant biblioteques com *MediaPipe.*
3. **Reconeixement de lletres estatiques:** Detectar lletres de l'alfabet en llengua de signes a partir d'imatges fixes, començant per vocals i ampliant cap a l'abecedari complet, emprant xarxes neuronals convolucionals[20].
4. **Reconeixement de signes dinamics:** Classificar seqüències gestuals mitjançant models temporals com les xarxes LSTM, utilitzant vídeos prèviament enregistrats per entrenar el sistema[10] [28].
5. **Reconeixement de signes en temps real:** Adaptar el sistema al reconeixement de gestos de manera contínua i en temps real, abordant seqüències més complexes com paraules o frases senceres, emprant arquitectures avançades com **xarxes convolucionals 3D**.

3 METODOLOGIA

Per a la implementació del projecte, s'ha optat per una metodologia capaç de gestionar el treball a través d'un conjunt de valors, principis i practiques, dividint la tasca en parts petites i manejables, amb una durada fixa entre les dues i quatre setmanes similar a **SCRUM**[15].
Per a valorar l'evolució del projecte s'han realitzat seguiments cada dues setmanes amb el tutor on s'ha avaluat si el treball realitzat ha sigut l'adient i una documentació clara y eficient de cara als informes.

La implementació del codi es troba disponible en un repositori de GitHub. La branca principal ("main") conté la versió més actualitzada del codi, mentre que les diferents branques reflecteixen el procés i les etapes de desenvolupament que s'han seguit per arribar a aquesta versió.
A través del següent enllaç es pot accedir al repositori que s'ha fet menció: https://github.com/1631153/Sign-Recognition-TFG

4 PLANIFICACIÓ

El projecte es divideix en diverses fases que inclouen la cerca del conjunt de dades, la detecció de mans, la interpretació dels gestos i la posterior producció d'un model. Seguint la metodologia establerta, cada fase s'ha planificat per tenir una duració estimada entre **dues i quatre setmanes** en base al calendari de seguiment d'entregues.

Planificació i busqueda de recursos:
A l'inici del projecte, es durà a terme una **investigació de recursos disponibles** per a la detecció de llengua de signes. En aquesta fase es treballarà la cerca de conjunts de dades, amb l'estudi i selecció de conjunts de dades, així com l'anàlisi de solucions prèvies desenvolupades per la comunitat, expressades en informació recopilada **d'articles informàtics** a més de pagines com **Kaggle** o **GitHub**. També es definirà l'abast funcional del projecte i s'establiran els requisits tècnics i objectius prioritaris.

Un cop establertes les bases de dades i els objectius, es procedirà al desenvolupament d'un sistema per a la **detecció**

de mans i extracció de **punts clau tridimensionals** utilitzant *MediaPipe Hands*[3] establert en el subobjectiu 2 que s'utilitzará durant la realització dels següents subobjectius.

Disseny del sistema
Seguint la metodologia en base al calendari de seguiment d'entregues, en aquesta fase s'iniciarà el disseny modular del sistema. Es desenvoluparan models per al reconeixement de **vocals** i posteriorment de tot l'abecedari mitjançant **imatges estàtiques** (CNN), en coherència amb el subobjectiu 3. Aquest codi servirà de base per a futurs mòduls més avançats.

Desenvolupament del reconeixement de signes
El sistema s'ampliarà per incorporar el reconeixement de **signes dinàmics** corresponents a paraules, entrenant models **LSTM** sobre vídeos etiquetats o punts clau seqüenciats. També s'avaluarà l'ús de tècniques per evitar el sobreajustament i millorar la generalització.

Finalment, s'abordarà l'ultim subobjectiu, **el reconeixement de signes en temps real,** en el que es plantejarà l'ús de codi preexistent per a resultats més complexos que es modificarà de manera significativa per integrar els components previs en un sistema capaç de reconèixer **seqüències contínues** i executar prediccions en temps real.

Desenvolupament final i integració del sistema
Durant les últimes setmanes, es procedirà a la finalització del sistema complet, incorporant les millores identificades durant les fases anteriors. Es documentarà el codi, es generaran gràfiques i taules de resultats, i es prepararà la presentació final del projecte.

La **figura A1**, disponible a l'apendix, mostra el **diagrama de Gantt** amb el detall de totes les tasques planificades que es realitzará al llarg del projecte.

5 ESTAT DE L'ART

El reconeixement automàtic de la llengua de signes és un camp de recerca actiu des de fa dècades. Tot i que no és el primer treball en la matèria, tampoc serà l'últim.

Al llarg dels anys, diversos projectes han abordat aquest repte aplicant enfocaments diferents segons el vocabulari a reconèixer, la quantitat i diversitat de signants, la llengua de signes utilitzada (ASL, LSE, LSC, etc.), així com la qualitat i modalitat dels conjunts de dades emprats (vídeo RGB, profunditat, marcadors 3D, etc.) [16]

Tant el nombre d'estudis publicats com la quantitat de conjunts de dades disponibles han experimentat un creixement notable. Amb els avenços en visió per computador i aprenentatge profund, s'ha produït una transició cap a sistemes que utilitzen exclusivament càmeres i algoritmes d'anàlisi d'imatges o seqüències de vídeo. Aquesta evolució ha permès abordar problemes com la detecció de gestos en temps real o el reconeixement de frases completes. Projectes com *Sign Language MNIST*[6] van ser claus en la classificació de lletres a partir d'imatges estàtiques, mentre que altres iniciatives com *WLASL*[12] o *How2Sign*[27] han proporcionat conjunts de dades amb milers de paraules i frases signades per múltiples persones.

En la **figura A2** que es pot trobar a l'apendix es mostren diversos conjunts de dades, oferint una visió global de la progressió històrica i la cobertura geogràfica d'aquestes iniciatives.

Des que es va assignar el treball, s'ha optat per treballar en un **entorn de desenvolupament basat en Python** gràcies a la seva àmplia varietat de biblioteques especialitzades en visió per computador i aprenentatge automàtic. Al llarg del treball s'utilitzarán eines com **TensorFlow, Keras** o **PyTorch** per al disseny i entrenament de models de xarxes neuronals, **OpenCV** per al processament d'imatges en temps real, i **MediaPipe** per a la detecció precisa de punts clau a les mans.

En aquest context tecnològic, cal tenir en compte que la llengua de signes es pot expressar-se de formes molt diverses. Alguns signes es poden detectar a través d'una sola imatge estàtica, ja que es basen en una posición concreta de la mà o dels dits. En canvi, d'altres signes impliquen **moviment**, **expressió facial** o una **seqüencia d'accions**, i per tant requereixen l'anàlisi de vídeos en temps real per poder ser detectats amb precisió. Aquesta diferència determina els enfocaments tecnològics que es poden aplicar a l'hora de desenvolupar un sistema de detecció de llenguatge de signes i trobem exemples com:

- **Detecció de gestos simples amb OpenCV mitjançant contorns i segmentació de color**. [2] [13]
- **Extracció de punts clau (landmarks) amb MediaPipe.** [3] [4]
- **Classificació de signes mitjançant models entrenats amb dades de landmarks.** [5] [8]
- **Adaptació de models preentrenats amb MobileNet per detectar signes.** [5] [11]
- **Ús de *datasets* públics per entrenar models de detecció de signes.** [6] [7]

Investigadors com **Oscar Koller**[16]**, Necati Cihan Camgoz**[17] **, Juan Carlos Niebles,** entre d'altres, han treballat en el desenvolupament de sistemes de reconeixement automàtic. Els seus treballs han integrat xarxes convolucionals (CNN), xarxes recurrents (RNN) i mètodes més recents com **Graph Convolutional Networks (GCN)** o **Multi-scale Temporal Convolutions** que han permet millorar significativament la precisió en el reconeixement de signes, tant a nivell de paraula com en contextos de frases contínues. A més, han contribuït al desenvolupament i publicació de conjunts de dades oberts, fonamentals per a l'entrenament i avaluació de models més robustos.

En estudis previs, les metodologies emprades han inclòs:

- Ús de CNNs amb datasets d'imatges per a signes estàtics (com ASL Alphabet).

- Arquitectures LSTM o GRU per gestos seqüencials.
- Preprocessament amb **MediaPipe** o **OpenPose** per obtenir punts clau.
- Tècniques d'augment de dades, transferència d'aprenentatge i callbacks com *early stopping* o *learning rate schedulers*

Segons diversos estudis[29] , la tendència dominant actual en el reconeixement automàtic de llengua de signes és l'aplicació de xarxes espaciotemporals, com les **Inflated 3D ConvNets** o **Transformers temporals**, ja que permeten capturar tant la morfologia dels gestos com la seva evolució en el temps. Aquestes arquitectures han demostrat una gran eficàcia en tasques de reconeixement d'accions, i han estat adaptades per a la llengua de signes.

Paral·lelament, com es destaca a *How2Sign*[27], han guanyat rellevància les investigacions orientades a la **traducció automàtica completa** de la llengua de signes a text, més enllà del reconeixement de paraules individuals. Aquest projecte, juntament amb altres aportacions rellevants[16] [18] [19], proporciona dades multimodals alineades entre vídeo, àudio i transcripció, fet que facilita l'entrenament de models més sofisticats i aplicables a contextos reals.

6 DESENVOLUPAMENT

6.1 Bases de dades

6.1.1 Sign Language MNIST

El conjunt de dades utilitzat per a les primeres proves va ser **Sign Language MNIST**[6], el qual conté imatges de 28x28 píxels en escala de grisos. Representa un problema de classificació multiclasse amb **24 categories**, corresponents a les lletres de l'alfabet en **llengua de signes americana**, amb l'excepció de les lletres 'J' i 'Z', que requereixen moviment i, per tant, no van ser incloses.

Les imatges del *dataset* s'emmagatzemen en un fitxer CSV, on cada fila representa una imatge amb una etiqueta numèrica del 0-25 i 784 valors de píxels en escala de grisos de rang 0-255.

6.1.2 Dataset personalitzat

Per abordar el reconeixement de gestos associats a paraules completes, es va optar per construir un **conjunt de dades específic** a partir de seqüències de vídeo enregistrades per l'usuari.

Per a cada acció, es generen 30 seqüències de vídeo, cadascuna amb una durada de 30 fotogrames. Aquestes seqüències s'han processat per tal d'extreure els **punts clau** (landmarks) de les mans mitjançant eines com *MediaPipe*, i emmagatzemarles en fitxers de format .npy[23]. Aquest format binari estàndard de *NumPy* permet guardar matrius amb informació estructurada, incloent la forma i el tipus de dades, facilitant així la posterior lectura i reconstrucció durant l'entrenament del model.

6.1.3 WLASL

El dataset **WLASL (Word-Level American Sign Language)**[25] va ser triat com a conjunt de dades per treballar amb els objectius més complexos. A Kaggle[31] es pot trobar una forma de descarregar-lo directament. Aquest conjunt conté més de **2000 paraules** en ASL representades per intèrprets de diferents edats, gèneres, ètnies, condicions d'il·luminació i fins i tot roba, ja que normalment es fan servir peces fosques ja que facilita a les persones, especialment aquells amb baixa visió, puguin seguir la interpretació sense distraccions. Els vídeos estan acompanyats d'un fitxer JSON[26] que recull metainformació com *fps, url i video_id*, facilitant la cerca i l'organització del material.

6.2 Detector de mans

Abans de poder reconèixer signes, és essencial disposar d'un mètode robust per identificar la posició i el moviment de les mans dins d'un vídeo. La detecció precisa de mans proporciona les dades bàsiques que permeten alimentar models més complexos de classificació.

En aquest context, es va començar a elaborar un **tracker de mans** amb *MediaPipe Hands* per a futures implementacions que funciona dibuixant 21 punts crítics en forma de cercles vermells sobre les mans en una posició fixa tal com es mostren a la **Figura 2.** Tot i que el tracker ja ve implementat en la llibreria de *MediaPipe*, es necessari adaptar-ho minimament a les especificacions del projecte[9].

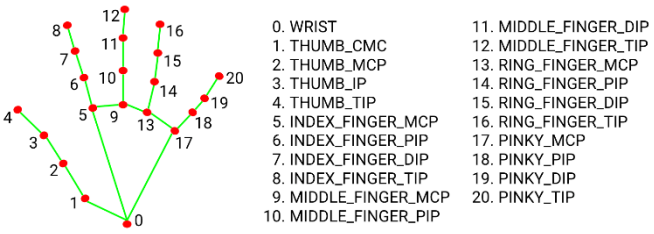

Fig 2. Punts clau de la mà per Mediapipe

6.3 Alfabet dactilològic

Per al reconeixement de l'alfabet en llengua de signes, s'ha treballat sobre el conjunt de dades **Sign Language MNIST**[6], on les dades s'han dividit en un 80% per a l'entrenament i un 20% per a la validació.

Inicialment, es va desenvolupar un model bàsic basat en una **xarxa neuronal convolucional (CNN**). Aquesta arquitectura consta de tres capes convolucionals amb 128, 64 i 32 filtres respectivament, seguides de capes de MaxPooling, Flatten i dues capes densament connectades. Per construir el model, s'han utilitzat principalment les biblioteques *TensorFlow i Keras*, que han permès definir i entrenar l'estructura amb flexibilitat i eficiència.

Tot i la bona precisió assolida en els resultats (97,67%. Veure la **Taula 1**), el model no presenta un comportament òptim quan s'aplica en condicions de predicció en temps real.

Per tal de millorar el rendiment, s'ha optat per una segona estratègia d'entrenament, basada en **transfer learning** utilitzant **VGG19**[21], preentrenada amb imatges generals, però adaptada a la classificació de signes, a més d'extraure una **regió d'interès** (ROI) delimitada per un requadre verd[10], que redimensiona i normalitza el frame abans de passar-la pel model (veure **Figura 4**) amb l'objectiu d'ajustar les imatges d'entrada al format del dataset MNIST.

A continuació es mostra una comparació entre les dues etapes d'entrenament:

Paràmetre	Entrenament bàsic (CNN)	Entrenament amb VGG19 (Transfer Learning)
Biblioteques utilitzades	TensorFlow i Keras	TensorFlow i Keras
Arquitectura de la xarxa	3 capes convolucionals (128, 64, 32 filtres), MaxPooling, Flatten, 2 capes denses	Mateixa arquitectura que a l'entrenament
Número d'èpoques	15	5
Batch size	200	952
Temps d'entrenament	Aproximadament 30 minuts	Aproximadament 2 hores i 50 minuts
Accuracy	97.67%	85%
Pèrdua (loss)		0.30
Objectiu de millora	Captura de detalls subtils amb tercera capa convolucional addicional	Avaluar la capacitat de generalització del model en condicions noves
Model de referència	Basat en *Sign-Language Classification CNN (99.40% Accuracy)* [15]	Basat en *Sign-Language Traslator* [10]

Taula 1. Comparació entre estratègies d'entrenament de model

Tot i que *l'accuracy* obtinguda amb el primer model es més elevada, el segon entrenament ofereix una estructura més solida per a escenaris pràctics, ja que incorpora millores automàtiques durant l'aprenentatge i una arquitectura amb major capacitat representativa.
Tal com es mostra a la **Figura 3**, la precisió de les prediccions es manté elevada fins i tot quan el model s'aplica sobre imatges capturades en temps real, amb variacions en la il·luminació, la distància o el fons.

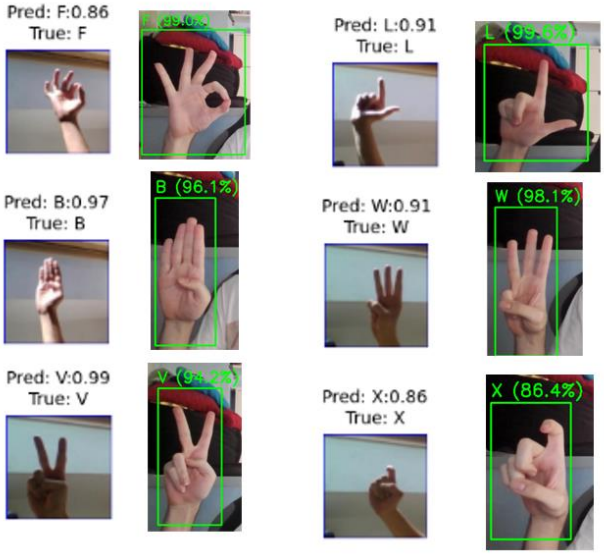

Fig 3. Comparació de percentatge de la predicció de signes

Durant les proves finals, s'observa que el programa tendeix a detectar millor la mà esquerra. Aquest comportament no es deu a una preferència del model, sinó al fet que la càmera utilitzada opera en **mode mirall**, invertint lateralment la imatge en temps real. Per tal d'uniformitzar les entrades i garantir una detecció coherent, quan el sistema identifica que la mà detectada és la dreta, la imatge es volteja horitzontalment abans de passar-la al model. D'aquesta manera, es manté la consistència en la interpretació dels gestos independentment de la mà utilitzada. El sistema mostra el resultat de la predicció superposat sobre el vídeo en temps real, indicant la lletra detectada al costat del requadre de la mà.

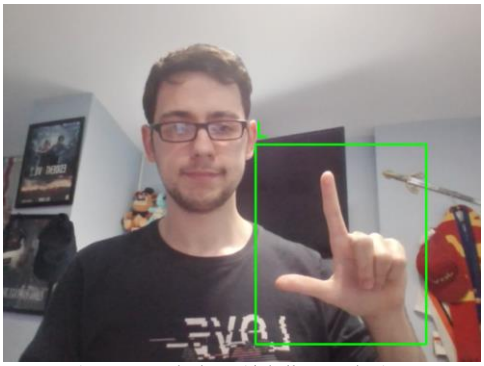

Fig 4. Prova de detecció de llengua de signes

6.4 Detector de signes amb moviment

6.4.1 Estudi previ del reconeixement de gestos

Per abordar aquest repte, es va plantejar l'ús d'una xarxa neuronal profunda combinada amb tècniques avançades de preprocessament, en línia amb els enfocaments descrits a l'apartat de **Estat de l'art**. Arquitectures com **ResNet**, **Inception**, **Xception** o **VGG** han demostrat ser útils per millorar la classificació de gestos i optimitzar el rendiment dels sistemes de reconeixement automàtic[22].

No obstant, s'ha decidit provar un enfocament alternatiu al tractament directe d'imatges mitjançant CNN, consistent en l'ús de **punts clau tridimensionals (landmarks)** obtinguts a partir del seguiment corporal. Aquesta aproximació, basada en informació recopilada[30], permet representar seqüències de moviment mitjançant estructures numèriques eficients i donar un enteniment de com funcionen aquests models.

Es defineixen tres classes d'accions representades per dades: *Please, Thanks i Iloveyou*, fent ús del conjunt de dades explicat a **l'apartat 6.1.2.**

El model d'aprenentatge profund construït amb Keras està compost per una xarxa seqüencial de tipus LSTM (**Long Short-Term Memory**)[24], dissenyada per captar la naturalesa temporal de les seqüències gestuals. (vegeu **Figura 5**, on es mostra l'esquema del flux d'entrada, processament i classificació del model).

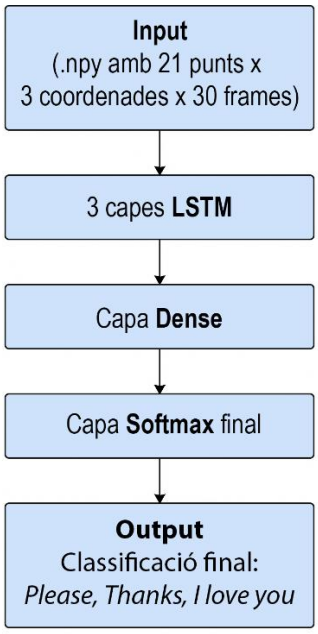

Fig 5. Esquema del model LSTM per a reconeixement d'accions

L'entrenament del model s'ha dut a terme durant 2000 èpoques amb un total de 3 lots (batches) per època, amb un temps d'execució d'aproximadament 15 minuts. El model aconsegueix una **precisió de validació del 80%**, amb una evolució progressiva del rendiment al llarg de l'entrenament. Per a la millora del comportament i la prevenció de sobreajustaments, s'apliquen estratègies com la monitorització amb **TensorBoard**, que permeten registrar la pèrdua i la precisió (*accuracy*) en cada època, així com la configuració de callbacks específics com **ModelCheckpoint** i **ReduceLROnPlateau**.

Amb aquest enfocament, s'obtenen resultats prometedors que, tot i no tenir aplicació directa com a sistema final, son essencials per **entendre el funcionament intern del reconeixement de gestos manuals** i **experimentar amb la representació i predicció de seqüències gestuals**. A més, es va implementar una condició de prova en què, si la probabilitat de la predicció superava un llindar de 0.8, el signe reconegut es mostrava a la pantalla com a text (vegeu Figura 6), amb l'objectiu de **visualitzar el comportament del model.**

Per visualitzar el funcionament d'aquesta versió del codi s'ha fet una demostració d'un minut i es pot trobar al seguent enllaç:
https://www.youtube.com/watch?v=eA_JtX0rEds

Fig 6. Captura de demostració de llengua de signes.

6.4.2 'World Level American Sign Language' Dataset

Amb l'objectiu de millorar la capacitat de reconeixement i generalització dels models, s'ha plantejat l'ampliació de la quantitat i varietat de dades, en línia amb els criteris descrits a l'apartat de *Estat de l'art*, on es destaca la importància de la diversitat en els conjunts de dades per assolir una millor generalització.

Per aquesta fase s'utilitza el conjunt de dades de WLASL que, un cop descarregat, es du a terme l'extracció de punts clau de les mans mitjançant la llibreria **MediaPipe**, i generat fitxers .npy per al seu ús posterior en l'entrenament. A partir d'una selecció de **12.000 vídeos**, es genera un subconjunt processable orientat a la prova de concepte. Aquest subconjunt incloïa paraules com *able*, *actor* o *adjust*.

Posteriorment, es construeix un **model seqüencial** basat en l'esquema vist a la **Figura 5**, dissenyat per captar la dependència temporal de seqüències gestuals. El model es prova amb diferents combinacions de paraules disponibles al conjunt de dades filtrat, incloent-ne algunes que formen estructures sintàctiques simples. Per exemple, es va testejar el reconeixement de la frase "*My name is Javi*" per avaluar la capacitat del sistema de captar relacions entre signes consecutius.

Inicialment es construeix una arquitectura LSTM amb diverses capes, però amb tan poca quantitat de dades, **no es podía capturar la variabilitat natural dels gestos**, que poden canviar segons la persona, la il·luminació, la velocitat del moviment o el fons del vídeo. Per això, es decideix simplificar l'arquitectura per adaptar-la millor a la mida del conjunt de dades.

A continuación, es presenten els resultats quantitatius de les dues configuracions d'entrenament testejades, comparades a la **Taula 2**.

Mètrica	Model inicial	Model millorat
`loss`	0.2648 – Aprèn bé en entrenament	1.0724 – Més raonable, no memoritza
`categorical_accuracy`	0.8571 – Alta precisió (possible sobreajust)	0.4762 – Millor, però amb marge
`val_loss`	24.9665 – Molt alt, generalitza malament	1.9072 – Molt més baix
`val_categorical_accuracy`	0.1667 – Pitjor que l'atzar	0.3333 – Ara prediu correctament 2 de 6 classes
`lr` *(learning rate)*	1e-5 – Possiblement caigut al mínim	0.001 – Encara no ha estat reduït automàticament

Taula 2. Comparació entre dues configuracions d'entrenament.

Amb el **nou model LSTM reduït**, s'observa una millora parcial:

- El sistema aconsegueix reconèixer correctament dues de les sis classes definides.
- Aquestes paraules assoleixen **un llindar de confiança superior al 50%** de forma consistent.

Malgrat les millores, el rendiment continua limitat per la **escassa quantitat de mostres per classe** tal i com es pot veure en el percentatge representat en la **Figura 7**. Tot i que el conjunt **WLASL** és extens, si només s'utilitzen 3–4 exemples per paraula, el model no pot aprendre patrons generalitzables. Aquesta situació no reflecteix una deficiència del dataset, sinó una necessitat d'ajustar l'estratègia d'entrenament per aprofitar millor el seu potencial. En l'estat actual, el model tendeix a memoritzar característiques visuals particulars dels vídeos d'entrenament, en lloc de captar patrons gestuals comuns.

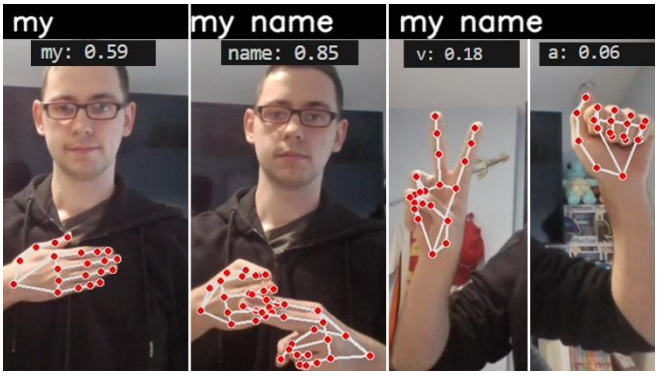

Fig 7. Captures de demostració de llengua de signes.

6.5 Estudi preeliminar amb xarxes convolucionals 3D

A causa dels resultats poc satisfactoris obtinguts amb els enfocaments seqüencials tradicionals de les xarxes LSTM, es decideix descartar aquest procediment i explorar una alternativa més efectiva: les **xarxes convolucionals tridimensionals** (3D CNN). Aquest nou enfocament permet capturar simultàniament la informació espacial i temporal dels vídeos de llengua de signes, superant les limitacions dels mètodes anteriors. Per al desenvolupament del model, s'ha pres com a referència la implementació publicada al repositori de GitHub de WLASL[12], la qual fa ús de la xarxa **Inception I3D (Inflated 3D ConvNet)**.

Aquest model es basa en la xarxa *Inception-V1*, però millora l'anàlisi de vídeos afegint convolucions en tres dimensions. Això li permet examinar els fotogrames com un conjunt i detectar patrons de moviment, útils per identificar gestos i transicions. A més, incorpora tècniques de max-pooling temporal i classificació a nivell de fotograma i vídeo complet, cosa que fa que el seu entrenament sigui més estable. Aquest model millora el reconeixement de llengua de signes perquè, en lloc de classificar caràcters un per un com feia el sistema LSTM, **identifica paraules completes i analitza el moviment global del gest**. Les xarxes LSTM tracten cada fotograma per separat dins la seqüència, cosa que pot dificultar l'anàlisi en vídeos complexos. En canvi, les convolucions 3D combinen la informació alhora, fent-les més eficients per a aquesta tasca.

Partint d'aquesta base, es realitzen modificacions importants en el codi original per adaptar-lo a les necessitats específiques d'aquest projecte. S'ha volgut establir tres canvis importants:

- Canviar l'estructura del bucle d'entrenament i validació.
- Utilitzar només *epochs* en comptes de *steps*.
- Afegir altres mètriques.

Un dels primers canvis ha estat el d'afegir una mètrica més completa amb **l'F1-score**, a més de l'accuracy. Això ha permès una millor avaluació del rendiment, sobretot amb conjunts de dades desequilibrats, on alguns signes tenen menys exemples que altres tal i com s'explica a l'apartat anterior.

A nivell d'entrenament, s'ha canviat la funció de pèrdua *CrossEntropyLoss* per **BCEWithLogitsLoss**, que funciona millor amb etiquetes en format *one-hot*. Aquesta nova funció permet una classificació més estable i robusta durant la fase inicial d'aprenentatge.

Mentre el codi original utilitza un scheduler de pas fix (*StepLR*), la versió més recent implementa **ReduceLROnPlateau**, utilizat en codis anteriors, es un programador de taxa d'aprenentatge adaptatiu reutilitzat que ajusta la taxa quan la validació no millora, evitant sobreentrenament i fent l'entrenament més eficient. Aquest enfocament no contempla directament la gestió del scheduler ni el desament automàtic del model dins de les funcions, i deixa aquestes responsabilitats al codi extern que orquestra el cicle d'entrenament complet.

En aplicar els nous canvis també es contempla un canvi en l'estratègia d'entrenament a l'utilitzar només les epochs, és a dir, passar per tot el conjunt de dades diverses vegades, en lloc de fer-ho en un nombre fix de passos. Això és considera útil perquè el model veu tots els exemples abans de fer ajustaments com la taxa d'aprenentatge o les avaluacions. Així, els resultats són més estables i no es basen només en una part reduïda de les dades.

7 RESULTATS

Al llarg de les diverses iteracions realitzades del projecte s'han obtingut resultats que han estat clau per entendre què era el que funcionava i què calia millorar. Cada prova i anàlisi ha aportat informació valuosa, permetent identificar els problemes i ajustar estratègies de manera progressiva.

Cal destacar que, per tal de dur a terme els entrenaments i recollir els resultats de manera eficient, s'ha utilitzat la **GPU del Departament de ciencies de la computació de la UAB**, que ha permès reduir considerablement els temps de processament i facilitar les proves a gran escala.

A la **Taula 3** es poden observar l'evolució constant dels canvis realitzats sobre el codi original (Train_i3d)[12] amb la intenció de millorar-ho o apropar-se a un resultat semblant.

Mentre que el codi original implementa un entrenament iteratiu amb acumulació de gradients i doble pèrdua usant **binary_cross_entropy_with_logits**, on es valida cada època per step i es guarda el millor model segons la seva precisió, les altres versions apliquen diferents canvis amb la intenció de diferenciar-se:

- **Javi_v1:** S'han separat les funcions d'entrenament i validació, afegint **early stopping** i càlcul de precisió.
- **Javi_v2:** S'ha passat a un entrenament **més clàssic per èpoques**, amb mètriques detallades **(accuracy i F1)** i estructura més clara.
- **Javi_v3:** S'ha fet **fine-tuning entrenant** només la capa *Mixed_5c*, deixant la resta congelada.
- **Javi_v4:** S'ha afegit **balanceig de classes**, mètriques avançades i early stopping per evitar sobreentrenament.
- **Javi_v5:** S'ha simplificat l'esquema: entrenament complet sense pesos, només amb **CrossEntropyLoss**, guardant el millor model.
- **Javi_v6:** S'ha canviat a un *scheduler* adaptatiu (**ReduceLROnPlateau**), amb **seguiment per batch** i early stopping allargat.
- **Javi_v7:** S'ha modificat la pèrdua a **BCEWithLogitsLoss** amb etiquetes one-hot i s'ha eliminat l'early stopping per fer el **control del sobreajustament més senzill.**

Subconjunt 100	Accuracy	F1 Score	Temps
Train_i3d	**74.80%**		**16h aprox**
Javi_v1	40.70%		10h aprox
Javi_v1-2	2.33%	0.49%	3h aprox
Javi_v3	29.07%	25.18%	21h aprox
Javi_v4	45.74%	42.90%	48h aprox
Javi_v4-2	1.55%	0.05%	10h aprox
Javi_v6	2.71%	0.49%	2h aprox *(early stopping)*
Javi_v7	**77.91%**	**78.68%**	**6h aprox**

Taula 3. Taula de resultats de l'entrenament.

D'aquesta manera, durant la fase d'entrenament del model final –realitzada amb 50 epochs– s'ha pogut superar els resultats de referencia del codi original tal i com es mostra en la Taula 3. Tot i que l'increment en el rendiment no ha estat significatiu, aquests canvis implementats han demostrat ser útils.

Tanmateix, és important remarcar que els resultats recollits a la Taula 3 corresponen a la fase d'entrenament i validació, i no reflecteixen amb exactitud el comportament real del model en un entorn de test. Per aquest motiu, es va realitzar una avaluació sobre conjunts de dades separats per obtenir una mesura més realista del rendiment generalitzat dels models.

Els resultats d'aquesta avaluació final es mostren a la Taula 4. Aquesta taula mostra clarament una disminució generalitzada del rendiment en comparar amb els valors d'entrenament. Concretament, el model **Train_i3d** conserva una millor robustesa i manté una precisió superior a mesura que augmenta la mida del subconjunt (del 72.09% al 31.68%). Per contra, el model **Javi_v7**, tot i haver assolit millors resultats durant l'entrenament en el subconjunt 100, presenta una caiguda molt més pronunciada en precisió a mesura que augmenta la mida del test, arribant només a un 2.63% en el subconjunt 2000.

Model	Subconjunt 100	Subconjunt 300	Subconjunt 1000	Subconjunt 2000
Train_i3d	72.09%	64.97%	47.33%	31.68%
Javi_v7	70.54%	56.14%	13.65%	2.63%

Taula 4. Taula de resultats de test.

Aquesta diferència posa de manifest que un bon rendiment durant l'entrenament no sempre es tradueix en una bona generalització.

De fet, diversos estudis en l'àmbit del reconeixement automàtic de llengua de signes reflecteixen una tendència similar: la dificultat de mantenir una alta precisió a mesura que augmenta el vocabulari. Per exemple, en l'article *One Model is Not Enough*[32], que parla d'un enfocament basat en **enensembles d'I3D, TimeSformer i SPOTER** va aconseguir fins al **60.66% de precisión top-1 en el subconjunt de 300.** A més, arquitectures com **StepNet** amb fusió RGB + flux òptic van assolir un **61 % de top-1 i 91 % de top-5** en el conjunt WLASL-2000[33].

Així doncs, el comportament del codi Javi_v7 és coherent i a mesura que es vol reconèixer un vocabulari més gran, el problema es fa més complicat i es necessari de tècniques més robustes per evitar errors i millorar els resultats.

8 CONCLUSIONS

El desenvolupament d'aquest treball ha posat en evidència la complexitat real que implica crear un sistema funcional de reconeixement de llenguatge de signes, especialment

quan es treballa en temps real i amb seqüències de moviment. Tot i partir d'un enfocament inicial més senzill, basat en la classificació d'imatges estàtiques mitjançant xarxes convolucionals (obtenint un *85% d'accuracy* en els resultats) i l'ús de models seqüencials com els LSTM per captar patrons temporals dins de vídeos (amb un *80% d'accuracy)*, s'ha identificat la necessitat d'utilitzar models més potents per millorar la capacitat d'extracció i representació de la informació visual i temporal.

Durant l'execució del projecte, s'han provat diferents fonts de dades, com WLASL i How2Sign, amb processos de filtratge, etiquetatge i transformació dels vídeos en seqüències de punts clau utilitzant *MediaPipe*. Aquesta evolució ha estat essencial per afrontar les limitacions dels datasets més senzills o desequilibrats, tot i que encara han persistit dificultats com el sobreajustament, la poca variabilitat gestual i la presència de classes amb comportaments molt similars entre si.

Per superar aquestes limitacions, s'ha incorporat l'ús de la xarxa Inception I3D. Aquesta nova aproximació permet una representació més rica i precisa dels moviments, millorant la detecció i classificació respecte als enfocaments anteriors basats en LSTM o xarxes convolucionals estàtiques.

Els resultats obtinguts, especialment en el **subconjunt de 100 paraules** (amb un **70.54%** per al model desenvolupat), demostren que, en escenaris acotats, el sistema pot oferir una **detecció funcional**. Tanmateix, la **caiguda progressiva del rendiment a mesura que augmenta la mida del vocabulari** (fins a només un 2.63% en el subconjunt de 2000 amb el model Javi_v7) posa de manifest les dificultats associades a l'escalabilitat del sistema i a la generalització en contextos més realistes.

Tot i els avenços assolits, el reconeixement automàtic de la llengua de signes a gran escala encara representa un repte. Malgrat això, les fites tècniques aconseguides durant el projecte ofereixen una **base sòlida** per continuar millorant, optimitzant i ampliant el sistema, tant pel que fa a la detecció com a la traducció automàtica.

8.1 Treball a futur

Un dels objectius més rellevants en el que es podria treballar més endavant és la possibilitat d'integrar el sistema amb un mòdul de detecció d'emocions. Aquesta funcionalitat permetria interpretar no només el significat literal dels gestos, sinó també el context emocional en què es realitzen. Per exemple, la mateixa senya pot tenir matisos diferents si es realitza amb una expressió facial alegre, neutra o trista.

Fig 8. Exemple de signes i emocions per escuelas.excepcionales.es

Una altra implementació és l'adaptació de la solució a dispositius mòbils, com ara telèfons intel·ligents, que permetria ampliar el seu abast i fer-la accessible a un públic molt més ampli. Això implica treballar en l'optimització del model perquè pugui funcionar amb recursos computacionals més limitats, així com garantir una interfície d'usuari intuïtiva i funcional.

Es considera també la possibilitat d'adaptar el sistema al castellà, aprofitant recursos ja existents com la base de dades desenvolupada per la **Universitat de València**[14].

AGRAÏMENTS

Voldria aprofitar aquest espai per agrair als meus companys i amics que han contribuït en aquest projecte amb les seves opinions. Gràcies per mostrar interès en ajudar-me, fins i tot posant a prova el programa quan encara estava en fase de proves. La vostra aportació, per petita que hagi sigut, ha estat clau per construir aquesta idea en una experiència amb sentit, de la qual em sento orgullós.

També vull expressar el meu més sincer agraïment a la meva tutora de TFG, Gemma Sánchez, per el seu suport i la seva visió del projecte, així com per guiar-me al llarg de tot el procés.

BIBLIOGRAFIA

[1] *IELSE*. "Lengua de signos" *Instituto de Enseñanza de la Lengua de Signos Española*. Disponible a: https://ielse.es/lengua-de-signos-espanola/

[2] N. Renotte. *Real Time Sign Language Detection with Tensorflow Object Detection and Python*. YouTube, accedit el 26 de març de 2025. Disponible a: https://www.youtube.com/watch?v=pDXdlXlaCco

[3] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, M. Grundmann. "MediaPipe Hands: On-device Real-time Hand Tracking." *arXiv preprint arXiv:2006.10214*, 2020. https://arxiv.org/pdf/2006.10214

[4] "Hand Tracking 30 FPS on CPU in 5 Minutes." *Medium*. Disponible a: https://medium.com/augmented-startups/hand-tracking-30-fps-on-cpu-in-5-minutes-986a749709d7

[5] M. Bali. "Sign Language Detection for Deaf using Deep Learning, MediaPipe & OpenCV." *Medium*. Disponible a: https://medium.com/@mayank.bali/sign-language-detection-for-deaf-using-deep-learning-mediapipe-u-opencv-4c5151e2374c

[6] "Sign Language MNIST." *Kaggle*. https://www.kaggle.com/datasets/datamunge/sign-language-mnist

[7] "ASL Alphabet." *Kaggle*. https://www.kaggle.com/datasets/grassknoted/asl-alphabet

[8] Aprende e Ingenia. "Lengua de señas con visión artificial." *YouTube*, accedit el 29 de març de 2025. Disponible a: https://www.youtube.com/watch?v=CxniTT0qe1A

[9] Aprende e Ingenia. "Detección y clasificación de manos."

YouTube, accedit el 3 de març de 2025. Disponible a: https://www.youtube.com/watch?v=RMI1vjFvV14

[10] E. Cabana. "Sign Language Translator." *GitHub,* accedit el 13 de març de 2025. https://github.com/ecabestadistica/sign-language-translator-python-opencv

[11] N. Renotte "Sign Language Detection using Action Recognition." *YouTube*, accedit l'1 d'abril de 2025. Disponible a: https://www.youtube.com/watch?v=doDUihpj6ro

[12] "WLASL." *GitHub*. https://github.com/dxli94/WLASL

[13] "Sign Language Recognition." *GitHub*. https://github.com/sumedhsp/Sign-Language-Recognition

[14] E. Gutierrez-Sigut, B. Costello, C. Baus, M. Carreiras. "LSE-Sign: A lexical database for Spanish Sign Language." *Springer*, 2016. https://link.springer.com/article/10.3758/s13428-014-0560-1

[15] Atlassian. "¿Qué es Scrum?" *Atlassian Agile Coach*. https://www.atlassian.com/es/agile/scrum

[16] O. Koller, et al. "Quantitative Survey of the State of the Art in Sign Language Recognition." *arXiv preprint arXiv:2008.09918*. https://arxiv.org/pdf/2008.09918

[17] N. C. Camgoz, et al. "Sign Language Transformers: Joint End-to-End Sign Language Recognition and Translation." *arXiv preprint arXiv:2003.13830*. https://arxiv.org/pdf/2003.13830

[18] A. Vazquez-Enriquez, et al. "Isolated Sign Language Recognition with Multi-Scale Spatial-Temporal Graph Convolutional Networks." *CVPR 2021 Workshops*. https://openaccess.thecvf.com/content/CVPR2021W/ChaLearn/papers/Vazquez-Enriquez_Isolated_Sign_Language_Recognition_With_Multi-Scale_Spatial-Temporal_Graph_Convolutional_Networks_CVPRW_2021_paper.pdf

[19] X. Cabot, et al. "Sign Language Translation based on Transformers for the How2Sign Dataset." *UPC - Image Processing Group*. https://imatge.upc.edu/web/sites/default/files/pub/xCabot22.pdf

[20] Sayak Dasgupta. "Sign Language Classification CNN (99.40% accuracy)." *Kaggle*. https://www.kaggle.com/code/sayakdasgupta/sign-language-classification-cnn-99-40-accuracy

[21] KeepCoding. "Arquitectura VGG19 en Deep Learning." *KeepCoding Blog*, 2024 https://keepcoding.io/blog/arquitectura-vgg16-vgg19-deep-learning

[22] A. E. M Ridwan, M. I. Chowdhury, M. M. Mary, Md. T. C. Abir. "Deep Neural Network-Based Sign Language Recognition." *arXiv*. https://arxiv.org/abs/2409.07426

[23] NumPy. ".npy Format – NumPy Docs." *numpy.org*. https://numpy.org/devdocs/reference/generated/numpy.lib.format.html

[24] Y. Bengio, P. Simard, P. Frasconi. "Learning long-term dependencies with gradient descent is difficult." *IEEE Transactions on Neural Networks*, 1994. https://ieeexplore.ieee.org/document/279181

[25] Dxli94. "WLASL Dataset." *GitHub*, accedit el 25 de juny de 2025. https://dxli94.github.io/WLASL/

[26] Wikipedia. "JSON." *Wikipedia, la enciclopedia libre*. https://es.wikipedia.org/wiki/JSON

[27] "How2Sign Dataset." *GitHub*, accedit el 8 de maig de 2025 https://how2sign.github.io/

[28] R. Rastgoo, K. Kiani, S. Escalera. "Sign Language Recognition: A Deep Survey." *Expert Systems with Applications*, 2021. https://doi.org/10.1016/j.eswa.2020.113794

[29] D. Li, C. R. Opazo, X. Yu, H. Li. "Word-level Deep Sign Language Recognition from Video." *arXiv preprint arXiv:1910.11006*, 2020. https://arxiv.org/pdf/1910.11006

[30] Nicknochnack. "Action Detection." *GitHub,* accedit el 13 d'abril de 2025. https://github.com/nicknochnack/ActionDetectionforSignLanguage

[31] "WLASL Complete." *Kaggle*. https://www.kaggle.com/datasets/utsavk02/wlasl-complete

[32] M. Hrúz, I. Gruber, J. Kanis, M. Bohácek, M. Hlavác, Z. Krnoul. "One Model is Not Enough: Ensembles for Isolated Sign Language Recognition." *mdpi*, 2022. https://www.mdpi.com/1424-8220/22/13/5043

[33] X. Shen, Z. Zheng, Y. Yang. "Spatial-temporal Part-aware Network for Isolated Sign Language Recognition." *arXiv preprint arXiv:2212.12857*, 2024. https://arxiv.org/html/2212.12857v2#S1

APENDIX

	Setmana 1	Setmana 2	Sermana 3	Setmana 4	Setmana 5	Setmana 6	Setmana 7	Sermana 8	Setmana 9	Setmana 10	Setmana 11	Setmana 12	Setmana 13	Setmana 14	Setmana 15	Setmana 16	Setmana 17	Setmana 18	Setmana 19	Setmana 20
Analisis de la informació recopilada																				
Especificació d'objectius																				
Cerca de una base de dades																				
Detector de mans																				
Entrega Informe Inicial																				
Definir objectius i metodologia																				
Disseny del sistema																				
Desenvolupament del sistema (abecedari)																				
Desenvolupament del sistema (paraules)																				
Entrega Informe Progrés I																				
Desenvolupament del sistema en datasets grans																				
Entrega Informe Progrés II																				
Implementació CNN 3D																				
Entrenament dels models																				
Implementació de millores																				
Elaboracio Informe Final																				
Proposta de presentació																				
Entrega Informe Final																				
Entrega Dossier																				

Figura A1. Taula del Diagrama de Gantt del projecte

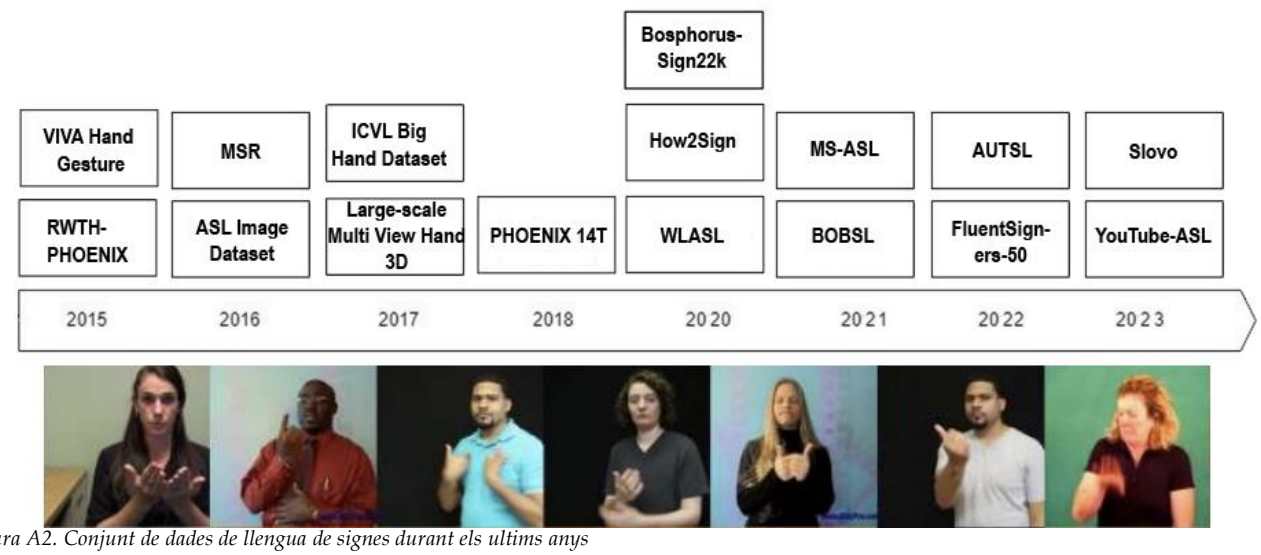

Figura A2. Conjunt de dades de llengua de signes durant els ultims anys